

Trabajo Fin de Grado

Universidad Carlos III de Madrid | Escuela Politécnica
Superior | Grado en Ingeniería Informática



Desarrollo de una aplicación para la gestión de
proyectos no gubernamentales

FUNDACIÓN MADRINA

Autor: Sofía Torres Mendiola

Tutor: Jorge Luis Morato Lara



Resumen

Este proyecto presenta el desarrollo de una aplicación web para la Fundación Madrina que mejora la gestión de voluntarios, trabajadores y actividades.

Actualmente el sitio presenta carencias tanto tecnológicas como de organización y adaptación. El objetivo de este proyecto se enmarca en la demanda de la fundación Madrina por actualizar y hacer más intuitivo el uso de su sitio web, desarrollando una aplicación web que incluya las funcionalidades de gestión y administración de voluntarios, trabajadores y actividades. A lo largo del proyecto se ha seguido metodología SCRUM, con entregas cada cuatro semanas en la que se evaluaba el trabajo y se iban añadiendo nuevas funcionalidades según las necesidades del cliente.

Los resultados de este proyecto han sido muy positivos. La evaluación por parte de los miembros de la Fundación ha resaltado una mejor interacción con la plataforma, no solo debida a criterios estéticos, sino también a la arquitectura del sitio y las nuevas funcionalidades incorporadas. La evaluación de usuarios ajenos también ha sido bastante positiva, ya que sus valoraciones han demostrado que es funcional y no han tenido problemas para su utilización. La asociación utilizará la nueva aplicación desarrollada y podrá conseguir mejores resultados y facilitar su trabajo.

Tras la realización de este proyecto he podido demostrar mis habilidades personales en relación con el diseño y desarrollo de un proyecto de ingeniería informática completo como parte final de mi formación académica, aprendiendo nuevos conceptos y demostrando los aprendidos durante estos últimos cuatro años en la Universidad.



Summary

The aim of this project is the development of a web application for *Fundación Madrina*. This application will allow this NGO to easily manage the planning of its volunteers, workers and activities.

Nowadays, the *Fundación Madrina* website can be considered as an obsolete web from the technological point of view because it is too simple, its contents are bad organised and it is not responsive. The objective of this project is to improve and update this website in order to make it more intuitive and user-friendly by developing a web application that includes the functions to manage and administrate volunteers, workers and activities. In order to achieve this, the SCRUM methodology has been applied through the project development, each four weeks deliveries were made and evaluated and new functionalities were integrated according to the customer's requirements.

The results of this project have been extremely positive and satisfactory, the members of this NGO selected to evaluate the application are very satisfied the new website interface, architecture and functionalities integrated in this application. The external evaluation made by users unconnected with *Fundación Madrina* has also been very positive since their valuations show that this is a very functional and user-friendly application. The association employees will use the new application and will obtain better results because it will be easier the management and administration processes.

Thanks to the development of this project, I have been able to show my personal skills and abilities to the design and development of a whole software engineering project as the culmination of my university studies by learning new concepts and applying those learnt over the last four years at university.



Agradecimientos

En primer lugar me gustaría agradecer a mi familia todo el esfuerzo, el apoyo y confianza que ha depositado en mí desde el primer día que entré en la Universidad. Especialmente a mis padres y a mi hermana por esos ánimos que me han dado cada día, épocas de exámenes, agobios en los que siempre recibía ¡Verás cómo todo va bien! ¡Suerte! sin ellos no hubiera conseguido llegar hasta aquí.

Agradecer a mi tutor del proyecto Jorge por confiar en mí para realizar este proyecto y por todo el tiempo que ha dedicado en ayudarme.

A mis compañeros de la universidad por todos los buenos ratos que me han hecho pasar, por todo lo que he aprendido de ellos.

A los miembros de la Fundación Madrina por confiar en mí, para la realización de este proyecto. En especial a Manuel que se ha encargado de estar en contacto conmigo durante todo este tiempo.

Muchas gracias a todos.

Índice de Contenido

Resumen	2
Summary.....	3
Agradecimientos.....	4
Índice de Contenido.....	5
Índice de tablas	8
Índice de ilustraciones.....	13
1 Introducción	15
1.1 Motivación	15
1.2 Objetivos.....	15
1.3 Estructura del documento.....	17
2 Estado del arte.....	18
2.1 Gestor de contenidos	18
2.2 Base de Datos	22
2.2.1 Tipos de bases de datos.....	22
2.2.2 Las bases de datos a día de hoy.....	23
2.2.3 Ventajas e inconvenientes de los sistemas de bases de datos.....	25
2.2.4 Tipos de usuarios de las bases de datos.....	27
2.3 Lenguajes a utilizar	28
2.3.1 MySQL.....	28
2.3.2 PHP.....	29
2.3.3 HTML.....	30
2.3.4 CSS.....	31
2.4 Análisis de sitios similares.....	33
3 Marco de trabajo	36
3.1 Metodología.....	36
3.1.1 Ciclo de vida.....	36
3.1.2 Roles.....	37
3.1.3 Sprints.....	37
3.2 Planificación.....	38
3.3 Presupuesto.....	38
3.3.1 Salario de los participantes	39
3.3.2 Amortizaciones	40
3.3.3 Costes	41
3.3.4 Resumen presupuesto final.....	42
3.3.5 Garantía.....	42
4 Impacto ético, social, económico y legal.....	43
5 Análisis.....	44
5.1 Situación de partida	44
5.2 Requisitos	44



5.2.1	Requisitos funcionales.....	46
5.2.2	Requisitos de interfaz.....	51
5.2.3	Requisitos de rendimiento	52
5.2.4	Requisitos de operación.....	53
5.2.5	Requisitos de recursos	53
5.2.6	Requisitos de comprobación.....	54
5.2.7	Requisitos de aceptación de las pruebas.....	55
5.2.8	Requisitos de daño.....	56
5.2.9	Requisitos de seguridad	56
5.2.10	Requisitos de mantenimiento	57
5.3	Análisis de casos de uso	57
6	Diseño.....	67
6.1	Definición de la arquitectura del sistema.....	67
6.1.1	Definición de niveles de arquitectura.....	67
6.1.2	Especificación de estándares y normas de diseño y construcción.....	68
6.1.3	Especificación de excepciones	69
6.2	Funcionamiento y taxonomía de la interfaz.....	70
6.3	Diseño de clases	94
6.4	Base de datos.....	99
7	Implementación e implantación del sistema	106
7.1	Establecimiento del plan de implantación.....	106
7.2	Formación necesaria para la implantación y mantenimiento del sistema.....	107
7.3	Elaboración de manuales	107
7.4	Incorporación del sistema al entorno de operación.....	108
7.5	Carga de datos al entorno de operación	109
7.6	Preparación del mantenimiento del sistema	109
7.6.1	Entorno de pruebas.....	109
7.6.2	Entorno de pre-producción.....	110
7.6.3	Entorno de producción	110
7.7	Presentación y aprobación del sistema	112
7.8	Paso a producción	112
8	Evaluación.....	113
8.1	Plan de pruebas	113
8.1.1	Pruebas Unitarias	113
8.1.2	Pruebas de integración	123
8.1.3	Pruebas del sistema	128
8.1.4	Pruebas de implantación del sistema.....	128
8.1.5	Pruebas de aceptación del sistema.....	130
8.1.6	Pruebas de usabilidad del sistema	130
8.2	Test de satisfacción de usuarios	134
8.2.1	Satisfacción de voluntarios.....	134



8.2.2	Satisfacción de miembros de la Fundación.....	137
9	Conclusiones y líneas futuras.....	141
9.1	Conclusiones.....	141
9.1.1	Producto.....	141
9.1.2	Proceso.....	142
9.1.3	Personales.....	142
9.2	Líneas futuras.....	143
10	Development of an Application for Non-Governmental Organizations.....	145
10.1	Introduction.....	145
10.2	Objectives.....	145
10.3	Abstract.....	147
10.4	Conclusions and Future Work.....	153
10.4.1	Conclusions.....	153
10.4.2	Future Work.....	155
	Bibliografía.....	157
	Glosario de términos.....	159
	Anexo I: Manual de voluntario.....	161
	Anexo II: Manual de trabajador.....	163



Índice de tablas

Tabla 1: Gestor de contenidos.....	21
Tabla 2: Comparación de otras fundaciones.....	34
Tabla 3: Costes empresariales.....	39
Tabla 4: Costes de equipos.....	40
Tabla 5: Amortizaciones.....	40
Tabla 6: Material fungible.....	40
Tabla 7: Gastos de Personal.....	41
Tabla 8: Presupuesto.....	41
Tabla 9: Coste del proyecto.....	42
Tabla 10: Plantilla de Requisitos.....	44
Tabla 11: Requisito Funcional 001.....	46
Tabla 12: Requisito Funcional 002.....	46
Tabla 13: Requisito Funcional 003.....	47
Tabla 14: Requisito Funcional 004.....	47
Tabla 15: Requisito Funcional 005.....	47
Tabla 16: Requisito Funcional 006.....	47
Tabla 17: Requisito Funcional 007.....	47
Tabla 18: Requisito Funcional 008.....	48
Tabla 19: Requisito Funcional 009.....	48
Tabla 20: Requisito Funcional 010.....	48
Tabla 21: Requisito Funcional 011.....	48
Tabla 22: Requisito Funcional 012.....	48
Tabla 23: Requisito Funcional 013.....	49
Tabla 24: Requisito Funcional 014.....	49
Tabla 25: Requisito Funcional 015.....	49
Tabla 26: Requisito Funcional 016.....	49
Tabla 27: Requisito Funcional 017.....	49
Tabla 28: Requisito Funcional 018.....	50
Tabla 29: Requisito Funcional 019.....	50
Tabla 30: Requisito Funcional 020.....	50
Tabla 31: Requisito Funcional 021.....	50
Tabla 32: Requisito Funcional 022.....	50
Tabla 33: Requisito Funcional 023.....	51
Tabla 34: Requisito de Interfaz 001.....	51
Tabla 35: Requisito de Interfaz 002.....	51
Tabla 36: Requisito de Interfaz 003.....	51
Tabla 37: Requisito de Interfaz 004.....	51
Tabla 38: Requisito de Interfaz 005.....	52
Tabla 39: Requisito de Interfaz 006.....	52



Tabla 40: Requisito de Interfaz 007	52
Tabla 41: Requisito de Rendimiento 001.....	52
Tabla 42: Requisito de Rendimiento 002.....	52
Tabla 43: Requisito de Rendimiento 003.....	53
Tabla 44: Requisito de Rendimiento 004.....	53
Tabla 45: Requisito de Operación 001.....	53
Tabla 46: Requisito de Operación 002.....	53
Tabla 47: Requisito de Recursos 001.....	53
Tabla 48: Requisito de Comprobación 001.....	54
Tabla 49: Requisito de Comprobación 002.....	54
Tabla 50: Requisito de Comprobación 003.....	54
Tabla 51: Requisito de Comprobación 004.....	54
Tabla 52: Requisito de Comprobación 005.....	54
Tabla 53: Requisito de Aceptación de las pruebas 001.....	55
Tabla 54: Requisito de Aceptación de las pruebas 002.....	55
Tabla 55: Requisito de Aceptación de las pruebas 003.....	55
Tabla 56: Requisito de Aceptación de las pruebas 004.....	55
Tabla 57: Requisito de Aceptación de las pruebas 005.....	55
Tabla 58: Requisito de Daño 001.....	56
Tabla 59: Requisito de seguridad 001.....	56
Tabla 60: Requisito de seguridad 002.....	56
Tabla 61: Requisito de seguridad 003.....	56
Tabla 62: Requisito de Mantenimiento 001.....	57
Tabla 63: Plantilla casos de uso.....	57
Tabla 64: Caso de uso 01.....	59
Tabla 65: Caso de uso 02.....	60
Tabla 66: Caso de uso 03.....	61
Tabla 67: Caso de uso 04.....	61
Tabla 68: Caso de uso 05.....	62
Tabla 69: Caso de uso 06.....	63
Tabla 70: Caso de uso 07.....	63
Tabla 71: Caso de uso 08.....	64
Tabla 72: Caso de uso 09.....	65
Tabla 73: Caso de uso 10.....	66
Tabla 74: Excepción 01.....	69
Tabla 75: Excepción 02.....	69
Tabla 76: Excepción 03.....	69
Tabla 77: Encabezado.....	71
Tabla 78: Menú Navegación voluntarios.....	71
Tabla 79: Menú Navegación trabajadores.....	72
Tabla 80: Menú Navegación principal.....	73



Tabla 81: Pie de página	74
Tabla 82: Pantalla Login	74
Tabla 83: Pantalla Hazte voluntario	75
Tabla 84: Pantalla mi perfil.....	76
Tabla 85: Pantalla Modificar Perfil.....	77
Tabla 86: Pantalla Mis Tareas	78
Tabla 87: Pantalla Perfil Trabajador	79
Tabla 88: Pantalla Modificar Perfil trabajador.....	79
Tabla 89: Pantalla Añadir Trabajador	80
Tabla 90: Pantalla Asignación de Tareas.....	81
Tabla 91: Pantalla Voluntarios	82
Tabla 92: Pantalla Voluntarios Eliminados	82
Tabla 93: Pantalla Voluntarios Sin Validar.....	83
Tabla 94: Pantalla Validar Voluntario	84
Tabla 95: Pantalla Detalles Voluntario	85
Tabla 96: Pantalla Modificar Voluntario.....	87
Tabla 97: Pantalla Ver actividades de un Voluntario.....	87
Tabla 98: Pantalla Ver Actividades	88
Tabla 99: Pantalla Detalles actividad	89
Tabla 100: Pantalla Modificar Actividad.....	90
Tabla 101: Pantalla Asignar Actividad.....	91
Tabla 102: Pantalla Ver Voluntarios Asignados a una Actividad	91
Tabla 103: Pantalla Nueva Actividad	92
Tabla 104: Pantalla Mis Actividades Creadas	93
Tabla 105: Pantalla Actividades Terminadas	93
Tabla 106: Clase de acceso a la información.....	94
Tabla 107: Clase lógica de voluntarios	94
Tabla 108: Clase lógica de trabajadores	94
Tabla 109: Clase lógica de actividades	94
Tabla 110: Clase lógica de asignaciones.....	94
Tabla 111: Clase DataBase.....	97
Tabla 112: Clase Voluntaries.....	97
Tabla 113: Clase Activities	98
Tabla 114: Clase Workers.....	98
Tabla 115: Clase ActivitiesUser	98
Tabla 116: Tabla user.....	101
Tabla 117: Tabla uservalidate.....	102
Tabla 118: Tabla studies.....	103
Tabla 119: Tabla experience	103
Tabla 120: Tabla languages	103
Tabla 121: Tabla activities.....	105



Tabla 122: Tabla activities_user	105
Tabla 123: Tabla workers	105
Tabla 124: Prueba unitaria 001	114
Tabla 125: Prueba unitaria 002	114
Tabla 126: Prueba unitaria 003	115
Tabla 127: Prueba unitaria 004	115
Tabla 128: Prueba unitaria 005	115
Tabla 129: Prueba unitaria 006	115
Tabla 130: Prueba unitaria 007	116
Tabla 131: Prueba unitaria 008	116
Tabla 132: Prueba unitaria 009	116
Tabla 133: Prueba unitaria 010	116
Tabla 134: Prueba unitaria 011	117
Tabla 135: Prueba unitaria 012	117
Tabla 136: Prueba unitaria 013	117
Tabla 137: Prueba unitaria 014	117
Tabla 138: Prueba unitaria 015	118
Tabla 139: Prueba unitaria 016	118
Tabla 140: Prueba unitaria 017	118
Tabla 141: Prueba unitaria 018	118
Tabla 142: Prueba unitaria 019	119
Tabla 143: Prueba unitaria 020	119
Tabla 144: Prueba unitaria 021	119
Tabla 145: Prueba unitaria 022	119
Tabla 146: Prueba unitaria 023	120
Tabla 147: Prueba unitaria 024	120
Tabla 148: Prueba unitaria 025	120
Tabla 149: Prueba unitaria 026	120
Tabla 150: Prueba unitaria 027	121
Tabla 151: Prueba unitaria 028	121
Tabla 152: Prueba unitaria 029	121
Tabla 153: Prueba unitaria 030	121
Tabla 154: Prueba unitaria 031	122
Tabla 155: Prueba unitaria 032	122
Tabla 156: Prueba unitaria 033	122
Tabla 157: Prueba unitaria 034	122
Tabla 158: Prueba de integración 001	123
Tabla 159: Prueba de integración 002	124
Tabla 160: Prueba de integración 003	124
Tabla 161: Prueba de integración 004	124
Tabla 162: Prueba de integración 005	125



Tabla 163: Prueba de integración 006.....	125
Tabla 164: Prueba de integración 007.....	125
Tabla 165: Prueba de integración 008.....	126
Tabla 166: Prueba de integración 009.....	126
Tabla 167: Prueba de integración 010.....	126
Tabla 168: Prueba de integración 011.....	127
Tabla 169: Prueba de integración 012.....	127
Tabla 170: Prueba de integración 013.....	127
Tabla 171: Prueba de integración 014.....	128
Tabla 172: Prueba de integración 015.....	128
Tabla 173: Prueba de implantación del sistema 001.....	129
Tabla 174: Prueba de implantación del sistema 002.....	129
Tabla 175: Prueba de implantación del sistema 003.....	130
Tabla 176: Prueba de implantación del sistema 004.....	130
Tabla 177: Test usuario 01.....	135
Tabla 178: Test usuario 02.....	135
Tabla 179: Test usuario 03.....	135
Tabla 180: Test usuario 04.....	136
Tabla 181: Test usuario 05.....	136
Tabla 182: Test usuario 06.....	136
Tabla 183: Test usuario 07.....	136
Tabla 184: Test usuario 08.....	137
Tabla 185: Test usuario 09.....	137
Tabla 186: Test usuario 10.....	137
Tabla 187: Test trabajador 01.....	139
Tabla 188: Test trabajador 02.....	139
Tabla 189: Test trabajador 03.....	140
Tabla 190: Test trabajador 04.....	140
Table 191: Project Budget	149

Índice de ilustraciones

Ilustración 1: GANTT	38
Ilustración 2: Diagrama casos de uso.....	58
Ilustración 3: Diagrama de secuencia CU_01	59
Ilustración 4: Diagrama de secuencia CU_02	60
Ilustración 5: Diagrama de secuencia CU_03	61
Ilustración 6: Diagrama de secuencia CU_04	62
Ilustración 7: Diagrama de secuencia CU_05	62
Ilustración 8: Diagrama de secuencia CU_06	63
Ilustración 9: Diagrama de secuencia CU_07	64
Ilustración 10: Diagrama de secuencia CU_08.....	65
Ilustración 11: Diagrama de secuencia CU_09.....	66
Ilustración 12: Diagrama de secuencia CU_10.....	66
Ilustración 13: Modelo vista controlador.....	68
Ilustración 14: Encabezado sin inicio sesión.....	70
Ilustración 15: Encabezado con inicio de sesión.....	70
Ilustración 16: Menú navegación voluntarios	71
Ilustración 17: Menú navegación trabajadores	72
Ilustración 18: Menú navegación principal	72
Ilustración 19: Pie de página	73
Ilustración 20: Pantalla Login	74
Ilustración 21: Pantalla Hazte voluntario	75
Ilustración 22: Pantalla mi Perfil	76
Ilustración 23: Pantalla Modificar Perfil.....	77
Ilustración 24: Pantalla Mis Tareas	78
Ilustración 25: Pantalla Perfil Trabajador	78
Ilustración 26: Pantalla Modificar Perfil trabajador.....	79
Ilustración 27: Pantalla Añadir Trabajador	80
Ilustración 28: Pantalla Asignación de Tareas.....	80
Ilustración 29: Pantalla Ver Voluntarios	81
Ilustración 30: Pantalla Voluntarios Eliminados	82
Ilustración 31: Pantalla Voluntarios Sin Validar.....	83
Ilustración 32: Pantalla Validar Voluntario	84
Ilustración 33: Pantalla Detalles Voluntario	85
Ilustración 34: Pantalla Modificar voluntario.....	86
Ilustración 35: Ver actividades de un Voluntario	87
Ilustración 36: Pantalla Ver Actividades	88
Ilustración 37: Pantalla Detalles de Actividad	89
Ilustración 38: Pantalla Modificar Actividad	90
Ilustración 39: Pantalla Asignar Actividad	91



Ilustración 40: Pantalla Ver voluntarios Asignados a una Actividad	91
Ilustración 41: Pantalla Nueva Actividad.....	92
Ilustración 42: Pantalla Mis Actividades Creadas.....	92
Ilustración 43: Pantalla Actividades Terminadas.....	93
Ilustración 44: Base de datos.....	99
Ilustración 45: Flujo de control	111
Ilustración 46: GANTT estimado-real	141
Illustration 47: GANT	149
Illustration 48: Model View Controller.....	150
Illustration 49: Control Flow.....	152
Ilustración 50: GANTT estimated-real	154

1 Introducción

Este primer apartado está dividido en tres secciones. En la primera sección se presentan los motivos por los que se decidió llevar a cabo el trabajo. En la segunda sección se anuncian los objetivos y por último se explica la estructura del presente documento.

1.1 Motivación

La motivación de este TFG viene dada por la necesidad que tiene la Fundación Madrina de actualizar su portal web, el cual es difícil de manejar y ha quedado obsoleto. En su estado actual, el sitio web es poco intuitivo y difícil de manejar por los diferentes usuarios que acceden a ella.

El principal problema viene dado por la falta de personal especializado en informática entre los miembros de la fundación que se encargan de mantener el portal. A pesar de la falta de estos conocimientos, intentan mantener activo un sitio web bastante obsoleto.

Dicho esto, necesitan de ayuda de una persona con conocimientos informáticos para poder desarrollar una nueva aplicación web, que sea accesible, moderna, que incorpore nuevas funcionalidades como la gestión de voluntarios, actividades y trabajadores de la fundación. Toda la información ha de almacenarse de forma segura, ya que es un sitio con información sensible, sin embargo el sitio actual aparte de obsoleto no cumple con las medidas mínimas de seguridad, por lo que pone en serio peligro la actividad de la Fundación. Gracias a la nueva aplicación conseguirán un trabajo más cómodo y eficiente.

“Desarrollo de una aplicación web para proyectos no gubernamentales” es un proyecto altamente motivador para un Trabajo fin de Grado, dada la naturaleza de la organización, necesidad de adaptación a unas tecnologías que quedan fuera del alcance de los recursos de la organización y la integración en un entorno intuitivo y sencillo.

1.2 Objetivos

El objetivo de este proyecto es realizar una aplicación web para la Fundación Madrina, como el actual no es accesible, el nuevo debe cumplir los estándares de conformidad siendo accesible a todo tipo de usuario, y facilitando la navegación por la web.

A su vez se propone desarrollar una BBDD que se gestione desde la aplicación, en la que una persona interesada pueda inscribirse como voluntario, completando un formulario de registro a través de la web. Una vez registrado podrá acceder a su perfil en todo momento. Además todos los trabajadores de la fundación tendrán una cuenta y un perfil de usuario en el que podrán gestionar a los voluntarios, las actividades a desarrollar en la fundación, así como asignar las distintas actividades a los distintos voluntarios.

Esta base de datos se va a integrar con la base de datos de un proyecto que están realizando alumnos de la Universidad Europea, en la que se gestiona los afectados y el almacén de productos. Nuestra BBDD va a gestionar la parte de voluntariado, actividades y trabajadores de la fundación, y en un futuro los alumnos de la Universidad Europea añadirán las funcionalidades mencionadas anteriormente. De esta manera se va a realizar una BBDD que sea multiplataforma, de modo que si en un momento se quiere realizar un cambio en el sistema operativo, no sea necesario migrar la BBDD a otro sistema, lo cual no es nada eficiente para una ONG.

La elección de las distintas herramientas a lo largo del proyecto corresponde a cada una de las necesidades de la fundación y por lo tanto están orientadas a una mayor facilidad a la hora de actualizar los contenidos, facilitar el trabajo de voluntariado, gestionar las afectadas, y el manejo del portal web. Esta tarea no es sencilla puesto que hay que satisfacer las necesidades de la fundación y que sea fácil de usar. La elección de los diferentes componentes que conforman el nuevo portal web responden las expectativas de la fundación de realizar uno nuevo con un esquema similar al antiguo pero más claro al usuario, ya que si se cambia al cien por cien se dificulta el acceso al usuario, a su vez, añadiendo nuevas funcionalidades como el registro y acceso de voluntarios.

El apartado de registro y acceso de usuarios será completamente nuevo en el portal, puesto que de esta manera el usuario que quiera ser voluntario solo tendrá que registrarse por la web facilitando el trabajo a cualquier miembro de la fundación.

1.3 Estructura del documento

El presente documento se ha estructurado en distintos apartados. A fin de ofrecer una visión global del documento, a continuación se enuncia el contenido de cada uno de estos capítulos:

- En el apartado 1 se relata cuáles han sido las motivaciones por las que se decidió realizar este trabajo, los objetivos y la estructura que sigue el presente documento.
- En el apartado 2 se realiza un estudio de la base teórica relacionada con el trabajo desarrollado.
- En el apartado 3 se describe el marco de trabajo, en el que se incluye metodología a utilizar, planificación y presupuesto.
- En el apartado 4 se muestra el impacto ético, social y económico del proyecto.
- En el apartado 5 se describe el alcance del trabajo explicando la fase de análisis que contiene situación de partida, análisis de requisitos y análisis de casos de uso.
- En el apartado 6 se describe la fase de diseño, en el que se explica el diseño de la arquitectura del sistema, diseño de la taxonomía del sistema, diseño de clases y diseño de la base de datos.
- En el apartado 7 se describe la fase de implementación e implantación. En este apartado encontraremos la información sobre el proceso de instalación y estableceremos un plan de pruebas que garantice el correcto funcionamiento y calidad del proyecto.
- En el apartado 8 se describe la fase de evaluación, en el que se define la batería de pruebas que debe pasar para alcanzar la primera versión del sistema.
- En el apartado 9 se describe la fase de conclusiones y líneas futuras, se explican las impresiones una vez terminado el desarrollo, y se proponen posibles ideas que podrían incluirse en el proyecto en un futuro.

En este documento también se encuentran las referencias bibliográficas usadas para el proyecto y un glosario de términos. En los anexos encontrará el manual de usuario y el manual de trabajadores utilizados para formar al personal que va a utilizar la aplicación.

[18]



2 Estado del arte

2.1 Gestor de contenidos

Analizando las diferencias de utilizar software libre o propietario podemos destacar:

- **Software libre:**
 - Costes iniciales reducidos: el bajo o nulo coste de los productos iniciales permiten proporcionar a las empresas servicios para ampliar sus infraestructuras sin que se vean afectados económicamente. A su vez una empresa que cuenta de personal formado podrá reducir los costes considerablemente.
 - Libertad de uso y redistribución: una licencia libre permite la instalación del software tantas veces como el usuario desee.
 - Independencia tecnológica: el acceso al código fuente permite el desarrollo de nuevos productos sin la necesidad de desarrollar todo el proceso de cero. El secretismo tecnológico es uno de los grandes frenos y desequilibrios existentes para el desarrollo en el modelo de propiedad intelectual.
 - Fomento de la libre competencia al basarse en servicios y no licencias: uno de los modelos de negocio que genera el software libre es la contratación de servicios de atención al cliente. Este sistema permite que las compañías que den el servicio compitan en igualdad de condiciones al no poseer la propiedad del producto del cual dan servicio.
 - Soporte y compatibilidad a largo plazo: es una gran ventaja en software libre ya que el propietario no puede dejar de dar soporte o de dar mejoras para sacar beneficios económicos de los clientes. Las modificaciones que se hagan será beneficiosas al usuario, puesto que si no, no se realizarían.
 - Formatos estándar: los formatos estándar permiten una interoperabilidad más alta entre sistemas, evitando incompatibilidades. Los estándares de facto son válidos en ocasiones para lograr una alta interoperabilidad si se omite el hecho que estos exigen el pago de royalties a terceros y por razones de mercado expuestas en el anterior punto no interesa que se perpetúen mucho tiempo.

- Sistemas sin puertas traseras y más seguros: el acceso al código fuente permite que tanto hackers como empresas de seguridad de todo el mundo puedan auditar los programas, por lo que la existencia de puertas traseras es ilógica ya que se pondría en evidencia y contraviene el interés de la comunidad que es la que lo genera.
- Corrección más rápida y eficiencia de fallos: el funcionamiento e interés conjunto de la comunidad ha demostrado solucionar más rápidamente los fallos de seguridad en el software libre, algo que desgraciadamente en el software propietario es más difícil y costoso.
- Métodos simples y unificados de gestión de software: actualmente la mayoría de distribuciones de Linux incorporan alguno de los sistemas que unifican el método de instalación de programas, librerías. por parte de los usuarios. Esto llega a simplificar hasta el grado de marcar o desmarcar una casilla para la gestión del software. A su vez permiten el acceso a las miles de aplicaciones existentes de forma segura y gratuita. A la par que evitan tener que recurrir a páginas web de dudosa ética desde las que los usuarios instalan sin saberlo spyware o virus informáticos en sus sistemas.
- Sistema de expansión: las ventajas especialmente económicas que aportan las soluciones libres a muchas empresas y las aportaciones de la comunidad han permitido un constante crecimiento del software libre, hasta superar en ocasiones como en el de los servidores web, al mercado propietario.
- **Software propietario:**
 - Propiedad y decisión de uso del software por parte de la empresa: el desarrollo de la mayoría de software requiere importantes inversiones para su estudio y desarrollo. Este esfuerzo, de no ser protegido se haría en balde, puesto que la competencia se podría apropiarse inmediatamente del producto una vez finalizado, para sus propios fines. Esto garantiza al productor ser compensado por la inversión, fomentando así el continuo desarrollo.
 - Soporte para todo tipo de hardware: el software privado tiene un mayor uso que el libre, por ello tiene un soporte mayor en aplicaciones y compatibilidades con el hardware.



- Mejor acabado de la mayoría de aplicaciones: el desarrollador de software propietario, generalmente, da un mejor acabado a las aplicaciones en cuestiones, tanto de estética, como de usabilidad de la aplicación.
- Las aplicaciones número uno son propietarias: el actual dominio de mercado no sólo interesa a los fabricantes de hardware, sino que también a los de software.
- El ocio para ordenadores personales está destinado al mercado propietario: los desarrolladores de juegos tienen sus miras en el mercado más lucrativo, que es el de las consolas y en última instancia el de los ordenadores personales.
- Menor necesidad de técnicos especializados: el mejor acabado de la mayoría de sistemas de software propietario y los estándares de facto actuales permiten una simplificación del tratamiento de dichos sistemas, dando acceso a su uso y administración, a técnicos que requieren una menor formación, reduciendo costes de mantenimiento.
- Mayor mercado laboral actual: como muchas de las ventajas restantes se deben al dominio del mercado, no hay que ignorar que cualquier trabajo relacionado con la informática pasará, en casi todos los casos, por conocer herramientas de software propietario.
- Mejor protección de las obras con copyright: las obras protegidas por copyright se ven beneficiadas por mecanismos anti-copia, como el DRM y otras medidas, que palían o dificultan en cierto grado la piratería.
- Unificación de productos: una de las ventajas más destacables del software propietario es la toma de decisiones centralizada que se hace en torno a una línea de productos, haciendo que no se desvíe de la idea principal y generando productos funcionales y altamente compatibles.

[1]

Una vez analizadas las características de software libre o propietario, el proyecto se realizará con software libre puesto que es un trabajo fin de grado para una fundación, el presupuesto es mínimo, por no decir nulo. De esta manera se decide utilizar gestor de contenidos para facilitar la tarea de administración y gestión de la plataforma. La utilización de un gestor de contenidos lo solicita la fundación como un requisito

indispensable para facilitarles a ellos el manejo y modificación de la plataforma en un futuro, ya que carecen de conocimientos informáticos avanzados.

Dentro de los gestores de contenidos de software libre podemos destacar los tres más populares, que son WordPress, Joomla y Drupal.

Se procedió a estudiar las características de los distintos gestores de contenidos, para escoger el que mejor características tuviera para nuestro proyecto:

	WordPress	Joomla	Drupal
Lenguaje	PHP	PHP	PHP
Opción gratis	Sí	Sí	Sí
¿Código abierto?	Sí	Sí	Sí
Cuota de mercado actual	46%	12%	3%
Dificultad para programar	Baja	Alta	Media
Flexibilidad en la implementación	Alta	Baja	Muy alta
Dificultad para la administración	Baja	Media	Alta
Dificultad en la implementación del diseño	Baja	Alta	Media
Dificultad en actualización	Baja	Alta	Alta
SEO recién instalado el CMS	Excelente	Medio/Malo	Malo
Calidad de la documentación	Excelente	Inexistente	Alta
Calidad de la comunidad de usuarios	Alta	Baja	Alta
Tráfico soportado	Bajo	Medio	Alto
Fecha de lanzamiento	27 de Mayo de 2003	17 de Agosto de 2005	1 de Enero de 2001

Tabla 1: Gestor de contenidos

*Los datos de la tabla no son propios

- **WordPress** es un sistema de gestión de contenido CMS enfocado a la creación de blogs. Está desarrollado en PHP para entornos que ejecuten MySQL y Apache, bajo licencia y código modificable. Es el CMS más conocido en la actualidad, y que más se utiliza.



- **Joomla** es un sistema de gestión de contenido CMS que permite desarrollar sitios web dinámicos e interactivos. Permite crear y modificar un sitio web de manera sencilla a través de su panel de administración. Como hemos comentado antes es de código abierto y está desarrollado en PHP al igual que WordPress.
- **Drupal** es un sistema de gestión de contenidos CMS libre, modular, y muy fácil de configurar que permite la publicación de artículos, imágenes entre otros tipos de archivos. Es un sistema dinámico, ya que almacena los contenidos y modificaciones en una base de datos, en lugar de un sistema de ficheros.

La conclusión a la que se llegó es que Joomla es muy buena opción, ya que es más eficiente que WordPress, y su funcionamiento es más sencillo que Drupal, no obstante se decidió realizarlo en WordPress ya que los miembros de la fundación lo solicitaron como requisito en la propuesta del proyecto.

[2], [3], [4], [5], [6], [14]

2.2 Base de Datos

Una base de datos es una colección de información organizada de forma que un programa pueda encontrar y utilizar fácilmente. Pueden contener distintos tipos de datos y se clasifican de diferentes formas dependiendo de su contenido o manera de almacenamiento.

2.2.1 Tipos de bases de datos

Podemos diferenciar distintos tipos de bases de datos:

- **Bases de datos jerárquicas:** en una base de datos jerárquica se organizan los datos utilizando estructuras en forma de árbol. Un árbol es una estructura jerárquica en la que los elementos se suelen denominar nodos y existen dependencias entre ellos. La dependencia es de 1:M, lo que implica que un hijo no pueda tener nunca varios padres, pero un padre sí puede tener varios hijos. Es muy útil cuando se maneja gran cantidad de información, ya que permite crear estructuras estables con gran rendimiento. Su principal inconveniente es que es poco eficiente cuando hay redundancia entre datos.

- **Bases de datos en red:** en una base de datos en red se utiliza la estructura de grafo/red, los datos están relacionados mediante relaciones del tipo 1:M, pero en este caso, a diferencia de las bases de datos jerárquicas, un hijo puede estar relacionado con varios padres. Se soluciona el problema de la redundancia entre datos, pero dificulta la administración de la base de datos.
- **Base de datos relacional:** consiste en organizar los datos en forma de tablas, las relaciones entre los objetos se consiguen incluyendo en la tabla del hijo, la clave del objeto padre.
- **Base de datos orientada a objetos:** es un modelo más reciente, trata de almacenar en la base de datos los objetos complejos. La información que contienen se organiza en atributos y el comportamiento en operaciones.
- **Base de datos deductivas:** se basa en reglas lógicas, de forma que con la información introducida por el usuario y siguiendo las reglas lógicas podemos extraer nueva información.
- **Base de datos distribuidas:** como indica su propio nombre su información está distribuida entre distintas ubicaciones, puede ser accesible desde cualquier punto de la red como si la información estuviera centralizada. Muchas de las bases de datos NoSQL siguen esta arquitectura.
- **Base de datos multidimensionales:** en una base de datos multidimensional los datos se almacenan en tablas de múltiples dimensiones en vez de tablas bidimensionales como las del modelo relacional. Se utiliza para grandes volúmenes de información.

2.2.2 Las bases de datos a día de hoy

Hoy en día se utilizan distintos tipos de bases de datos, entre ellas podemos destacar las cinco más usadas, clasificadas de mayor a menor número de usuarios:

- **Oracle**, clasifica sus bases de datos en dos tipos:
 - Procesado transaccional en línea: base de datos orientada a gran número de transacciones en poco tiempo. La respuesta es rápida y con integridad de los datos a la hora de realizar inserciones,



actualizaciones o borrados de registros. Suele utilizar un modelo relacional evitando redundancias e inconsistencias. Es el más utilizado hoy en día.

- Procesado analítico en línea: es orientado a análisis de datos. El volumen de transacciones es pequeña, pero de gran volumen de datos. Suele utilizar modelos multidimensionales desnormalizados, esto implica el consumo de mucho espacio, no obstante, se podrían utilizar modelos relacionales.
- **SQL Server de Microsoft**: permite crear aplicaciones críticas y soluciones Big Data mediante tecnología en memoria y de alto rendimiento.
- **MySQL**: es el Sistema Gestor de Base de Datos de código abierto más popular en internet. Está diseñado para entornos de producción con mucha carga de trabajo, así como para integrarse en software para ser distribuido. Al ser un servidor de base de datos relacionales, MySQL se convierte en una herramienta veloz en la accesibilidad a los datos introducidos en las distintas tablas independientes que forman las bases de datos de este lenguaje.
- **MongoDB**: es un sistema de base de datos NoSQL orientado a documentos desarrollados bajo el concepto de código abierto. MongoDB forma parte de la nueva familia de sistemas de bases de datos NoSQL, que en vez de guardar datos en tablas como las bases de datos relacionales, guarda estructuras de datos en documentos tipo JSON con un esquema dinámico, haciendo que la integración de los datos en algunas aplicaciones sea más fácil y rápido.
- **PostgreSQL**: es un sistema de base de datos relacional orientado a objetos. Es de código libre, por lo que el desarrollo no es manejado por una persona o empresa, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada.

Otro de los más usados y conocidos es Hadoop sistema de base de datos NoSQL más potente hoy en día. Se utiliza para procesamiento y análisis de datos. Su

principal diferencia con MongoDB es que no está orientado a documentos sino a columnas.

En nuestro proyecto vamos a realizar una base de datos MySQL puesto que WordPress es el sistema de bases de datos que soporta.

[11]

2.2.3 Ventajas e inconvenientes de los sistemas de bases de datos

Los sistemas de bases de datos presentan muchas ventajas, que se pueden clasificar en dos grandes grupos:

- **Ventajas por la integración de datos:**
 - Control sobre la redundancia de datos: los sistemas de ficheros almacenan varias copias de los mismos datos en distintos ficheros, por lo tanto se desperdicia gran cantidad de espacio y provoca falta de consistencia de datos. Los sistemas de bases de datos incluyen estos ficheros, por lo que no almacena los datos varias veces. La redundancia es imposible de eliminar al completo, pero si se reduce notablemente, esto se debe a que en ocasiones es necesario modelar relaciones entre los datos para mejorar las prestaciones.
 - Más información sobre la misma cantidad de datos: al estar todos los datos integrados, se puede extraer información sobre los mismos.
 - Compartición de datos: en los sistemas de ficheros, cada parte corresponde a un usuario, en cambio en un sistema de base de datos los datos son compartidos entre todos, por ejemplo en una empresa todos los trabajadores tienen acceso a ella independientemente del departamento al que pertenezcan.
 - Mantenimiento de estándares: gracias a la integración es más fácil respetar los estándares establecidos.
- **Ventajas por la existencia del Sistema Gestor de Base de Datos (SGBD):**
 - Mejora la integridad de los datos: la integridad se refiere a la validez y consistencia de los datos almacenados, se expresa mediante reglas



que se tienen que cumplir. El SGBD tiene que cumplir todas estas reglas.

- Mejora la seguridad: se tienen medidas de seguridad para evitar el acceso a usuarios no autorizados. A diferencia de los sistemas de ficheros no vulnerables, los sistemas de bases de datos permiten mantener la seguridad mediante el establecimiento de claves para identificar el acceso autorizado.
- Mejora la accesibilidad a los datos: muchos SGBD incluyen lenguajes de consultas o generadores de informes que permiten al usuario realizar cualquier tipo de consulta, sin necesidad de que un programador haya escrito estas tareas.
- Mejora la productividad: gracias a los sistemas de bases de datos, el programador no tiene que estar atento de las funciones estándar, y permite que se centre en una tarea específica requerida por el usuario, sin preocuparse de los detalles a bajo nivel.
- Mejora el mantenimiento gracias a la independencia de datos: en los sistemas de ficheros los datos están incluidos en los programas de aplicación que los maneja. Esto produce una dependencia entre los datos y los programas, de modo que cualquier cambio en los datos produce una alteración del programa. Sin embargo, los sistemas de bases de datos tienen independencia de datos, lo que simplifica el mantenimiento de las aplicaciones a la base de datos.
- Aumento de la concurrencia: en los sistemas de ficheros si varios usuarios acceden a un mismo fichero simultáneamente, puede perderse la integridad. Sin embargo, los sistemas de bases de datos gestionan el acceso concurrente a la base de datos.
- Mejora en los servicios copias de seguridad y de recuperación ante fallos: muchos de los sistemas de ficheros dejan que sea el usuario el que se encargue de realizar copias de seguridad, de manera que cuando ocurra un error, el usuario pueda restaurar la base de datos, perdiendo todo el trabajo que se hizo desde la última copia. Sin

embargo, los sistemas de bases de datos funcionan de modo que se minimiza la cantidad de trabajo perdido cuando se produce un fallo.

Aunque presenta muchas ventajas, también podemos encontrar algunos inconvenientes en los sistemas de bases de datos:

- **Complejidad:** los sistemas de bases de datos son conjuntos de programas muy complejos y con gran funcionalidad, por lo que es preciso comprender muy bien la funcionalidad para sacarle el máximo partido.
- **Tamaño:** los sistemas de bases de datos son muy extensos y requieren gran cantidad de espacio en disco y memoria para que sean eficientes.
- **Coste económico:** el coste varía dependiendo de la funcionalidad que ofrece.
- **Coste del equipamiento adicional:** tanto el sistema de base de datos, como la propia base de datos puede que necesite ampliar su memoria, esto puede llevar varias consecuencias, como por ejemplo, que sea necesario ampliar la potencia de la maquina a una superior.
- **Prestaciones:** un sistema de ficheros está escrito para una aplicación, por lo que las prestaciones que ofrece son muy buenas. Sin embargo, los sistemas de bases de datos están escritos para ser más generales y ser útiles en diferentes aplicaciones, lo que puede hacer que muchas de ellas no sean igual de rápidas que antes.
- **Vulnerable a los fallos:** al estar todo centralizado en el sistema de bases de datos, el sistema es más vulnerable a fallos.

2.2.4 Tipos de usuarios de las bases de datos

Podemos definir a los usuarios como toda persona que tenga todo tipo de contacto con el sistema de la base de datos desde que se diseña, elabora, termina y se usa.

Tenemos cuatro tipos de usuarios principales:

- **Analista y programador de aplicaciones:** los analistas determinan los requerimientos de los usuarios finales y desarrollan especificaciones para transacciones programadas que satisfagan estos requerimientos. Los



programadores implementan estas especificaciones en forma de programas y luego realizan pruebas, y por último lo documentan.

- **Administradores:** persona que se encarga de mantener el sistema de base de datos, autoriza el acceso a la base de datos, coordina y vigila al resto de usuarios. Tiene la obligación de estar pendiente de que este operativo siempre el sistema.
- **Diseñadores de base de datos:** identifican los datos que se almacenaran en la base de datos, así como las estructuras para representar y almacenar los datos.
- **Usuarios finales:** persona que tiene acceso para consultar, actualizar la base de datos. Las consultas las realiza en un lenguaje de alto nivel, mediante la interfaz que ha desarrollado el programador, ya que el usuario no tiene por qué tener conocimientos sobre base de datos.

[11], [12], [13]

2.3 Lenguajes a utilizar

En este apartado se va a explicar en detalle el sistema gestor de base de datos que se va a utilizar en el proyecto, así como el lenguaje de programación que se va a implementar.

2.3.1 MySQL

MySQL es el Sistema Gestor de Base de Datos de código abierto, uno de los más populares hoy en día. Como base de datos relacional utiliza múltiples tablas para almacenar y organizar la información. Fue escrito en C y C++, y destaca la gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados PHP, Perl y Java.

2.3.1.1 Ventajas

Podemos destacar varias ventajas que ofrece MySQL:

- Tiene software *Open Source*.
- Es bastante rápido al realizar las operaciones, lo que lo hace uno de los gestores con mejor rendimiento.

- Bajo coste en la elaboración de base de datos, ya que su bajo consumo permite que se ejecute en cualquier máquina.
- Facilidad de configuración e instalación. Se puede instalar en cualquier sistema operativo.
- Baja probabilidad de corromper datos.
- Su conexión, velocidad y seguridad.
- Usa licencia GPL.

2.3.1.2 Características principales

Está desarrollado en C y C++ y se distribuyen ejecutables para cerca de diecinueve plataformas diferentes. La API se encuentra disponible en C, C++, Eiffel, Java, Perl, PHP, Python, Ruby y TCL.

MySQL está optimizado para equipos de múltiples procesadores, y destaca su velocidad de respuesta, puesto que es bastante elevada. Se puede utilizar como cliente-servidor o incrustado en aplicaciones. Tiene una amplia gama de tipos de datos y soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes para optimizar cada sistema de bases de datos.

Permite conectividad TCP/IP, sockets UNIX y sockets NT, además de soportar completamente ODBC.

[13]

2.3.2 PHP

PHP es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

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



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

2.3.2.1 Ventajas

Las principales razones de utilizar este lenguaje se debe a su poder y sencillez. Podemos destacar tres razones:

- **Coste:** es gratuito, y con gran variedad de documentación en internet.
- **Sencillez y versatilidad:** es muy sencillo y fácil de aprender, además posee gran variedad de funciones que pueden ser utilizadas para mejorar el rendimiento de los programas.
- **Seguridad:** debido al gran uso de este lenguaje en las aplicaciones Web, y al ser gratuito la comunidad de programadores ofrece mejoras de continuo, por lo que la seguridad va aumentando. Cada versión de PHP es más segura.

2.3.2.2 Características

PHP es un lenguaje que soporta diversas bases de datos: Oracle, MySQL, MS SQL Server, PostgreSQL. etc. Dispone de varias bibliotecas externas, que permite generar documentos hasta analizar el propio código XML.

Es un lenguaje fácil de mantener y tener actualizado, ya que ofrece una solución simple para el desarrollo de páginas web. Al ser usado por tantos programadores y ser de código libre, los errores se solucionan rápidamente, ya que cualquiera puede solucionarlo y ponerlo al alcance de todos.

En las últimas versiones PHP ofrece soporte a Orientación a Objetos, lo que lo hace aún más completo.

[7]

2.3.3 HTML

HTML es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código para la definición de contenido de la página web, como imágenes, texto, etc.

2.3.3.1 Ventajas

Las principales ventajas de utilizar HTML son:

- Es nativo y de código libre.
- Tiene etiquetas que permiten clasificar y ordenar en distintos niveles y estructuras el contenido. Además contiene metadatos de manera más formal, favoreciendo el posicionamiento SEO y la accesibilidad.
- El código es más simple, permite hacer páginas web más ligeras.
- Es compatible con gran variedad de navegadores.
- Tiene la etiqueta de dibujo *canvas*, la cual permite ofrecer mejores efectos visuales.
- Ofrece soporte a *codecs* específicos.
- Permite la inserción de audio y video directamente sobre el código.
- Permite la geolocalización del usuario. Hoy en día muy útil en dispositivos portátiles.
- Permite ejecutar páginas sin estar conectado.
- Incorpora nuevas capacidades JavaScript que aumentan la capacidad de almacenamiento.

[8]

2.3.4 CSS

CSS es la tecnología desarrollada por el *World Wide Web Consortium (W3S)*.

Fue creada en 1996 como estándar para las páginas web, el objetivo era solucionar un problema, puesto que los navegadores existentes en aquel entonces estaban añadiendo etiquetas a la especificación de HTML, por lo que dificultaba el entendimiento de una web en otros navegadores. Aunque ha tardado en ser soportada correctamente por muchos navegadores, hoy en día es utilizada por la mayoría de ellos.



2.3.4.1 Ventajas

Las principales ventajas de usar CSS son:

- **Separación del contenido y presentación:** las hojas de estilo suelen estar separadas del código principal, de esta manera permite realizar tareas independientes, pero de forma paralela.
- **Dar estilo:** CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación.
- **Características enriquecidas:** utilizando un estilo declarativo simple, los diseñadores pueden establecer el posicionamiento, márgenes y alineación, colores, estilo de texto, listas numeradas entre otras.
- **Reutilización:** las hojas de estilo pueden compartirse en múltiples páginas, haciendo más fácil la utilización de un sitio web completo, simplemente cambiando la línea del CSS.
- **Flexibilidad:** podemos cambiar en cualquier momento alguna parte o la totalidad del diseño de nuestra página, tan solo con modificar la hoja de estilo, sin modificar el contenido.
- **Unificación del diseño de páginas de estilo:** un sitio web está formado por diferentes páginas, si enlazamos las hojas de estilo, mantenemos la misma apariencia en cada una de las páginas.
- **Optimizador de los tiempos de carga y de tráfico en el servidor:** al dividir contenido y apariencia, mejoramos los tiempos de carga, y reducimos el volumen de tráfico en nuestro servidor.
- **Precisión o elasticidad:** desde el momento en que usemos CSS, el tamaño y posicionamiento de los elementos que formen nuestras páginas podrá ser exacto. Podremos indicarle al navegador en qué píxel debe colocar una imagen, o qué alto y ancho deberá mostrar.
- **Accesibilidad y estructuración:** la separación entre etiquetado y estilo permiten a los agentes de accesibilidad combinar información de acuerdo a las necesidades de los usuarios con discapacidad.
- **Limpieza del código fuente:** si escribimos el estilo en una hoja independiente, resultará más fácil realizar cualquier tarea o búsqueda.

[9], [10]

2.4 Análisis de sitios similares

En este apartado vamos a analizar sitios web de otras ONG, para poder observar cómo son sus plataformas, servicios que ofrecen, lenguaje de programación en el que están implementadas, etc. Se analizarán seis sitios web, entre los que destacamos dos muy grandes, y otras cuatro fundaciones bastante conocidas pero de un tamaño similar a la nuestra:

- **Fundación Aladina** (<http://www.aladina.org/>): Fundación Aladina, es una entidad privada sin ánimo de lucro creada en 2005 con el objetivo de ayudar a niños y adolescentes enfermos de cáncer y a sus familias, proporcionándoles apoyo integral incluyendo apoyo material, psicológico y emocional. Esta plataforma web contiene distintos apartados con información de sus actividades y los distintos temas que tratan. Contiene un apartado de voluntariado bastante escueto, ya que solo proporciona información de que realizan los voluntarios. No hay ningún lugar donde registrarse como voluntario.
- **Fundación Tomillo** (<http://www.tomillo.org/>): se encarga del trabajo para el desarrollo de la persona y para la integración social de los individuos y colectivos más vulnerables a través de la educación, la mejora de la empleabilidad, el fortalecimiento de redes sociales, el análisis y la innovación. Su plataforma Web ofrece distintos apartados sobre su origen, donde se encuentra, donaciones, y voluntariado, todas ellas con gran cantidad de texto y poco intuitivo.
- **Fundación Theodora** (<http://es.theodora.org/>): Los Doctores Sonrisa de la Fundación Theodora son artistas profesionales de ámbitos diferentes (payasos, magos, cuentacuentos, músicos, actores...) que adaptan sus conocimientos artísticos al trabajo en los hospitales con el objetivo de hacer esta estancia más humana. Su plataforma es bastante sencilla, únicamente nos muestra información y permite realizar donativos, a su vez se pueden comprar diferentes productos en el que los fondos recaudados irán destinados a la fundación. En ningún sitio muestra un apartado para gente que quiera colaborar de voluntariado.
- **Fundación Yo Niño** (<http://www.fundacionyonino.com/>): Yo Niño es una fundación benéfica española dedicada exclusivamente a la infancia donde más nos necesiten. Su plataforma es totalmente estática, por lo que es muy sencilla,



únicamente muestra información de los trabajos que realiza, las últimas noticias y un email de contacto.

- **UNICEF** (<http://www.unicef.es/>): ONG que trabaja para concienciar a la sociedad Española sobre los problemas que afectan a la infancia, movilizar recursos para financiar los programas que llevamos a cabo en más de 150 países en desarrollo y dar respuesta a las emergencias humanitarias que afectan a los niños y las familias. Su plataforma Web es bastante completa, ya que se trata de una de las ONG más grandes del país, es un buen ejemplo para nuestro proyecto.
- **Greenpeace** (<http://www.greenpeace.org/>): ONG ecologista que se encarga de proteger y defender el medio ambiente, interviniendo en diferentes puntos del planeta cuando se cometen atentados contra la naturaleza. Lleva a cabo campañas para detener el cambio climático, proteger la biodiversidad, disminuir la contaminación, acabar con el uso de armas, proteger bosques naturales, entre otras muchas tareas. Su Web al igual que UNICEF es muy completa ya que se trata de una plataforma a nivel mundial.

En la siguiente tabla podemos comparar las ONG analizadas anteriormente, mostrando los beneficios de cada una de ellas.

	Aladina	Tomillo	Theodora	Yo Niño	Unicef	Greenpeace
Registro y acceso a usuarios	✓	✓	✗	✗	✓	✓
Permite donativos	✓	✓	✓	✗	✓	✓
Apartado de voluntariado	✓	✓	✓	✗	✓	✓
HTML y CSS	✓	✓	✓	✓	✓	✓
JavaScript o JQuery	✓	✓	✓	✓	✓	✓
PHP	✓	✗	✗	✗	✗	✗
ASP	✗	✓	✗	✗	✗	✗

Tabla 2: Comparación de otras fundaciones



Desarrollo de una aplicación para la gestión de proyectos no gubernamentales

Las web de UNICEF y Greenpeace, han sido propuestas por los miembros de la ONG, puesto que les parecían atractivas, y que cumplían la mayoría de los requisitos que deseaban que tuviera la web. Como vemos ambas cumplen la mayoría de los requisitos. Otra de las web que tiene la mayoría de las características deseadas es la Fundación Aladina, por lo que extraeremos ideas de diseño de ellas.

3 Marco de trabajo

3.1 Metodología

A lo largo del proyecto se va a seguir una metodología ágil SCRUM, los motivos de esta decisión se detallarán a continuación.

El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada Sprint, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión más completa del software. En cada nuevo Sprint, se va ajustando la funcionalidad ya construida y se añaden nuevas prestaciones.

A pesar de seguir metodología SCRUM no se ha podido realizar la reunión diaria de 15 minutos, por lo tanto podemos denominar que se ha llevado a cabo un SCRUM modificado a la casuística de un TFG personal y con la problemática de reunirse con la ONG y Universidad Europea que realiza otras partes.

3.1.1 Ciclo de vida

El ciclo de vida está formado por varios elementos:

- **Product Backlog:** conjunto de requisitos que se van introduciendo en las distintas tareas a desarrollar. Esta pila de tareas ha de ser dinámica y evolucionar a lo largo de la vida del proyecto, por ello se pueden ir incorporando y reordenando según las necesidades.
- **Sprint Planning:** El equipo determina la cantidad de tareas que puede comprometerse a completar en ese sprint, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir.
- **Sprint:** iteración de duración prefijada durante la cual el equipo trabaja para convertir las tareas del *Product Backlog* a las que se ha comprometido, en una nueva versión del software totalmente operativo.
- **Sprint Backlog:** lista de las tareas necesarias para llevar a cabo las historias del sprint.
- **Daily sprint meeting:** reunión diaria de cómo máximo 15 minutos en la que el equipo se sincroniza para trabajar de forma coordinada. Cada

miembro comenta que hizo el día anterior, que hará hoy, y si hay impedimentos.

- **Demo y retrospectiva:** reunión que se celebra al final del sprint y en la que el equipo presenta las tareas conseguidas mediante una demostración del producto. Posteriormente, en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y se discute acerca de cómo perfeccionarlos.

3.1.2 Roles

En SCRUM, el equipo se focaliza en construir software de calidad. La gestión de un proyecto SCRUM se centra en definir cuáles son las características que debe tener el producto a construir y en vencer cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo.

El equipo SCRUM está formado por los siguientes roles:

- **Scrum master:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología.
- **Product owner (PO):** Se focaliza en la parte de negocio. Traslada la visión del proyecto al equipo, formaliza las prestaciones en historias a incorporar en el *Product Backlog* y las reprioriza de forma regular.
- **Team:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.

3.1.3 Sprints

Se pueden diferenciar los tres roles de esta metodología en este proyecto con facilidad:

- **Product owner:** miembro de la fundación en establecer requisitos e ir controlando la evolución del proyecto con el funcionamiento deseado.
- **Scrum master:** tutor del proyecto que se ha encargado de comprobar que se cumplieran los plazos establecidos, así como el correcto funcionamiento del trabajo establecido en cada sprint.

- **Team:** este rol, es el mío que me he encargado de llevar a cabo el desarrollo de proyecto.

El proyecto se va a dividir en distintos *Sprints*, de una duración aproximada de 4 semanas, en los que se va a ir fijando las distintas tareas de nuestro *product backlog*. Después de cada sprint se va a proceder a una reunión para analizar los objetivos de cada sprint e ir añadiendo nuevas funcionalidades a nuestro *product backlog* fijando las tareas del nuevo *sprint*, conocidas como *sprint backlog*.

3.2 Planificación

La planificación de este proyecto viene marcada firmemente por los requerimientos del cliente y los requerimientos del Trabajo fin de Grado que tiene un peso de 12 créditos, cada uno de ellos equivale 25 horas de trabajo, con un total de 300 horas aproximadamente. Teniendo en cuenta este requisito y proponiendo una organización de no más de 10 horas de trabajo a la semana, el proyecto tendría una duración de 33 semanas aproximadamente. Esta estimación del tiempo se realiza teniendo en cuenta que pilla de por medio el verano, contemplando posibles demoras por parte del cliente, y teniendo en cuenta que estoy trabajando durante toda la realización del proyecto a jornada completa, lo que limita bastante las horas diarias de realizar el TFG.

Se propone la siguiente planificación, comenzando el 27 de Enero de 2015 y con fecha esperada de finalización el 15 de Septiembre de 2015.

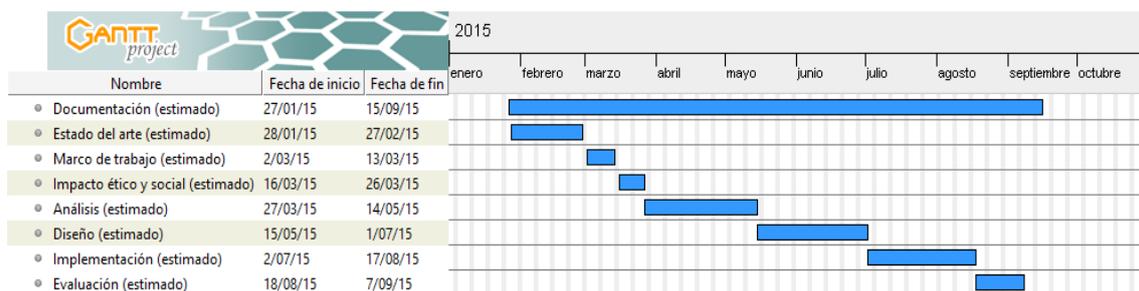


Ilustración 1: GANT

3.3 Presupuesto

Para calcular una estimación del coste del proyecto nos basamos en cuatro factores principales: salario de los participantes, amortizaciones, costes y precio final.

3.3.1 Salario de los participantes

A continuación se muestra una tabla con el coste de los trabajadores de este proyecto. Tenemos dos miembros, el jefe de proyecto y programador. En la siguiente tabla se incluyen los costes del trabajador a la empresa, retenciones de impuestos, seguridad social, y desglose del salario:

Cálculo de Costes Empresariales		
Datos del Empleado		
Puesto	Jefe de Proyecto	Programador
Jornada Laboral	4	6
Horas Semanales	20	30
Retribuciones Brutas		
Retribución Anual	11.000,00 €	12.000,00 €
Salario Base/Mes	916,67 €	1.000,00 €
Pluses		
Salario Neto/Mes	916,67 €	1.000,00 €
Salario Neto/Año	12.833,33 €	14.000,00 €
Salario Cotizable	916,67 €	1.000,00 €
Pagas Extra	152,78 €	166,67 €
Base de Cotización	1.069,44 €	1.166,67 €
Impuesto sobre la Renta de las Personas Físicas		
Retención mensual IRPF	110,00 €	110,00 €
Retención anual IRPF	1.320,00 €	1.320,00 €
Retenciones de la Seguridad Social		
Contingencias Comunes (9,8%)	104,81 €	114,33 €
Accidentes Profesionales (1,8%)	19,25 €	21,00 €
Desempleo (6%)	64,17 €	70,00 €
Formación Profesional (0,6%)	6,42 €	7,00 €
Aportación FOGASA (0,4%)	4,28 €	4,67 €
Total Seguridad Social/Mes	198,92 €	217,00 €
Total Seguridad Social/Año	2.387,00 €	2.604,00 €
Salario Bruto Anual	16.540,33 €	17.924,00 €
Coste de Hora Trabajada	17,23 €	12,45 €

Tabla 3: Costes empresariales

3.3.2 Amortizaciones

Se calcula el coste de los equipos necesarios y su correspondiente amortización para el proyecto, además se incluye el importe del material fungible, aunque sea muy pequeño.

Costes de Equipos				
Concepto	Modelo	Cantidad	Precio Ud.	Total
Ordenador Portátil	HP Envy- J100NS	1	900,00 €	900,00 €
Impresora	Xerox WorkCentre™	1	190,00 €	190,00 €
Tablet	Hacer Iconia 8	1	250,00 €	250,00 €
Total				1340,00€

Tabla 4: Costes de equipos

Amortizaciones		
Tiempo (Semanas)	Amortización Semanal	Base Imponible
33	27,27 €	381,82 €
33	5,76 €	80,61 €
33	7,58 €	106,06 €
Totales	40,61 €	568,48 €

Tabla 5: Amortizaciones

Material Fungible			
Concepto	Cantidad	Precio Ud.	Total
Paquete Folios 500	2	2,50 €	5,00 €
Subrayadores	2	0,80 €	1,60 €
Bolígrafos	2	1,00 €	2,00 €
Memoria USB	3	12,0 €	36,00 €
Post-It	6	0,80 €	4,80 €
Discos ópticos	10	1,20 €	12,00 €
Disco duro	1	82,90	82,90 €
Bolígrafos	2	1,20 €	2,40 €
Total			146,70 €

Tabla 6: Material fungible

3.3.3 Costes

Cálculo del coste real del proyecto.

Gastos de Personal Imputables al Proyecto			
Datos del Empleado			
Puesto	Jefe de Proyecto	Programador	
Horas Semanales	20	30	
Desglose por Categorías y Actividades (Participación en el proyecto en semanas de cada empleado y coste para la empresa)			
Documentación	2,00	4,00	
Estado del arte	0,00	2,00	
Marco de trabajo	1,00	2,00	
Impacto ético, social, económico y legal	0,00	2,00	
Análisis	2,00	6,00	
Diseño	2,00	8,00	
Implementación	0,00	6,00	
Evaluación	1,00	3,00	
Total Semanas Invertidas	8,00	33,00	
Coste Hora Trabajada	17,23 €	12,45 €	
Total Coste	2.756,72 €	12.322,75 €	

Tabla 7: Gastos de Personal

Presupuesto del Proyecto (Costes de la Empresa)			
Desglose por Actividades			
	Personal	Equipos	Fungible
Documentación	2.182,85 €	71,06 €	18,34 €
Estado del arte	746,83 €	71,06 €	18,34 €
Marco de trabajo	1.091,42 €	71,06 €	18,34 €
Impacto ético, social, económico y legal	746,83 €	71,06 €	18,34 €
Análisis	2.929,68 €	71,06 €	18,34 €
Diseño	3.676,51 €	71,06 €	18,34 €
Implementación	2.240,50 €	71,06 €	18,34 €
Evaluación	1.464,84 €	71,06 €	18,34 €
Total	15.079,47 €	568,48 €	146,70 €

Tabla 8: Presupuesto

3.3.4 Resumen presupuesto final

Precio del Proyecto
Costes Empresariales
15.794,66 €
Costes Indirectos (8%)
17.058,23 €
Margen de Seguridad (10%)
18.764,05 €
Margen de Beneficio (15%)
21.578,66 €

Tabla 9: Coste del proyecto

Finalmente, y tras los cálculos oportunos, el precio final del proyecto será de 21.578,66€ (VEINTIÚN MIL QUINIENTOS SETENTA Y OCHO CON SESENTA Y SEIS) sin IVA. Se estipulará un pago del 30% del total por adelantado, con mensualidades posteriores de un 10% sobre el precio final. Una vez entregado el proyecto, se completará el pago con el dinero restante.

Para más información, se adjunta un esquema de la forma de pago:

- Pago inicial (30%): 6473,60€*
- Primera, segunda, tercera y cuarta mensualidad (10% cada una): 2157,87€*
- Pago final (30%): 6473,60€*

* IVA no incluido

3.3.5 Garantía

El plazo de garantía del proyecto será de 24 meses contado a partir de la fecha de aprobación del mismo. Si dentro de este plazo se acreditara la existencia de errores u otro tipo de incidencias en los trabajos efectuados, el cliente tendrá derecho a reclamar la subsanación de los mismos de conformidad con lo preceptuado en este documento.

Terminado el plazo de garantía, quedaremos exentos de responsabilidad en el proyecto sin perjuicio alguno.

[15], [16], [17]

4 Impacto ético, social, económico y legal

La Fundación Madrina es una Organización sin ánimo de lucro que se encarga de proporcionar ayuda a madres y familias en situaciones desfavorables que tienen niños pequeños. Gracias a esta ayuda muchas familias pueden darle los alimentos necesarios para el crecimiento de los más pequeños, ropa, material didáctico para sus estudios, incluso un pequeño detalle en Navidades, evitando la exclusión social.

Como observamos realizan una gran labor social, pero la falta de recursos económicos para invertir en informática les repercute en la eficiencia de sus tareas. Por ejemplo, almacenan todos los datos en ficheros Excel, si se almacenara en una base de datos, reduciríamos notablemente el tiempo al realizar cualquier tipo de consulta. También podemos observar que el sitio web es antiguo y no es adaptativo. De esta manera la Fundación busca una persona que les actualice el sitio web incluyendo una base de datos que gestione los voluntarios, trabajadores de la fundación y actividades que se realizan. Gracias a este proyecto se cubre una necesidad difícilmente de afrontar económicamente por la organización dada que es una ONG.

A lo largo de todo el proyecto se tendrán en cuenta las leyes actuales como L.S.S.I que es la Ley de Servicios de la Sociedad de la información y L.O.P.D. su nombre completo indica Ley Orgánica de Protección de datos. Cumpliendo estas dos leyes, estamos manteniendo la seguridad del sistema para evitar vulnerabilidades. Esto es imprescindible puesto que en las situaciones en las que se encuentra la ONG debe mantener la privacidad de todos los datos que se almacenen en la aplicación desarrollada. La propia fundación ha de registrar la web en La Agencia Española de Protección de Datos para garantizar la fiabilidad de los datos que se almacenen en la fase de producción.

Para evitar posibles ataques mediante la introducción de texto, se valida cada uno de los campos de la aplicación, evitando que se introduzcan caracteres no permitidos, además tienen una longitud limitada. Estos ataques se conocen como *phising* y causan uno de los mayores problemas en la red, ya que se producen miles de ataques de este tipo por segundo.

5 Análisis

5.1 Situación de partida

Actualmente la ONG dispone de una Web totalmente estática, en la que podemos obtener información sobre la misma, interactuando de una manera un tanto compleja y poco eficiente.

La Web no es adaptativa, por lo que no se adapta al tamaño de la pantalla, puesto que si se visualiza en una pantalla grande se ve bien, pero si se ve en una pequeña se cortan algunos trozos y tenemos que hacer *scroll*, no tiene un diseño eficiente.

Por otro lado no existe ninguna gestión de voluntariado. La única manera de inscribirte como voluntario es mediante un correo electrónico, bastante pesado para el voluntario, y los miembros de la ONG que tienen que mirar uno a uno cada email, y enviar una contestación uno a uno.

A su vez la Base de Datos que se va a implementar, se va a adaptar con la de un proyecto que están realizando alumnos de la Universidad Europea, en la que se gestiona la comunicación con los afectados que tienen ayuda por parte de la fundación. De esta manera en un futuro se integrarán ambas bases de datos consiguiendo una aplicación más completa y eficiente.

5.2 Requisitos

A continuación especificaremos la descripción de la tabla de requisitos y los distintos campos que la componen para poder incluir posteriormente la enumeración de todos ellos. Los requisitos vendrán definidos en una tabla como la siguiente:

IDENTIFICADOR	
PRIORIDAD: <input type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	
DESCRIPCIÓN:	

Tabla 10: Plantilla de Requisitos

A continuación se van a explicar en detalle cada uno de los campos de la tabla:

- **Identificador:** este campo sirve para distinguir un requisito de otro inequívocamente. Cada identificador seguirá la siguiente nomenclatura:
 - SXXX-nnn. Los requisitos de capacidad llevarán este identificador, en el que cada letra indica:
 - S: Requisito de software.
 - XXX: Indica de que tipo es el requisito:
 - FNM: Funcionales.
 - RND: de rendimiento.
 - INF: de interfaz.
 - OPN: de operación.
 - RCS: de recursos.
 - CMP: de comprobación.
 - ACP: de aceptación de las pruebas.
 - SGU: de seguridad.
 - CAL: de calidad.
 - MNT: de mantenimiento.
 - DMG: de daño.
 - nnn: Indica el número de requisito, este número es un valor comprendido entre 1 y 999.
- **Prioridad:** este campo indica el grado de prioridad con el que debe ser resuelto un requisito. Podrá tener uno de los siguientes valores:
 - Alta: El diseño es de carácter prioritario.
 - Media: El diseño tiene prioridad media.
 - Baja: El diseño es de prioridad baja.
- **Fuente:** Indica el origen del requisito. Puede ser:
 - Fundación Madrina: Requisito solicitado por el cliente.
 - Desarrollador: Requisito impuesto por el desarrollador.
- **Necesidad:** Este campo indica la necesidad de incorporar el requisito en el sistema. Los valores que puede tener son:
 - Esencial: El requisito debe introducirse en el sistema.



- Deseable: El requisito debería incluirse en el sistema si no causa errores o contradicciones con otros.
- Opcional: El requisito puede o no introducirse en el sistema.
- **Claridad:** Indica si el requisito está expresado correctamente.
 - Alta: el requisito solo tiene una interpretación.
 - Media: El requisito está bien definido y es improbable que surjan dudas.
 - Baja: El requisito puede tener varias interpretaciones dependiendo del contexto.
- **Verificabilidad:** Este campo indica la posibilidad de comprobar que el requisito se haya incorporado al sistema.
 - Alta: Se puede comprobar de forma sencilla.
 - Media: Comprobar si el requisito que se ha incorporado puede suponer algo de trabajo.
 - Baja: Es prácticamente imposible comprobar si el requisito se ha incluido correctamente en el sistema.
- **Estabilidad:** Define la posibilidad de que un requisito no se modifique durante el desarrollo del proyecto.
- **Descripción:** En este campo se incluye una descripción del requisito.

5.2.1 Requisitos funcionales

FNM-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Desarrollo en WordPress.

Tabla 11: Requisito Funcional 001

FNM-002	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Registrarse como voluntario.

Tabla 12: Requisito Funcional 002

FNM-003	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Logearse como voluntario.

Tabla 13: Requisito Funcional 003

FNM-004	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Tener área de donaciones.

Tabla 14: Requisito Funcional 004

FNM-005	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Realizar donaciones redirigiendo a PayPal o tarjeta de crédito/débito.

Tabla 15: Requisito Funcional 005

FNM-006	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Acceso desde múltiples puntos a la aplicación al mismo tiempo.

Tabla 16: Requisito Funcional 006

FNM-007	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema
DESCRIPCIÓN:	Identificación mediante email y <i>password</i> .

Tabla 17: Requisito Funcional 007



FNM-008	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Debe ser posible realizar consultas de los datos de los voluntarios registrados.

Tabla 18: Requisito Funcional 008

FNM-009	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Acceso a la base de datos desde Internet.

Tabla 19: Requisito Funcional 009

FNM-010	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un miembro de la fundación puede modificar los datos de su perfil.

Tabla 20: Requisito Funcional 010

FNM-011	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un miembro de la fundación puede modificar los datos de los voluntarios.

Tabla 21: Requisito Funcional 011

FNM-012	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un miembro de la fundación puede modificar los datos de una actividad.

Tabla 22: Requisito Funcional 012

FNM-013	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un miembro de la fundación puede crear nuevas actividades.

Tabla 23: Requisito Funcional 013

FNM-014	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un miembro de la fundación que sea administrador puede crear nuevos trabajadores.

Tabla 24: Requisito Funcional 014

FNM-015	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un miembro de la fundación que sea administrador puede borrar un voluntario.

Tabla 25: Requisito Funcional 015

FNM-016	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un miembro de la fundación que sea administrador puede borrar una actividad.

Tabla 26: Requisito Funcional 016

FNM-017	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Debe ser posible realizar consultas de las actividades registradas.

Tabla 27: Requisito Funcional 017

FNM-018	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un voluntario puede modificar los datos introducidos, desde su perfil.

Tabla 28: Requisito Funcional 018

FNM-019	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Logearse como miembro de la fundación.

Tabla 29: Requisito Funcional 019

FNM-020	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Un voluntario puede eliminarse de la base de datos.

Tabla 30: Requisito Funcional 020

FNM-021	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Cuando un voluntario se registre, llegará un email a la persona encargada de gestionar voluntarios para que confirme la cuenta de usuario.

Tabla 31: Requisito Funcional 021

FNM-022	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Cuando se haya realizado la validación del voluntario, recibirá un email que tiene la cuenta activa.

Tabla 32: Requisito Funcional 022

FNM-023	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Cuando se asigne una actividad a un voluntario, el mismo recibirá un email con la actividad creada.

Tabla 33: Requisito Funcional 023

5.2.2 Requisitos de interfaz

INF-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Tener una interfaz clara y sencilla, para facilitar el uso a todo tipo de usuarios.

Tabla 34: Requisito de Interfaz 001

INF-002	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	En la pantalla principal del sitio web deben aparecer las últimas noticias de la Fundación.

Tabla 35: Requisito de Interfaz 002

INF-003	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	En la pantalla principal del sitio web debe aparecer en el lado derecho un apartado de donar.

Tabla 36: Requisito de Interfaz 003

INF-004	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	En la pantalla principal del sitio web debe aparecer en el lado derecho un apartado de hazte voluntario.

Tabla 37: Requisito de Interfaz 004



INF-005	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Todas las pantallas estarán compuestas de un menú principal situado en el parte superior de la pantalla.

Tabla 38: Requisito de Interfaz 005

INF-006	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	En la parte inferior de la pantalla aparecerán todos los colaboradores con la Fundación.

Tabla 39: Requisito de Interfaz 006

INF-007	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	En la página de inicio, justo debajo del menú, se muestra un <i>slide</i> con imágenes que acceden a diferentes apartados de la web.

Tabla 40: Requisito de Interfaz 007

5.2.3 Requisitos de rendimiento

SRND-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	El sistema no debe tardar más de 5 segundos en procesar una acción del usuario y generar una respuesta.

Tabla 41: Requisito de Rendimiento 001

SRND-002	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	El sistema estará disponible las 24 horas del día y los 365 días del año.

Tabla 42: Requisito de Rendimiento 002

SRND-003	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	El acceso a la información por parte del sistema será inferior a 5 segundos.

Tabla 43: Requisito de Rendimiento 003

SRND-004	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input checked="" type="checkbox"/> FUNDACIÓN MADRINA <input type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Si ocurre algún error, el sistema debe responder entre 5 y 15 segundos como máximo.

Tabla 44: Requisito de Rendimiento 004

5.2.4 Requisitos de operación

SOPN-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	El periodo operativo abarca las 24 horas del día.

Tabla 45: Requisito de Operación 001

SOPN-002	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	En caso de pérdida de información, el sistema no tardará más de 2 horas en sustituir los datos del servidor principal desde los back-up's.

Tabla 46: Requisito de Operación 002

5.2.5 Requisitos de recursos

SRCS-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Es necesario un navegador Web para poder utilizar la aplicación.

Tabla 47: Requisito de Recursos 001

5.2.6 Requisitos de comprobación

SCMP-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Emplear la tecnología JavaScript para comprobar que los datos introducidos por el usuario (CP, e-mail, etc.) son correctos.

Tabla 48: Requisito de Comprobación 001

SCMP-002	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	No admitir caracteres que no pertenezcan al alfabeto latino.

Tabla 49: Requisito de Comprobación 002

SCMP-003	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Comprobar que el email tenga un formato correcto.

Tabla 50: Requisito de Comprobación 003

SCMP-004	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Comprobar que el número de teléfono sea correcto.

Tabla 51: Requisito de Comprobación 004

SCMP-005	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Comprobar que el código postal tenga 5 dígitos numéricos.

Tabla 52: Requisito de Comprobación 005

5.2.7 Requisitos de aceptación de las pruebas

SACP-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Superar pruebas de integración.

Tabla 53: Requisito de Aceptación de las pruebas 001

SACP-002	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Superar pruebas del sistema.

Tabla 54: Requisito de Aceptación de las pruebas 002

SACP-003	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Superar pruebas de aceptación.

Tabla 55: Requisito de Aceptación de las pruebas 003

SACP-004	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Superar pruebas de regresión.

Tabla 56: Requisito de Aceptación de las pruebas 004

SACP-005	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Superar pruebas unitarias.

Tabla 57: Requisito de Aceptación de las pruebas 005

5.2.8 Requisitos de daño

SDMG-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Comprobar datos bancarios cuando se realice una donación, para que no se introduzca un número de cuenta sin saldo, ni inválido.

Tabla 58: Requisito de Daño 001

5.2.9 Requisitos de seguridad

SSGU-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Las contraseñas en la base de datos se almacenaran de forma segura.

Tabla 59: Requisito de seguridad 001

SSGU-002	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Seguridad en el pago, avalado por PayPal.

Tabla 60: Requisito de seguridad 002

SSGU-003	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Los datos almacenados en la Base de Datos cumplirán la normativa LOPD.

Tabla 61: Requisito de seguridad 003

5.2.10 Requisitos de mantenimiento

SMNT-001	
PRIORIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	FUENTE: <input type="checkbox"/> FUNDACIÓN MADRINA <input checked="" type="checkbox"/> DESARROLLADORA
NECESIDAD: <input checked="" type="checkbox"/> ESENCIAL <input type="checkbox"/> DESEABLE <input type="checkbox"/> OPCIONAL	
CLARIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA	VERIFICABILIDAD: <input checked="" type="checkbox"/> ALTA <input type="checkbox"/> MEDIA <input type="checkbox"/> BAJA
ESTABILIDAD:	Durante toda la vida del sistema.
DESCRIPCIÓN:	Configurar el sistema de seguridad para que todas las incidencias puedan resolverse rápidamente en caso de avería o accidente.

Tabla 62: Requisito de Mantenimiento 001

5.3 Análisis de casos de uso

Definiremos los casos de uso utilizando como base la siguiente tabla:

Identificador	
Caso de uso	
Actor	
Objetivo	
Flujo	
Precondiciones	
Post-condiciones	

Tabla 63: Plantilla casos de uso

- **Identificador:** Nombre con el que se hará referencia al caso de uso. Su nomenclatura será "CU_XX", donde "CU" hace referencia al caso de uso y "XX" a la secuencia numérica.
- **Caso de uso:** Nombre que recibe en el diagrama.
- **Actor:** Agente que interactúa con el caso de uso.
- **Objetivo:** Descripción detallada del caso de uso.
- **Flujo:**
- **Precondiciones:** Condiciones iniciales que se han de cumplir para poder realizar el caso de uso.
- **Post-condiciones:** Estado del sistema después del caso de uso.

A continuación se muestra el diagrama de casos de uso:

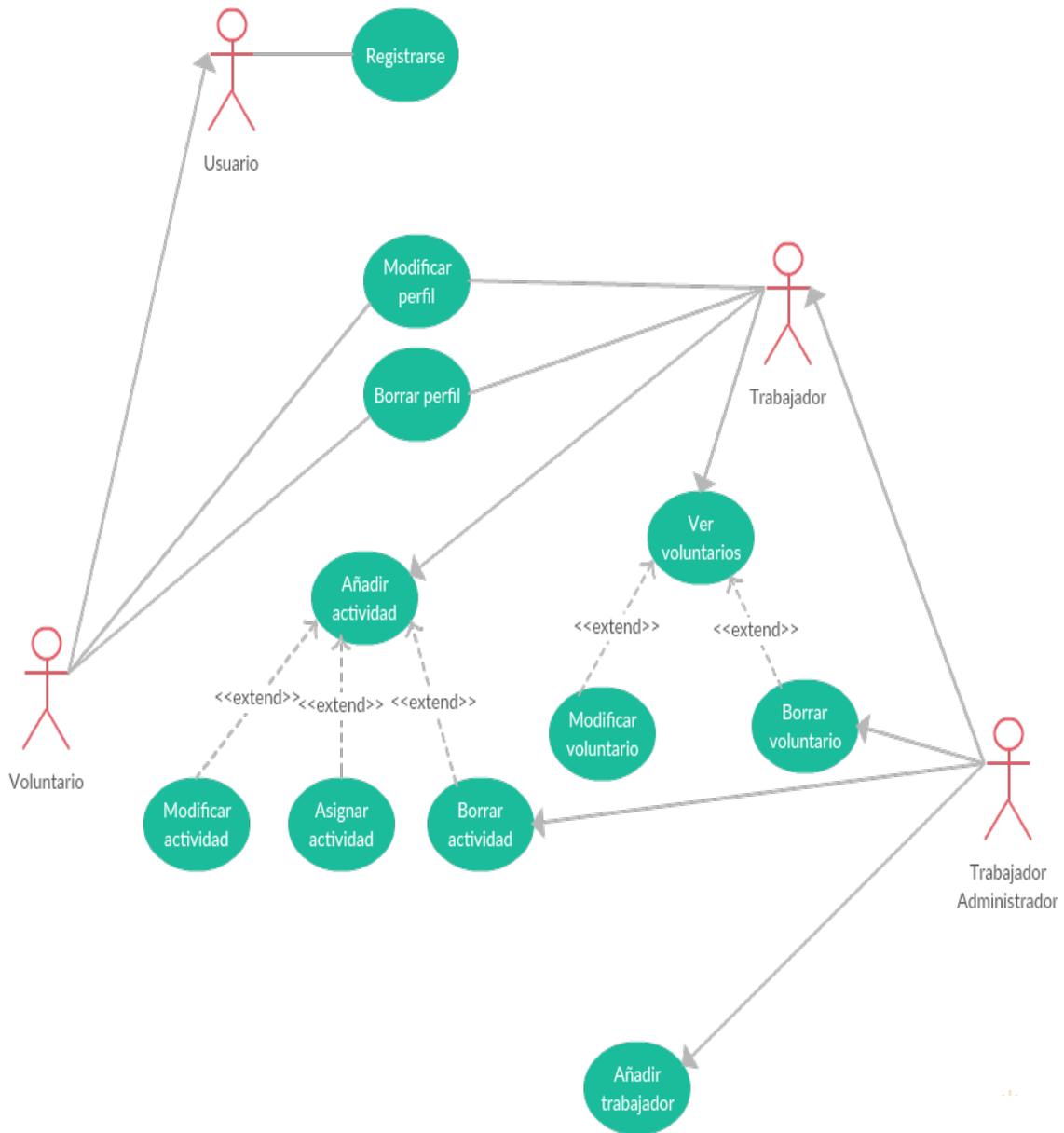


Ilustración 2: Diagrama casos de uso

Los casos de uso son los siguientes:

CU_01	
Caso de uso	Registrarse.
Actor	Usuario.
Objetivo	El usuario accede al área de registro, para inscribirse como voluntario.
Flujo básico	<ol style="list-style-type: none"> 1. El usuario accede a la página de hazte voluntario. 2. Completa los datos del formulario. 3. El sistema notifica que se ha registrado el usuario. 4. El sistema envía un email de confirmación.
Precondiciones	<ul style="list-style-type: none"> • Ninguna.
Post-condiciones	<ul style="list-style-type: none"> • Si uno de los campos este vacío, despliega un mensaje de error diciendo que se completen los campos restantes. • Si alguno de los campos no es del tipo esperado, despliega un mensaje de error diciendo que se completen con el tipo adecuado de dato.

Tabla 64: Caso de uso 01

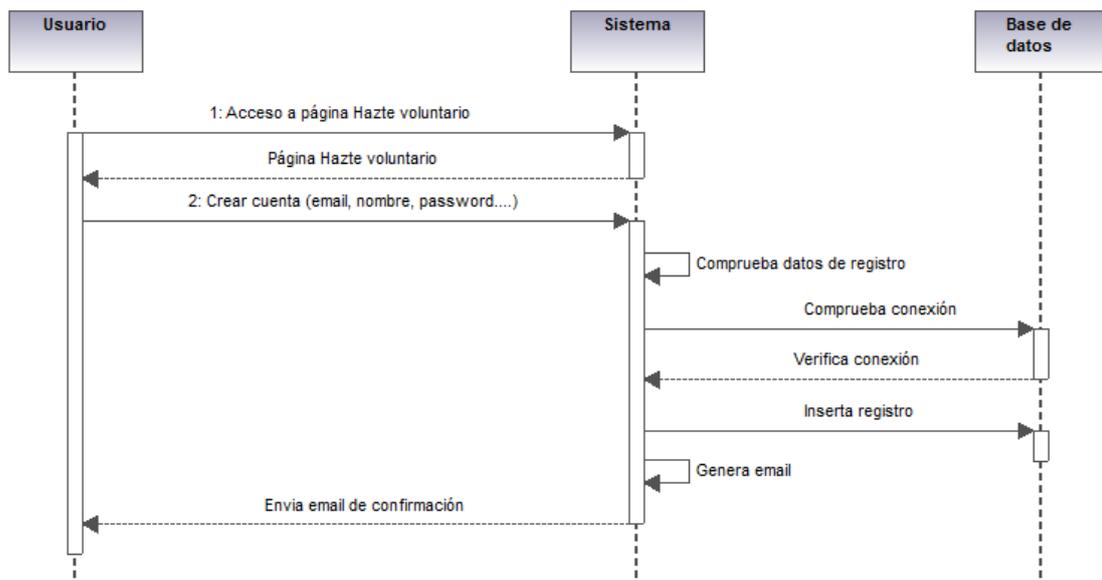


Ilustración 3: Diagrama de secuencia CU_01



CU_02	
Caso de uso	Modificar perfil.
Actor	Voluntario, Trabajador.
Objetivo	El usuario una vez iniciada la sesión puede modificar los datos de usuario.
Flujo básico	<ol style="list-style-type: none"> 1. El usuario accede a la página de modificar perfil. 2. El sistema muestra los datos del perfil de usuario mediante campos editables. 3. El usuario modifica los datos que le interesan. 4. El usuario pulsa modificar. 5. El sistema notifica que la información ha sido actualizada.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión. • Estar en la página mi perfil.
Post-condiciones	<ul style="list-style-type: none"> • Si uno de los campos este vacío, despliega un mensaje de error diciendo que se completen los campos restantes. • Si alguno de los campos no es del tipo esperado, despliega un mensaje de error diciendo que se completen con el tipo adecuado de dato.

Tabla 65: Caso de uso 02

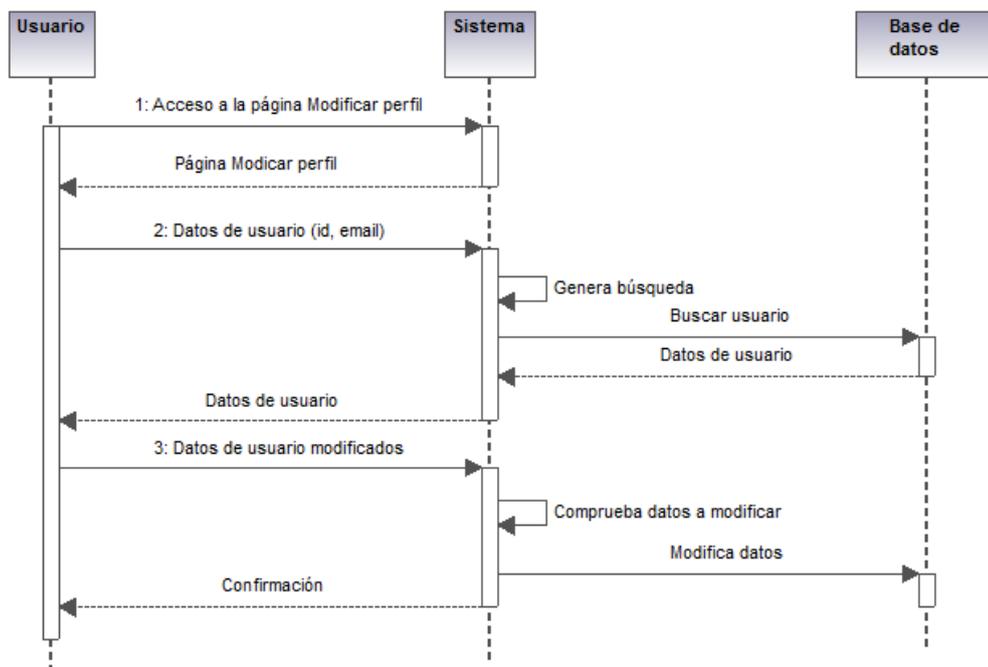


Ilustración 4: Diagrama de secuencia CU_02

CU_03	
Caso de uso	Borrar perfil.
Actor	Voluntario, Trabajador.
Objetivo	El usuario una vez iniciada la sesión puede eliminarse del sistema.
Flujo básico	<ol style="list-style-type: none"> 1. El usuario selecciona la opción borrar usuario. 2. El sistema notifica que se ha eliminado la cuenta de usuario.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión. • Estar en la página de mi perfil.
Post-condiciones	<ul style="list-style-type: none"> • Ninguna.

Tabla 66: Caso de uso 03

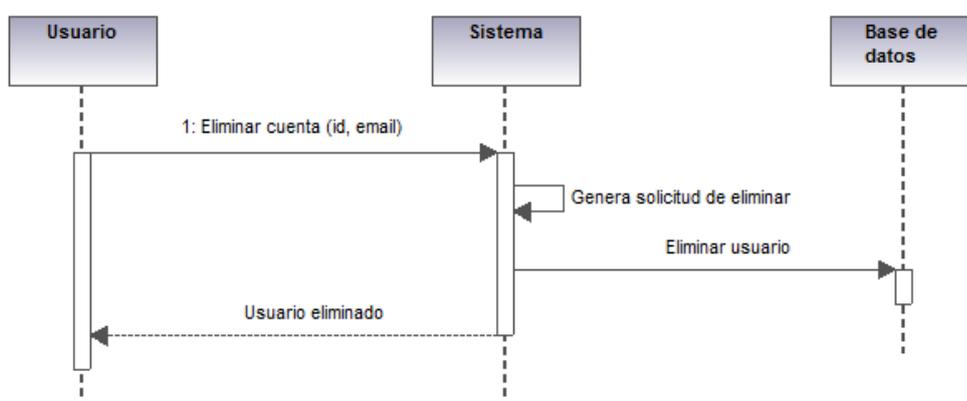


Ilustración 5: Diagrama de secuencia CU_03

CU_04	
Caso de uso	Modificar una actividad.
Actor	Trabajador.
Objetivo	Los trabajadores pueden ver los modificar los datos de una actividad.
Flujo	<ol style="list-style-type: none"> 1. El usuario accede a la página de modificar actividad. 2. El sistema muestra los datos de la actividad mediante campos editables. 3. El usuario modifica los datos de la actividad que le interesan. 4. El usuario pulsa modificar. 5. El sistema notifica que la información ha sido actualizada.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión como trabajador de la fundación. • Tener esa actividad en la base de datos.
Post-condiciones	<ul style="list-style-type: none"> • Ninguna.

Tabla 67: Caso de uso 04

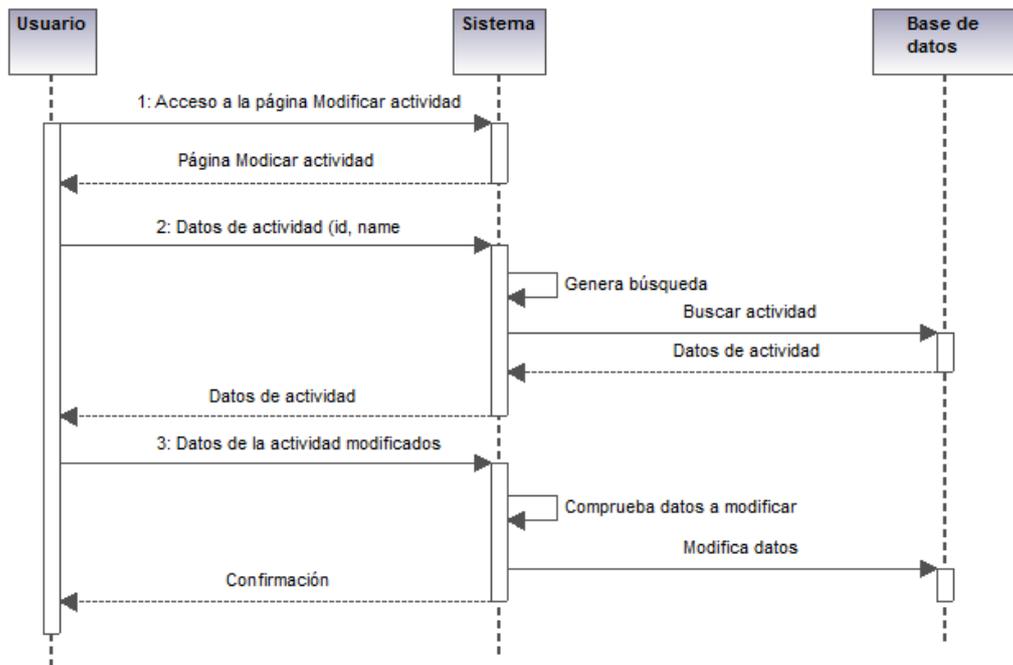


Ilustración 6: Diagrama de secuencia CU_04

CU_05	
Caso de uso	Borrar una actividad.
Actor	Trabajador.
Objetivo	Los trabajadores pueden borrar una actividad.
Flujo	<ol style="list-style-type: none"> 1. El usuario selecciona la opción borrar actividad. 2. El sistema notifica que se ha eliminado la actividad.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión como trabajador de la fundación. • Tener esa actividad en la base de datos. • Ser administrador.
Post-condiciones	<ul style="list-style-type: none"> • Ninguna.

Tabla 68: Caso de uso 05

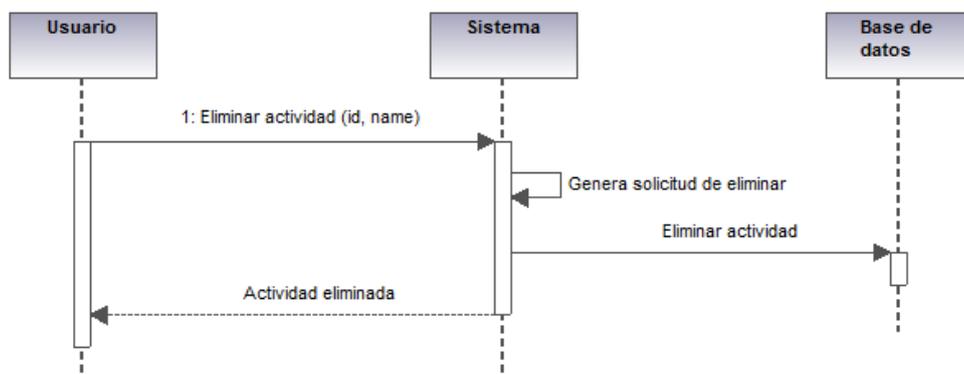


Ilustración 7: Diagrama de secuencia CU_05

CU_06	
Caso de uso	Añadir actividad.
Actor	Trabajador.
Objetivo	Los trabajadores pueden añadir una actividad.
Flujo	<ol style="list-style-type: none"> 1. El usuario accede a la página de añadir actividad. 2. Completa los datos del formulario. 3. El sistema notifica que se ha añadido la actividad.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión como trabajador de la fundación.
Post-condiciones	<ul style="list-style-type: none"> • Si uno de los campos este vacío, despliega un mensaje de error diciendo que se completen los campos restantes. • Si alguno de los campos no es del tipo esperado, despliega un mensaje de error.

Tabla 69: Caso de uso 06

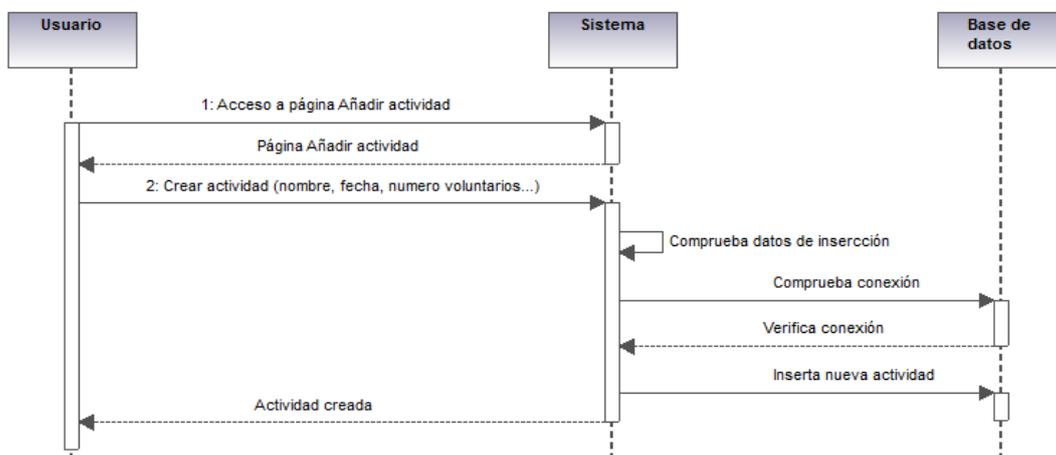


Ilustración 8: Diagrama de secuencia CU_06

CU_07	
Caso de uso	Asignar actividad.
Actor	Trabajador.
Objetivo	Los trabajadores pueden asignar actividades a los voluntarios.
Flujo básico	<ol style="list-style-type: none"> 1. El usuario accede a la página de asignar actividad. 2. El sistema muestra una lista con todos los voluntarios almacenados en el sistema. 3. El usuario selecciona el usuario a asignar. 4. El sistema comprueba que no esté asignado y que él no supere el límite de horas asignadas a un voluntario. 5. El sistema notifica que se ha insertado la asignación.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión como trabajador de la fundación. • Tener la actividad almacenada en la base de datos.
Post-condiciones	<ul style="list-style-type: none"> • Ninguna.

Tabla 70: Caso de uso 07

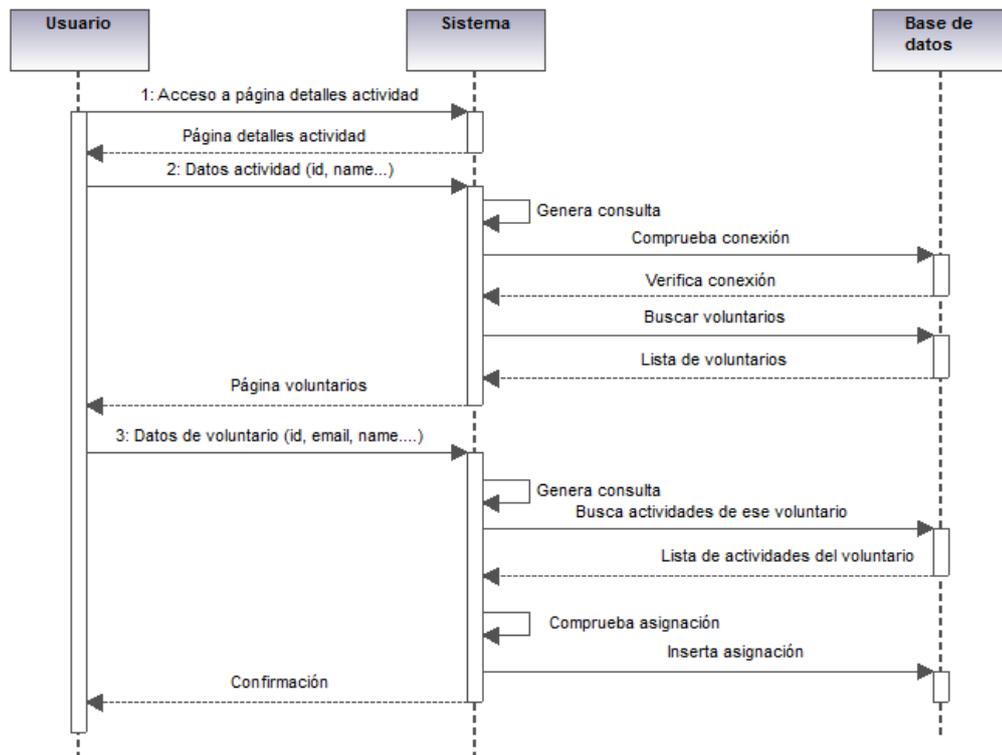


Ilustración 9: Diagrama de secuencia CU_07

CU_08	
Caso de uso	Añadir trabajador.
Actor	Trabajador.
Objetivo	Los trabajadores pueden añadir un trabajador.
Flujo	<ol style="list-style-type: none"> 1. El usuario accede a la página de añadir trabajador. 2. Completa los datos del formulario. 3. El sistema notifica que se ha añadido el trabajador. 4. El sistema envía un email de confirmación.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión como trabajador de la fundación. • Ser administrador.
Post-condiciones	<ul style="list-style-type: none"> • Si uno de los campos este vacío, despliega un mensaje de error diciendo que se completen los campos restantes. • Si alguno de los campos no es del tipo esperado, despliega un mensaje de error diciendo que se completen con el tipo adecuado de dato.

Tabla 71: Caso de uso 08

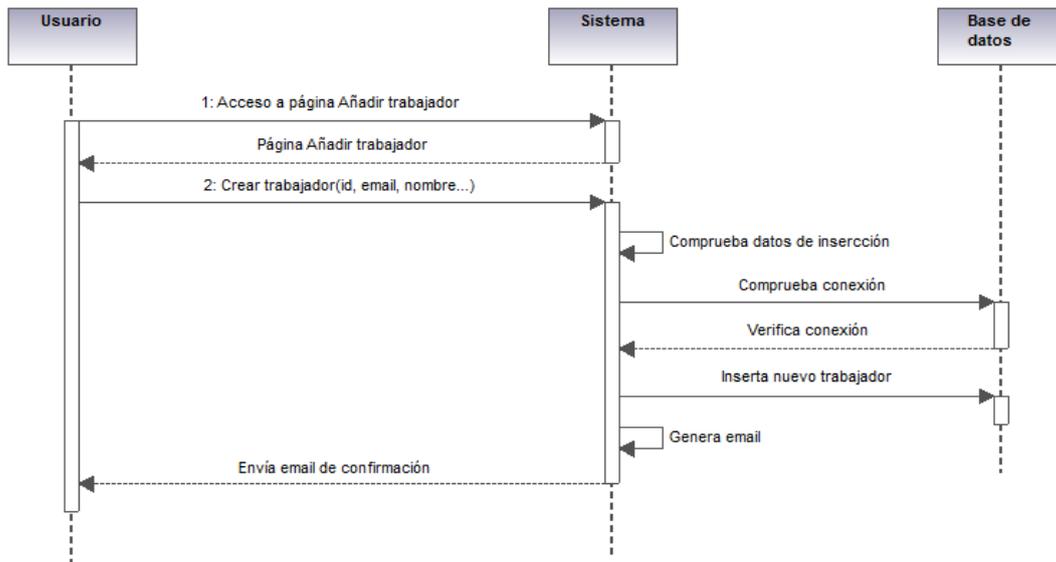


Ilustración 10: Diagrama de secuencia CU_08

CU_09	
Caso de uso	Modificar voluntarios.
Actor	Trabajador.
Objetivo	Los trabajadores de la fundación pueden modificar los datos de los usuarios registrados.
Flujo	<ol style="list-style-type: none"> 1. El usuario accede a la página de modificar voluntario. 2. El sistema muestra los datos del perfil de voluntario mediante campos editables. 3. El usuario modifica los datos que le interesan. 4. El usuario pulsa modificar. 5. El sistema notifica que la información ha sido actualizada.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión como trabajador de la fundación. • Tener seleccionado un voluntario.
Post-condiciones	<ul style="list-style-type: none"> • Si uno de los campos este vacío, despliega un mensaje de error diciendo que se completen los campos restantes. • Si alguno de los campos no es del tipo esperado, despliega un mensaje de error diciendo que se completen con el tipo adecuado de dato.

Tabla 72: Caso de uso 09

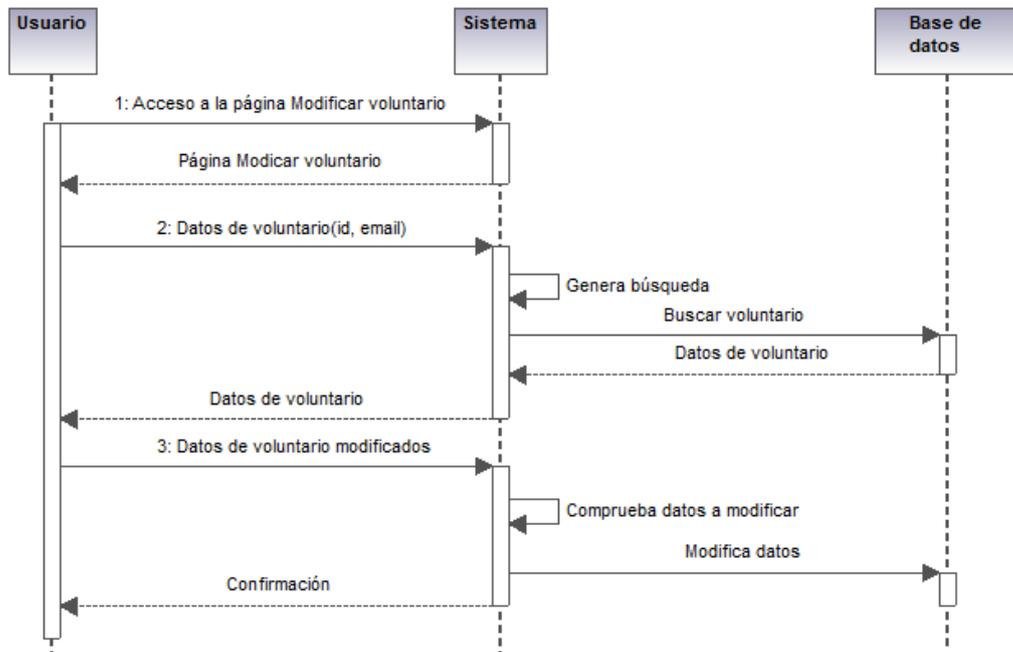


Ilustración 11: Diagrama de secuencia CU_09

CU_10	
Caso de uso	Borrar voluntarios.
Actor	Trabajador.
Objetivo	Los trabajadores de la fundación borrar a un usuario registrado.
Flujo	<ol style="list-style-type: none"> 1. El usuario selecciona la opción borrar voluntario. 2. El sistema notifica que se ha eliminado la cuenta de voluntario.
Precondiciones	<ul style="list-style-type: none"> • Haber iniciado sesión como trabajador de la fundación. • Tener seleccionado un voluntario. • Ser administrador.
Post-condiciones	<ul style="list-style-type: none"> • Ninguna.

Tabla 73: Caso de uso 10

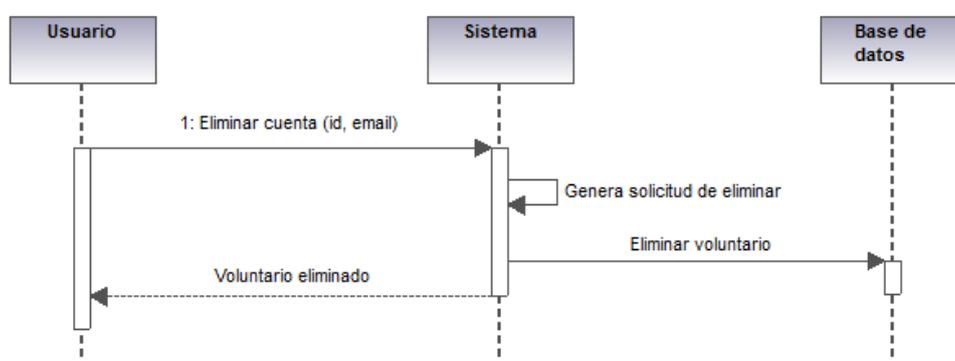


Ilustración 12: Diagrama de secuencia CU_10

6 Diseño

6.1 Definición de la arquitectura del sistema

Para el diseño de la aplicación haremos uso de las especificaciones detalladas en el apartado anterior de Análisis. La finalidad es definir la arquitectura del sistema y del entorno tecnológico que le va a dar soporte, junto con una especificación detallada de los componentes de información que forman parte de la misma.

6.1.1 Definición de niveles de arquitectura

Se ha decidido implementar una arquitectura Modelo Vista Controlador, que se divide en los siguientes subsistemas:

- **Modelo:** almacena los datos útiles del sistema. En él encontramos los datos que representan los objetos relevantes del dominio.
- **Vista:** es la parte del sistema con la que va a interactuar el usuario. Realizará la gestión de la interfaz y se comunicará con el modelo y el controlador, además de presentar la aplicación final al usuario.
- **Controlador:** la parte del sistema que recibe peticiones de la vista, recoge los datos del modelo y devuelve una respuesta al cliente. Desempeña las siguientes funciones:
 - La gestión de actividades: será el encargado de poner a disposición de los voluntarios y trabajadores todas actividades almacenadas en el sistema, así como la gestión de las mismas.
 - La gestión de voluntarios: será el encargado de poner a disposición la información almacenada en el sistema a voluntarios y trabajadores.
 - La gestión de trabajadores: será el encargado de poner a disposición la información almacenada en el sistema a los trabajadores.
 - El acceso a la información: encargado de almacenar toda la información relativa a la aplicación y de su posible modificación.
 - Notificación: encargado de enviar los email a los usuarios.

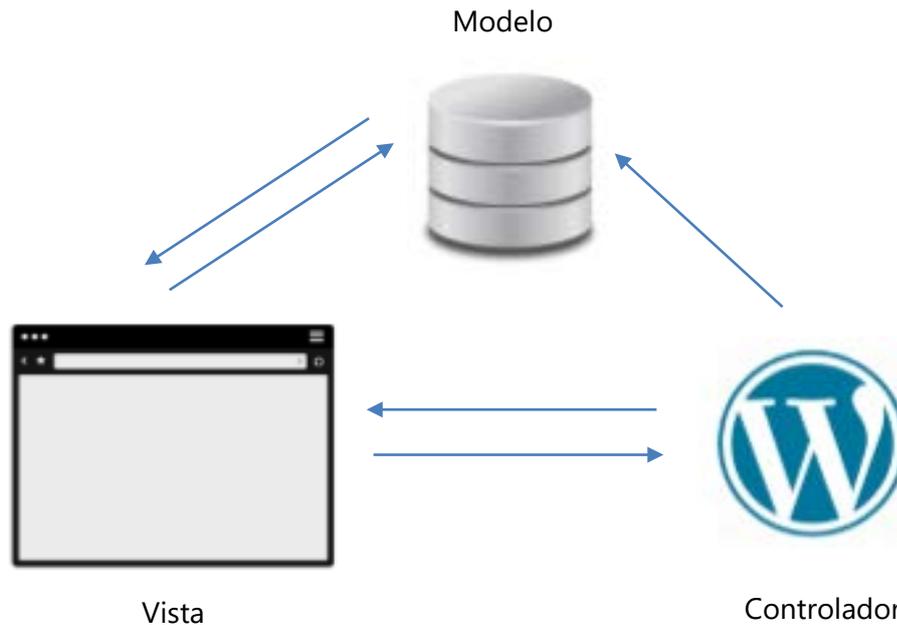


Ilustración 13: Modelo vista controlador

6.1.2 Especificación de estándares y normas de diseño y construcción

Para el diseño también es importante definir los estándares técnicos y de nomenclatura a seguir, así como una serie de normas y recomendaciones que se deben emplear en la construcción del sistema. Se definen pues, los siguientes elementos:

- **Idioma:** el predeterminado será el español aunque en el diseño de la base de datos se va a realizar en inglés por interoperabilidad del sistema.
- **Nombre de las clases:** las clases a implementar siempre deberán empezar por mayúscula. En caso de estar formado por más de una palabra, la primera letra de cada palabra deberá empezar por mayúscula. En caso de que el nombre este compuesto de dos palabras, la segunda comenzará en mayúsculas.
- **Nombre de las variables:** deben estar formados por letras minúsculas, números y guiones bajos. Los nombres de las variables deben ser identificativos.
- **Nombre de las constantes:** deben estar escritas en mayúsculas.
- **Métodos o funciones:** deben ser nombrados de tal manera que cada letra de las palabras que los compongan deben ser mayúsculas y deben ir acompañados de los paréntesis.

- **Comentarios:** para escribir un comentario este debe empezar por “/*” luego se pone el comentario y se finaliza con “*/”. Pero también se pueden escribir si al principio se ponen “//”. El comentario debe ser legible y aclarativo.
- **Nombre de los paquetes:** deben indicar claramente y de manera concisa lo que se está realizando.

6.1.3 Especificación de excepciones

Es importante tener en cuenta las situaciones anómalas que se puedan producir en el funcionamiento y ejecución de la aplicación.

EX-001	
TIPO:	Caída del Servicio de la Base de datos.
DESCRIPCIÓN:	Producida cuando se cae el servidor de la base de datos y no se pueden atender las peticiones que requieren de dicho servicio.
ESTADO:	Caída del servidor de la base de datos.
ELEMENTOS AFECTADOS:	Controlador.
RESPUESTA:	El sistema mostrará un mensaje de error por pantalla, indicando al usuario que se ha perdido la conexión a Internet.

Tabla 74: Excepción 01

EX-002	
TIPO:	Error de conexión de red.
DESCRIPCIÓN:	Se produce cuando el dispositivo pierde la conexión a internet y se intenta realizar alguna acción.
ESTADO:	El dispositivo pierde el acceso a la red.
ELEMENTOS AFECTADOS:	Todos los módulos.
RESPUESTA:	El sistema mostrará un mensaje de error por pantalla, indicando al usuario que se ha perdido la conexión a internet.

Tabla 75: Excepción 02

EX-003	
TIPO:	Caída del servidor Web.
DESCRIPCIÓN:	Se produce cuando se cae el servidor web y no se puede atender a ninguna petición.
ESTADO:	Caída del servidor web.
ELEMENTOS AFECTADOS:	Todos los módulos.
RESPUESTA:	El sistema dejará de funcionar de forma completa manteniendo seguros los datos.

Tabla 76: Excepción 03

6.2 Funcionamiento y taxonomía de la interfaz

La interfaz del sistema diseñada debe permitir a los usuarios del sistema realizar todas las funciones definidas anteriormente en los casos de uso de manera sencilla y cómoda. A continuación analizaremos la interfaz diseñada.

Encabezado

Sin iniciar sesión

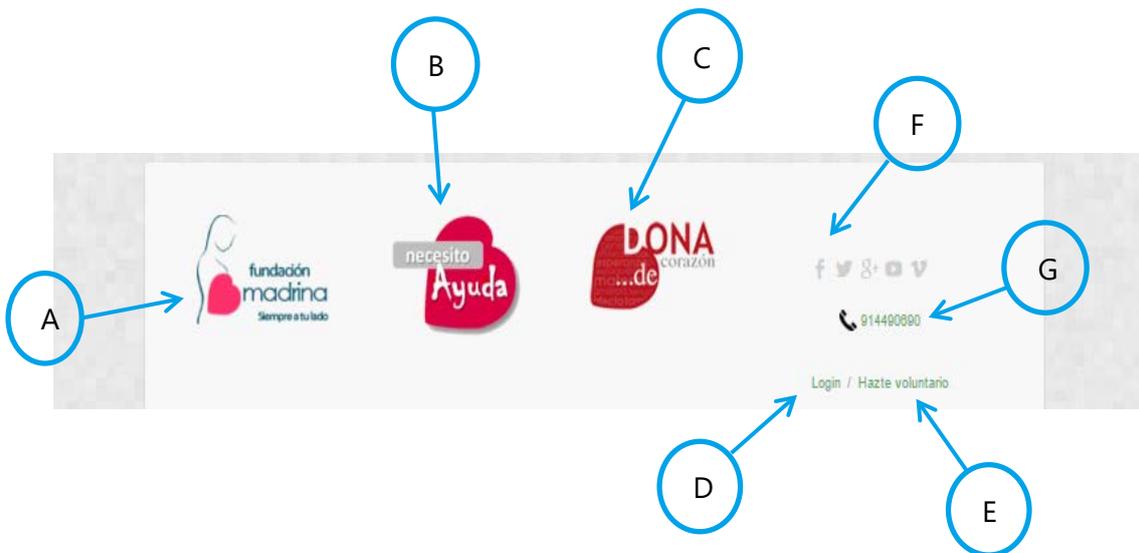


Ilustración 14: Encabezado sin inicio sesión

Con sesión iniciada



Ilustración 15: Encabezado con inicio de sesión

El encabezado principal de la aplicación, aparecerá en todas las páginas y cambiará dependiendo si el usuario tiene iniciada la sesión o no y consta de los siguientes elementos:

ENCABEZADO			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Inicio	Botón	Enlace con la página de inicio
B	Necesito ayuda	Botón	Enlace con la página necesito ayuda
C	Dona	Botón	Enlace con la página dona
D	Login	Botón	Enlace con la página login
E	Hazte voluntario	Botón	Enlace con la página hazte voluntario
F	Redes sociales	Botón	Enlace con las redes sociales
G	Teléfono	Botón	Enlace con el teléfono para llamar
H	Nombre	Botón	Enlace con el perfil del usuario logeado
I	Logout	Botón	Cierra la sesión

Tabla 77: Encabezado

Menús de navegación

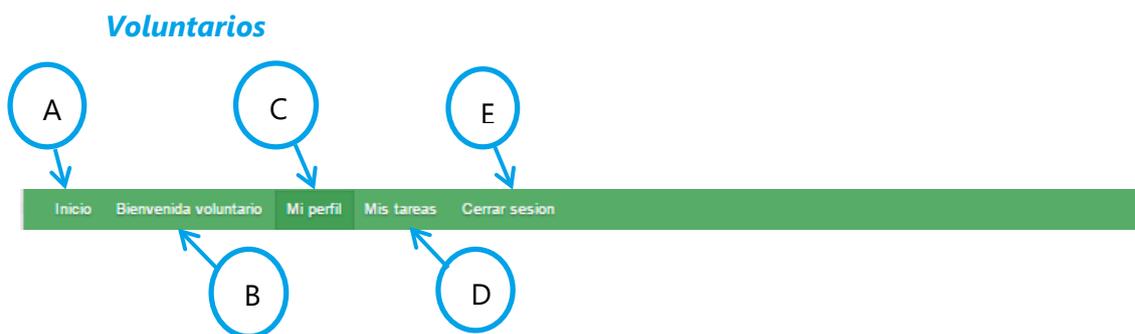


Ilustración 16: Menú navegación voluntarios

El menú de navegación de voluntarios, muestra las diferentes secciones a las que tendrá acceso un voluntario, y consta de los siguientes elementos:

MENÚ DE NAVEGACIÓN VOLUNTARIOS			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Inicio	Botón	Enlace con la página Inicio
B	Bienvenida voluntario	Botón	Enlace con la página Bienvenida voluntario
C	Mi perfil	Botón	Enlace con la página Mi perfil
D	Mis tareas	Botón	Enlace con la página Mis tareas
E	Cerrar sesión	Botón	Cierra sesión

Tabla 78: Menú Navegación voluntarios

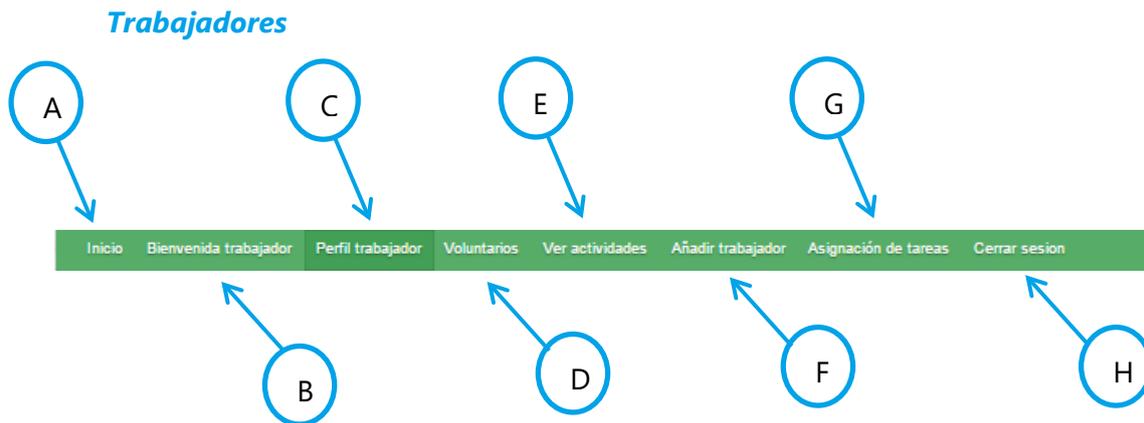


Ilustración 17: Menú navegación trabajadores

El menú de navegación de trabajadores, muestra las diferentes secciones a las que tendrá acceso un trabajador, y consta de los siguientes elementos:

MENÚ DE NAVEGACIÓN TRABAJADORES			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Inicio	Botón	Enlace con la página Inicio
B	Bienvenida trabajador	Botón	Enlace con la página Bienvenida trabajador
C	Perfil trabajador	Botón	Enlace con la página Perfil trabajador
D	Voluntarios	Botón	Enlace con la página Voluntarios
E	Ver actividades	Botón	Enlace con la página Ver actividades
F	Añadir trabajador	Botón	Enlace con la página Añadir trabajador
G	Asignación de tareas	Botón	Enlace con la página Asignación de tareas
H	Cerrar sesión	Botón	Cierra sesión

Tabla 79: Menú Navegación trabajadores



Ilustración 18: Menú navegación principal

El menú de navegación principal de la aplicación, muestra las diferentes secciones de nuestra aplicación, este menú es igual para usuarios sin registrar, voluntarios y trabajadores, y consta de los siguientes elementos:

MENÚ DE NAVEGACIÓN PRINCIPAL			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Inicio	Botón	Enlace con la página Inicio
B	Quienes somos	Botón	Enlace con la página Quienes somos
C	Que hacemos	Botón	Enlace con la página Que hacemos
D	Empresas solidarias	Botón	Enlace con la página Empresas solidarias
E	Voluntariado	Botón	Enlace con la página Voluntariado
F	Sala de prensa	Botón	Enlace con la página Sala de prensa
G	Hazte voluntario	Botón	Enlace con la página Hazte voluntario

Tabla 80: Menú Navegación principal

Pie de página

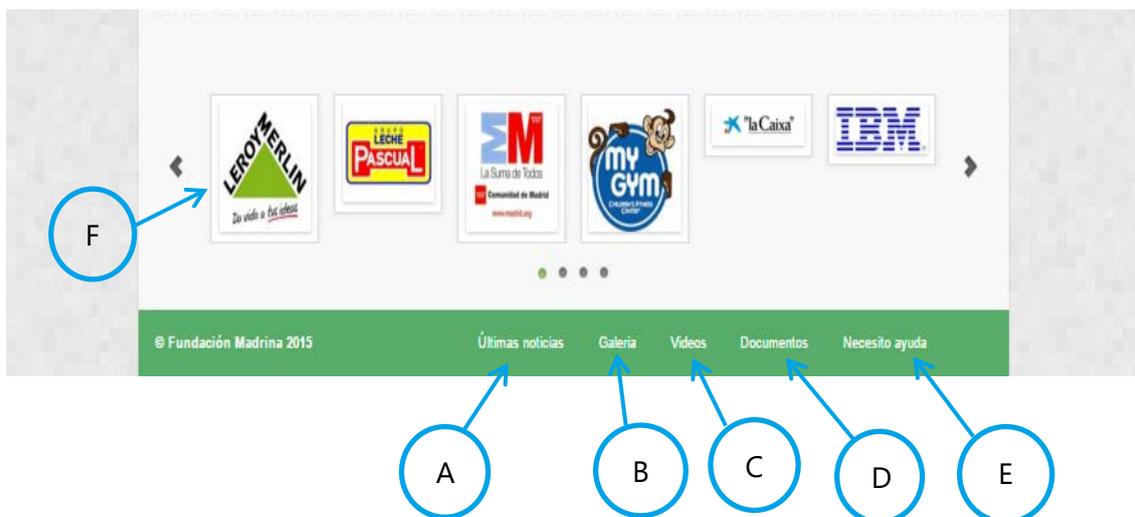


Ilustración 19: Pie de página

El pie de página de la aplicación, aparecerá en todas las páginas y consta de los siguientes elementos:

PIE DE PÁGINA			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Últimas noticias	Botón	Enlace con la página Últimas noticias
B	Galería	Botón	Enlace con la galería de fotos de Google
C	Videos	Botón	Enlace con los videos de YouTube
D	Documentos	Botón	Enlace a documentos en SlideShare
E	Necesito ayuda	Botón	Enlace con la página Necesito ayuda
F	Slide	Slide	Muestra imágenes de los patrocinadores

Tabla 81: Pie de página

Login

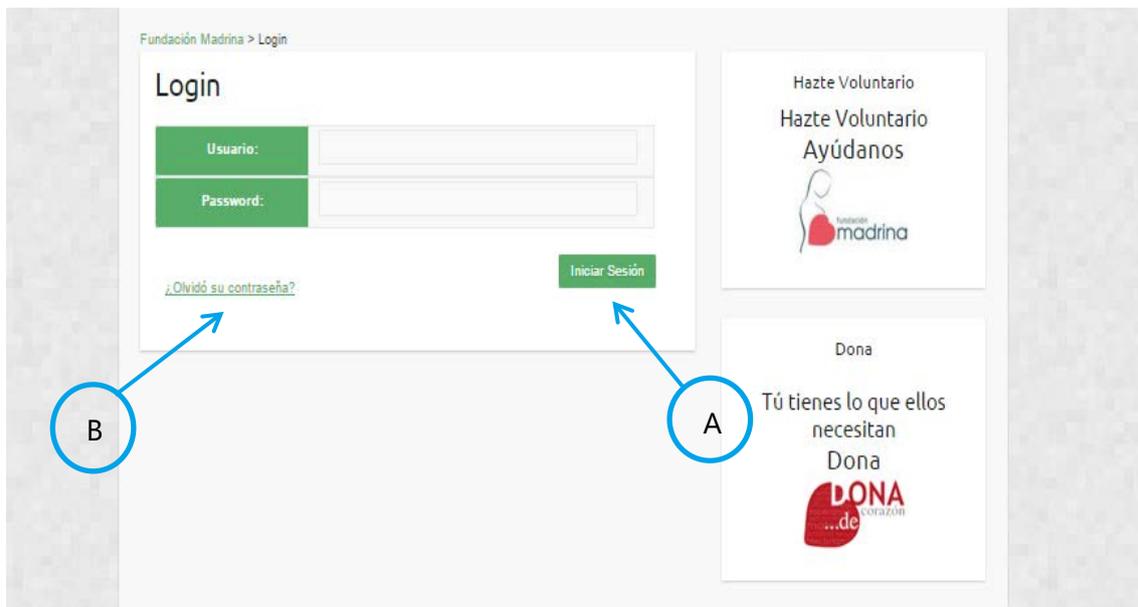


Ilustración 20: Pantalla Login

La pantalla de login permite a los usuarios acceder su área de servicios y consiste en un sencillo y habitual formulario de inicio de sesión:

LOGIN			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Iniciar sesión	Botón	Inicia sesión
B	Olvido su contraseña	Botón	Enlace con la página Olvidó su contraseña

Tabla 82: Pantalla Login

Hazte voluntario

Ilustración 21: Pantalla Hazte voluntario

La pantalla de hazte voluntario permite a cualquier usuario registrarse como voluntario. Consta de los siguientes elementos:

HAZTE VOLUNTARIO			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Hazte voluntario	Formulario	Añade un nuevo voluntario a la base de datos
B	Tabla hazte voluntario	Tabla	-

Tabla 83: Pantalla Hazte voluntario

Mi perfil

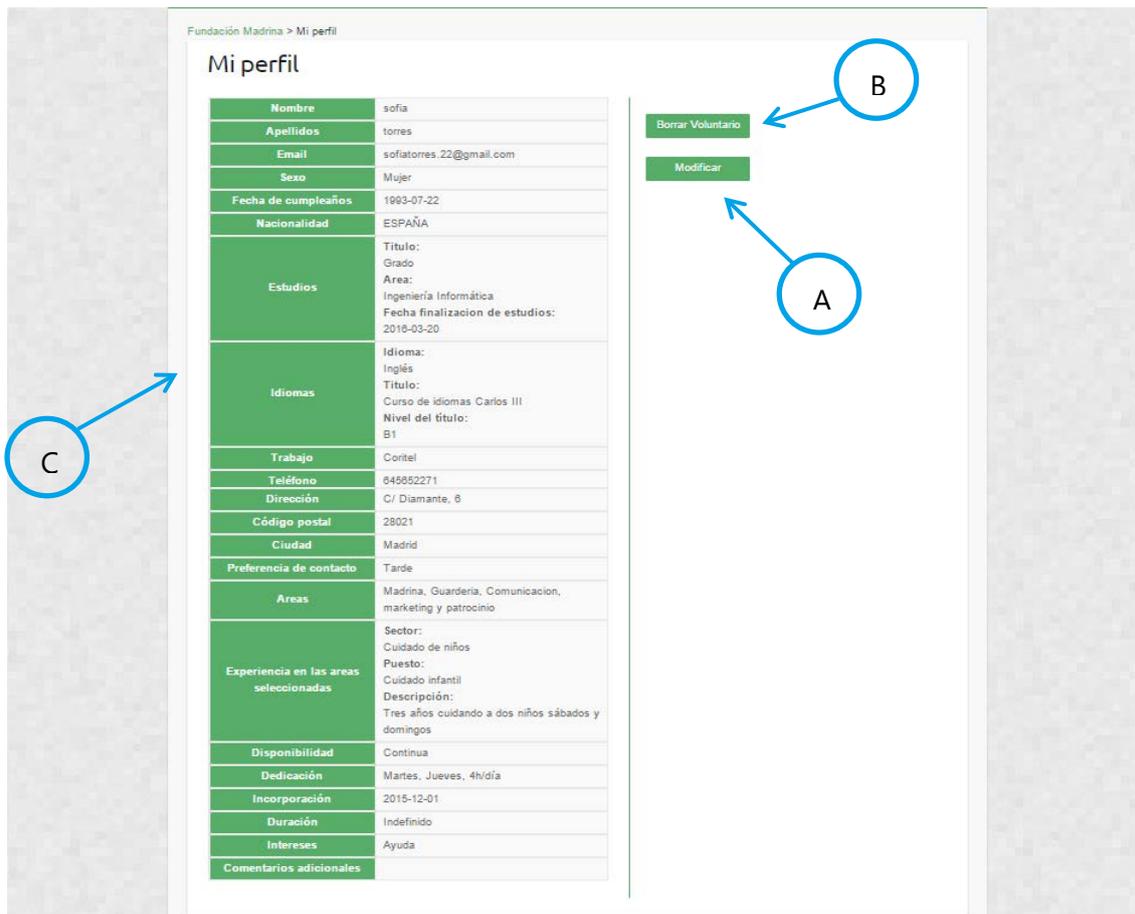


Ilustración 22: Pantalla mi Perfil

La pantalla de mi perfil permite a los voluntarios ver su información de perfil, editarla y eliminarse como voluntario. Consta de los siguientes elementos:

MI PERFIL			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Modificar	Botón	Enlace con la página Modificar mi perfil
B	Borrar	Botón	Elimina el usuario de la base de datos
C	Tabla mi perfil	Tabla	-

Tabla 84: Pantalla mi perfil

Modificar perfil

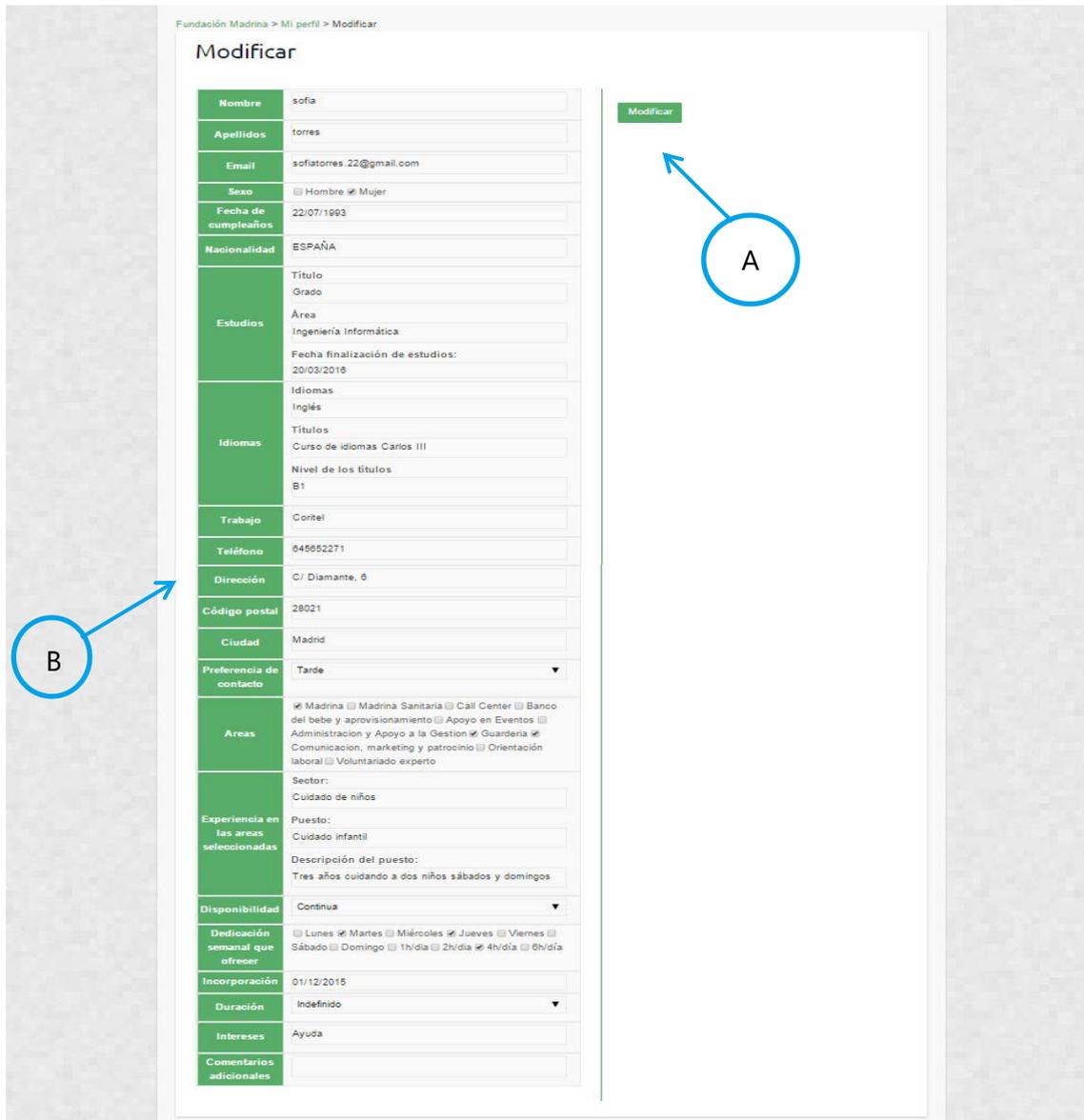


Ilustración 23: Pantalla Modificar Perfil

La página de modificar perfil, permite al voluntario almacenar en la base de datos los cambios deseados en sus datos de usuario. Consta de los siguientes elementos:

MODIFICAR MI PERFIL			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Modificar	Formulario	Almacena los nuevos datos del usuario en la base de datos
B	Tabla modificar perfil	Tabla	-

Tabla 85: Pantalla Modificar Perfil

Mis tareas



Ilustración 24: Pantalla Mis Tareas

La pantalla de mis tareas, muestra al usuario las tareas que debe realizar, y las que ha realizado. Contiene los siguientes elementos:

MIS TAREAS			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Tabla actividades en curso	Tabla	-
B	Tabla actividades terminadas	Tabla	-

Tabla 86: Pantalla Mis Tareas

Perfil trabajador

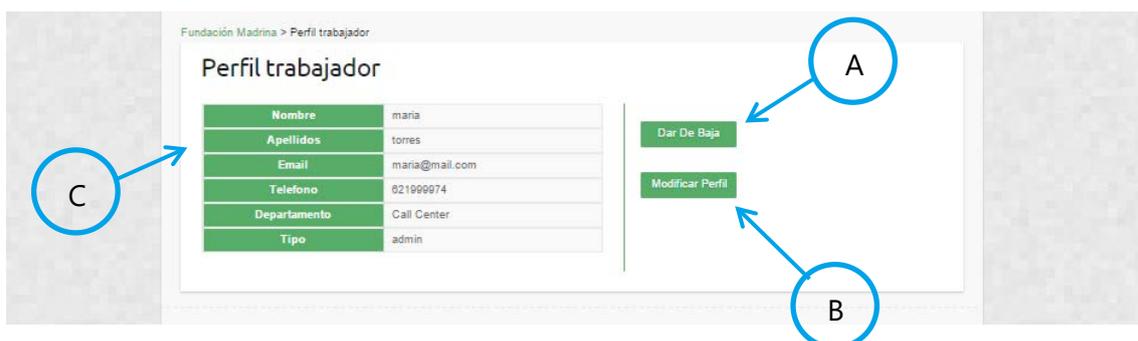


Ilustración 25: Pantalla Perfil Trabajador

La pantalla de perfil trabajador permite a los trabajadores ver su información de perfil, editarla y eliminarse como trabajador. Consta de los siguientes elementos:

PERFIL TRABAJADOR			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Dar de baja	Botón	Elimina el trabajador de la base de datos
B	Modificar perfil	Botón	Enlace con la página Modificar mi perfil
C	Tabla perfil trabajador	Tabla	-

Tabla 87: Pantalla Perfil Trabajador

Modificar perfil trabajador



Ilustración 26: Pantalla Modificar Perfil trabajador

La página de modificar perfil, permite al trabajador almacenar en la base de datos los cambios deseados en sus datos de usuario. Consta de los siguientes elementos:

MODIFICAR PERFIL			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Modificar	Formulario	Almacena los nuevos datos del trabajador en la base de datos
B	Tabla modificar trabajador	Tabla	-

Tabla 88: Pantalla Modificar Perfil trabajador

Añadir trabajador

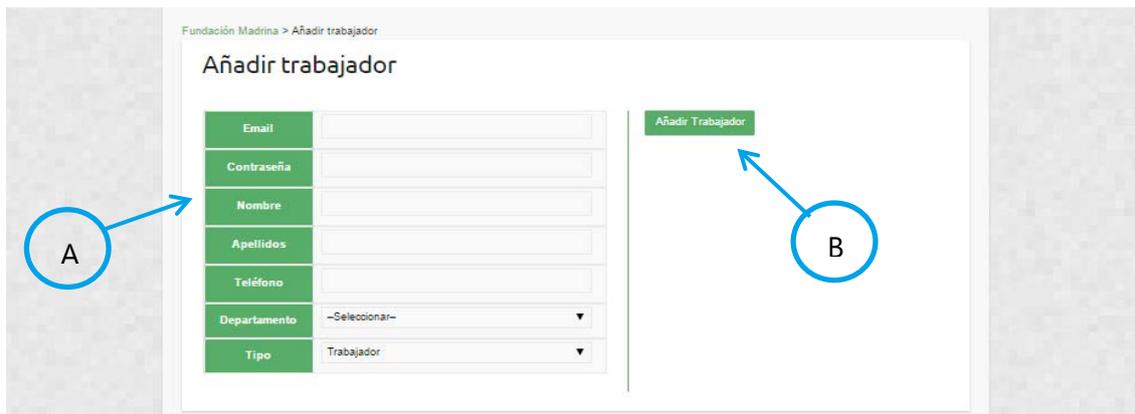


Ilustración 27: Pantalla Añadir Trabajador

La pantalla de añadir trabajador permite a un administrador insertar un nuevo trabajador de la fundación. Consta de los siguientes elementos:

AÑADIR TRABAJADOR			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Añadir trabajador	Formulario	Añade un nuevo trabajador a la base de datos
B	Tabla añadir trabajador	Tabla	-

Tabla 89: Pantalla Añadir Trabajador

Asignación de tareas



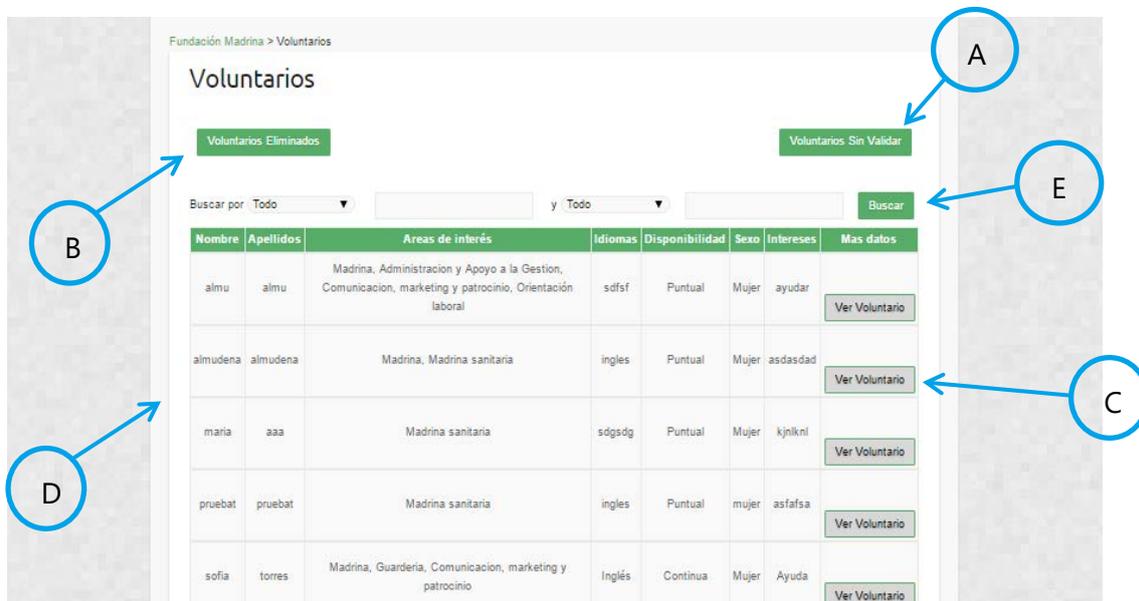
Ilustración 28: Pantalla Asignación de Tareas

La pantalla de asignación de tareas, muestra el listado de las tareas que están en curso y que voluntarios las realizan, así como las que ya han finalizado con los voluntarios que las realizaron. Contiene los siguientes elementos:

ASIGNACIÓN DE TAREAS			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Tabla actividades en curso	Tabla	-
B	Tabla actividades terminadas	Tabla	-

Tabla 90: Pantalla Asignación de Tareas

Ver voluntarios



The screenshot shows the 'Fundación Madrina > Voluntarios' page. It features a search bar with dropdown menus and a 'Buscar' button. Below the search bar is a table of volunteers with columns for Name, Surnames, Areas of Interest, Languages, Availability, Gender, Interests, and More Data. Each row has a 'Ver Voluntario' button. Callouts A through E point to specific elements: A points to the 'Voluntarios Sin Validar' button, B points to the search dropdowns, C points to a 'Ver Voluntario' button, D points to the table header, and E points to the 'Buscar' button.

Ilustración 29: Pantalla Ver Voluntarios

En la pantalla voluntarios, se muestra una tabla con el listado de voluntarios activos, así como un buscador para facilitar la búsqueda de un voluntario con unas determinadas características. Esta pantalla está compuesta de los siguientes elementos:

VOLUNTARIOS			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Voluntarios eliminados	Botón	Enlace con la página Voluntarios eliminados
B	Voluntarios sin validar	Botón	Enlace con la página Voluntarios sin validar
C	Ver voluntario	Botón	Enlace con la página Ver voluntario
D	Tabla voluntarios	Tabla	-
E	Buscar	Buscador	Busca por parámetros

Tabla 91: Pantalla Voluntarios

Voluntarios eliminados



Ilustración 30: Pantalla Voluntarios Eliminados

En la pantalla voluntarios eliminados, se muestra una tabla con el listado de voluntarios eliminados, así como un buscador para facilitar la búsqueda de un voluntario con unas determinadas características. Esta pantalla está compuesta de los siguientes elementos:

VOLUNTARIOS ELIMINADOS			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Buscar	Buscador	Busca por parámetros de voluntario
B	Ver voluntario	Botón	Enlace con la página Ver voluntario
C	Tabla voluntarios eliminados	Tabla	-

Tabla 92: Pantalla Voluntarios Eliminados

Voluntarios sin validar



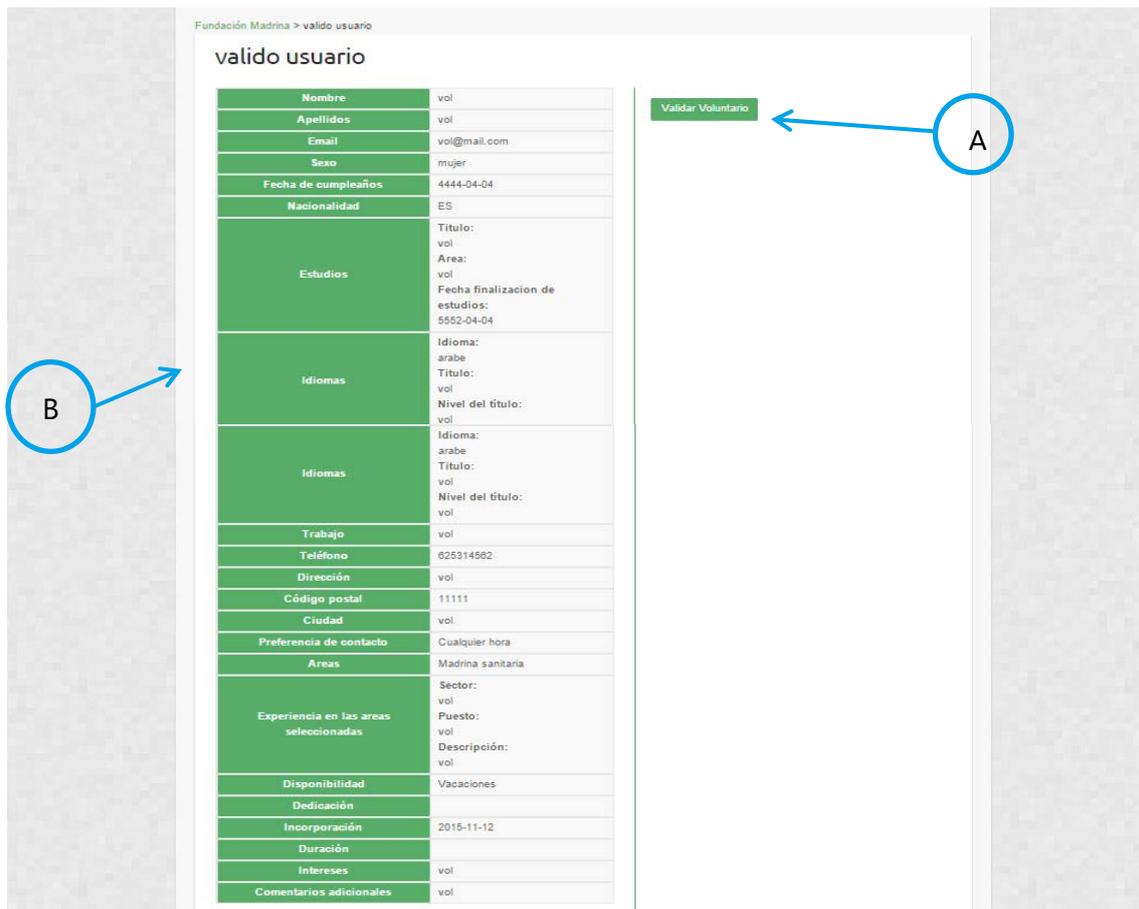
Ilustración 31: Pantalla Voluntarios Sin Validar

En la pantalla voluntarios sin validar, se muestra una tabla con el listado de voluntarios sin validar. Esta pantalla está compuesta de los siguientes elementos:

VOLUNTARIOS SIN VALIDAR			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Validar voluntario	Botón	Enlace con la página Validar voluntario
B	Tabla voluntarios sin validar	Tabla	-

Tabla 93: Pantalla Voluntarios Sin Validar

Validar voluntario



Fundación Madrina > valido usuario

valido usuario

Nombre	vol
Apellidos	vol
Email	vol@mail.com
Sexo	mujer
Fecha de cumpleaños	4444-04-04
Nacionalidad	ES
Estudios	Título: vol Área: vol Fecha finalizacion de estudios: 5552-04-04
Idiomas	Idioma: arabe Título: vol Nivel del título: vol
Idiomas	Idioma: arabe Título: vol Nivel del título: vol
Trabajo	vol
Teléfono	825314562
Dirección	vol
Código postal	11111
Ciudad	vol
Preferencia de contacto	Cualquier hora
Áreas	Madrina sanitaria
Experiencia en las áreas seleccionadas	Sector: vol Puesto: vol Descripción: vol
Disponibilidad	Vacaciones
Dedicación	
Incorporación	2015-11-12
Duración	
Intereses	vol
Comentarios adicionales	vol

Validar Voluntario

Ilustración 32: Pantalla Validar Voluntario

En la pantalla validar voluntario, muestra los datos de un nuevo voluntario para comprobar si cumple con los requisitos establecidos, si es así el administrador lo procede a validarlo para que el usuario tenga acceso al sistema:

VALIDAR VOLUNTARIO			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Validar voluntario	Botón	Valida el voluntario introduciéndolo en la base de datos
B	Tabla validar voluntario	Tabla	-

Tabla 94: Pantalla Validar Voluntario

Detalles voluntario

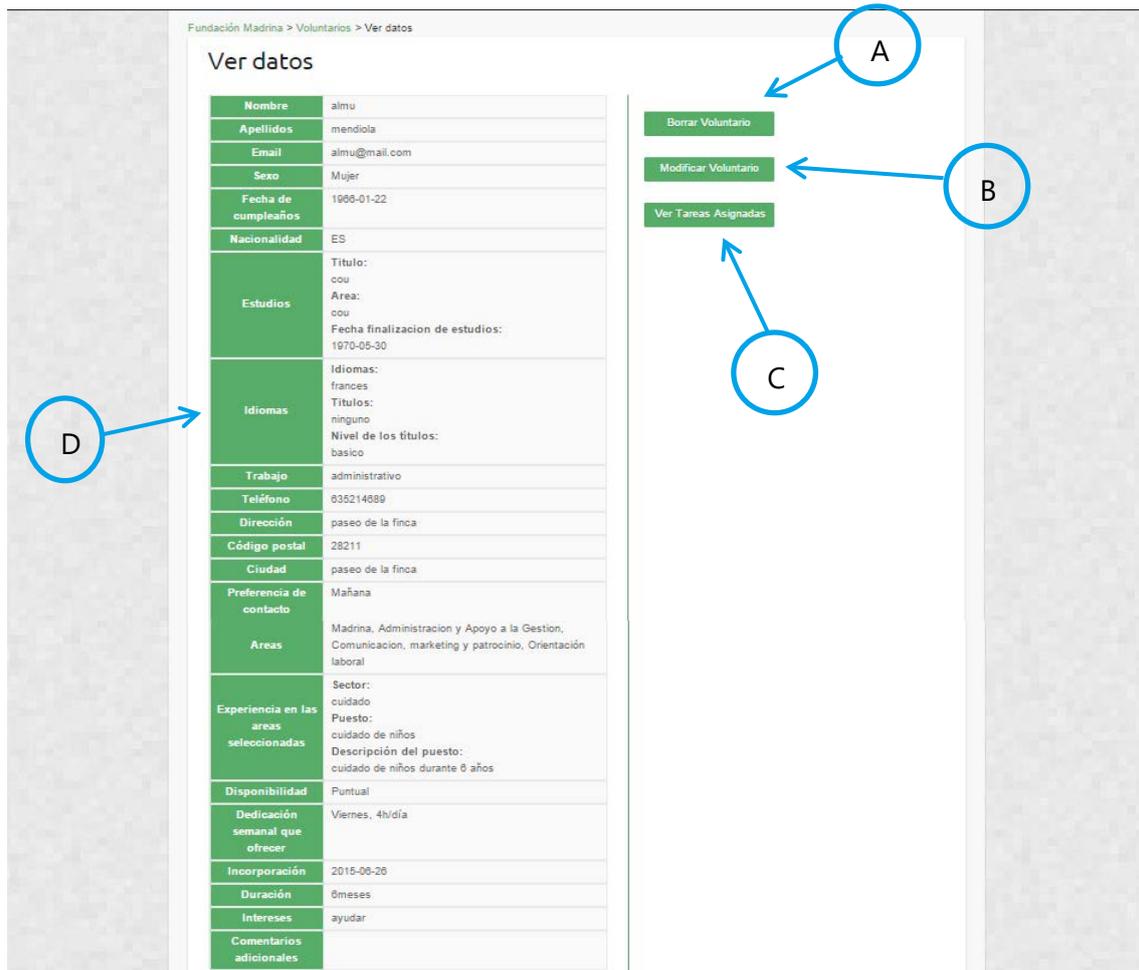


Ilustración 33: Pantalla Detalles Voluntario

En la pantalla detalles voluntario, muestra una tabla con la información detallada de un voluntario. Esta pantalla está compuesta de los siguientes elementos:

DETALLES VOLUNTARIO			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Borrar voluntario	Botón	Elimina el voluntario de la base de datos
B	Modificar voluntario	Botón	Enlace con la página Modificar voluntario
C	Ver tareas asignadas	Botón	Enlace con la página Ver tareas asignadas
D	Tabla detalles voluntario	Tabla	-

Tabla 95: Pantalla Detalles Voluntario

Modificar voluntario

Fundación Madrina > Voluntarios > Ver datos > Modificar voluntario

Modificar voluntario

Nombre	almu
Apellidos	mendiola
Email	almu@mail.com
Sexo	<input type="checkbox"/> Hombre <input checked="" type="checkbox"/> Mujer
Fecha de cumpleaños	22/01/1988
Nacionalidad	ES
Estudios	Título: cou Área: cou Fecha finalización de estudios: 30/05/1970
Idiomas	Idiomas: frances Títulos: ninguno Nivel de los títulos: basico
Trabajo	administrativo
Teléfono	635214689
Dirección	paseo de la finca
Código postal	28211
Ciudad	madrid
Preferencia de contacto	Mañana
Areas	<input checked="" type="checkbox"/> Madrina <input type="checkbox"/> Madrina Sanitaria <input type="checkbox"/> Call Center <input type="checkbox"/> Banco del bebe y aprovisionamiento <input type="checkbox"/> Apoyo en Eventos <input checked="" type="checkbox"/> Administración y Apoyo a la Gestion <input checked="" type="checkbox"/> Guardería <input checked="" type="checkbox"/> Comunicación, marketing y patrocinio <input checked="" type="checkbox"/> Orientación laboral <input type="checkbox"/> Voluntariado experto
Experiencia en las areas seleccionadas	Sector: cuidado Puesto: cuidado de niños Descripción del puesto: cuidado de niños durante 0 años
Disponibilidad	Puntual
Dedicación semanal que ofrecer	<input type="checkbox"/> Lunes <input type="checkbox"/> Martes <input type="checkbox"/> Miércoles <input type="checkbox"/> Jueves <input checked="" type="checkbox"/> Viernes <input type="checkbox"/> Sábado <input type="checkbox"/> Domingo <input type="checkbox"/> 1h/día <input type="checkbox"/> 2h/día <input checked="" type="checkbox"/> 4h/día <input type="checkbox"/> 8h/día
Incorporación	28/06/2015
Duración	6 Meses
Intereses	ayudar
Comentarios adicionales	

Modificar

B → (Form fields)

← **A** (Modificar button)

Ilustración 34: Pantalla Modificar voluntario

En la pantalla modificar voluntario muestra una tabla con los datos que tiene almacenados en la base de datos, el trabajador realiza los cambios deseados y se almacenan de nuevo en la base de datos. Esta pantalla contiene los siguientes elementos:

MODIFICAR VOLUNTARIO			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Modificar	Formulario	Almacena los nuevos datos del voluntario en la base de datos
B	Tabla modificar voluntario	Tabla	-

Tabla 96: Pantalla Modificar Voluntario

Ver actividades de un voluntario



Ilustración 35: Ver actividades de un Voluntario

La pantalla de ver actividades de un voluntario, muestra a un trabajador las tareas que debe realizar este usuario, y las que ha realizado. Consta de los siguientes elementos:

VER ACTIVIDADES DE UN VOLUNTARIO			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Tabla actividades en curso	Tabla	-
B	Tabla actividades terminadas	Tabla	-

Tabla 97: Pantalla Ver actividades de un Voluntario

Ver actividades

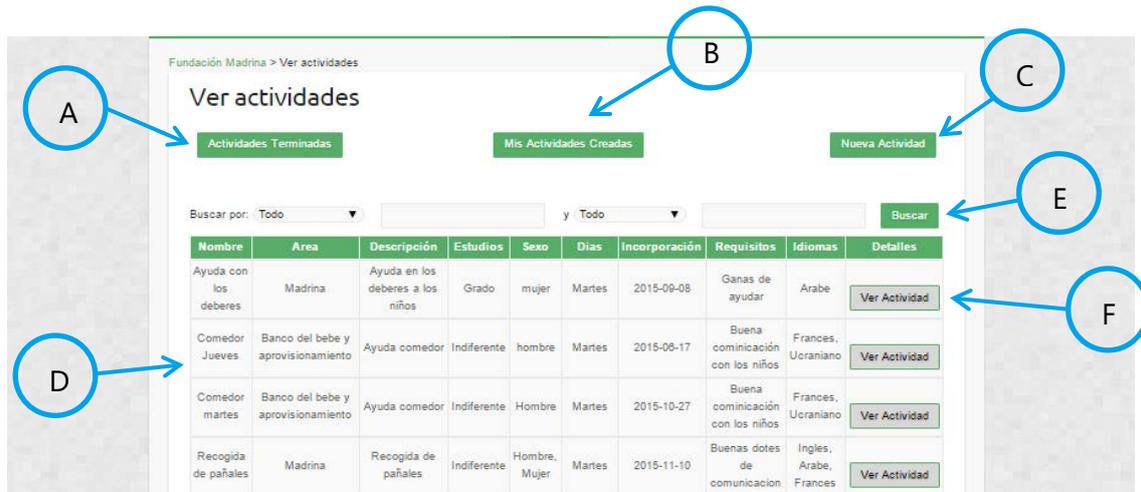


Ilustración 36: Pantalla Ver Actividades

La pantalla ver actividades muestra las actividades activas que hay almacenadas en la base de datos, así como un buscador para facilitar el la búsqueda a un trabajador. Está compuesta de los siguientes elementos:

VER ACTIVIDADES			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Actividades terminadas	Botón	Enlace con la página Ver actividades terminadas
B	Ver actividades creadas	Botón	Enlace con la página Ver actividades creadas
C	Nueva actividad	Botón	Enlace con la página Nueva actividad
D	Tabla de actividades	Tabla	-
E	Buscar	Buscador	Busca por parámetros de actividad
F	Ver detalles	Botón	Enlace con la página Ver detalles

Tabla 98: Pantalla Ver Actividades

Detalles de actividad



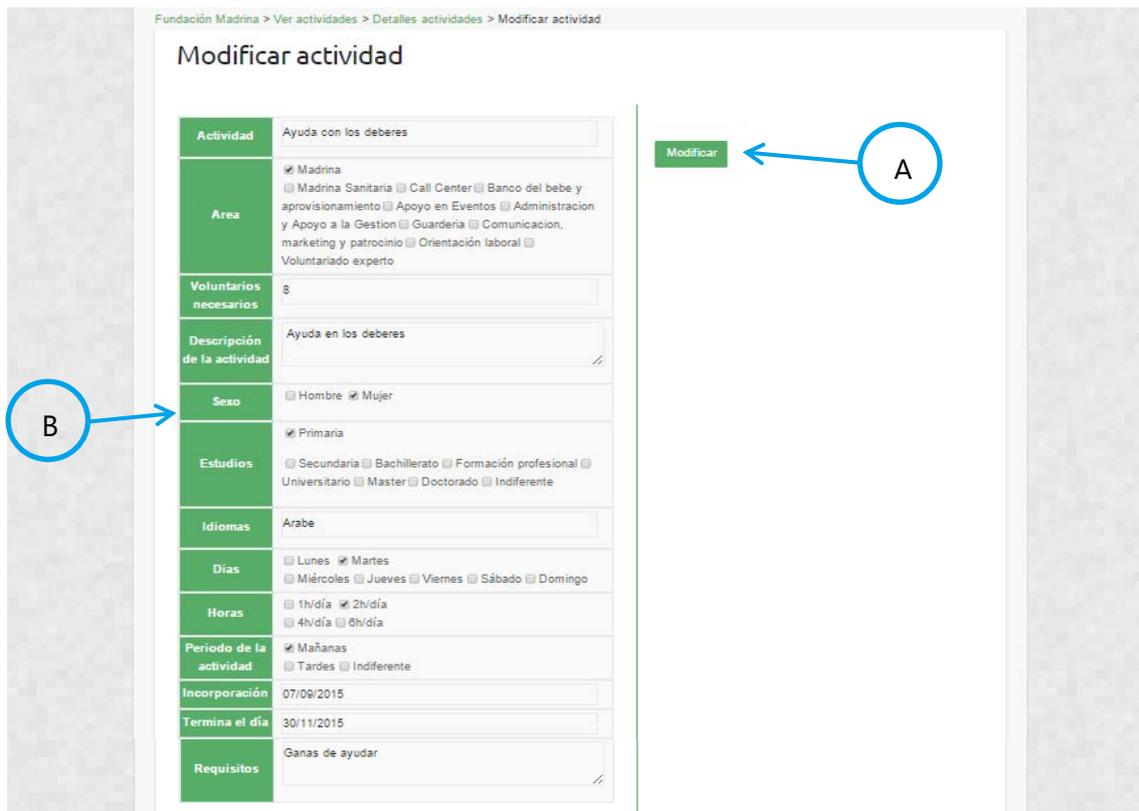
Ilustración 37: Pantalla Detalles de Actividad

La pantalla detalles de actividad muestra una tabla con los datos de una actividad. Consta de los siguientes elementos:

DETALLES ACTIVIDAD			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Borrar actividad	Botón	Borra la actividad de la base de datos
B	Modificar actividad	Botón	Enlace con la página Modificar actividad
C	Asignar actividad	Botón	Enlace con la página Asignar actividad
D	Ver voluntarios asignados	Botón	Enlace con la página Ver voluntarios asignados
E	Tabla detalles actividad	Tabla	-

Tabla 99: Pantalla Detalles actividad

Modificar actividad



Fundación Madrina > Ver actividades > Detalles actividades > Modificar actividad

Modificar actividad

Actividad: Ayuda con los deberes

Area: Madrina Madrina Sanitaria Call Center Banco del bebe y aprovisionamiento Apoyo en Eventos Administración y Apoyo a la Gestión Guardería Comunicación, marketing y patrocinio Orientación laboral Voluntariado experto

Voluntarios necesarios: 8

Descripción de la actividad: Ayuda en los deberes

Sexo: Hombre Mujer

Estudios: Primaria Secundaria Bachillerato Formación profesional Universitario Master Doctorado Indiferente

Idiomas: Árabe

Días: Lunes Martes Miércoles Jueves Viernes Sábado Domingo

Horas: 1h/día 2h/día 4h/día 6h/día

Periodo de la actividad: Mañanas Tardes Indiferente

Incorporación: 07/09/2015

Termina el día: 30/11/2015

Requisitos: Ganas de ayudar

Modificar

Ilustración 38: Pantalla Modificar Actividad

La pantalla de modificar actividad permite al trabajador hacer cambios en los datos de una actividad. Consta de los siguientes elementos:

MODIFICAR ACTIVIDAD			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Modificar	Formulario	Almacena los nuevos datos del trabajador en la base de datos
B	Tabla modificar actividad	Tabla	-

Tabla 100: Pantalla Modificar Actividad

Asignar actividad

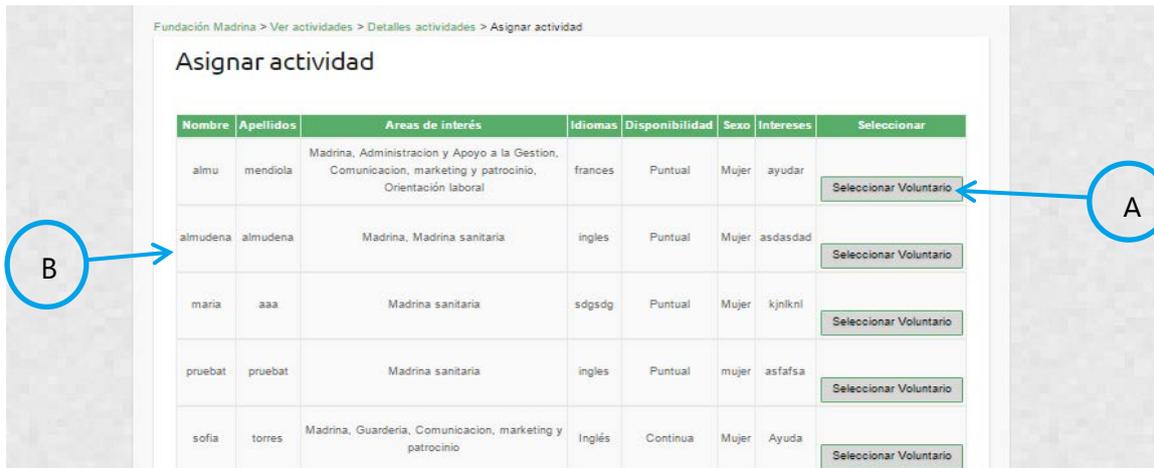


Ilustración 39: Pantalla Asignar Actividad

La pantalla asignar actividad, permite asignar una actividad a un voluntario. Consta de los siguientes elementos:

ASIGNAR ACTIVIDAD			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Seleccionar voluntario	Formulario	Almacena en la base de datos la actividad asignada al usuario seleccionado
B	Tabla asignar actividad	Tabla	-

Tabla 101: Pantalla Asignar Actividad

Ver voluntarios asignados a una actividad



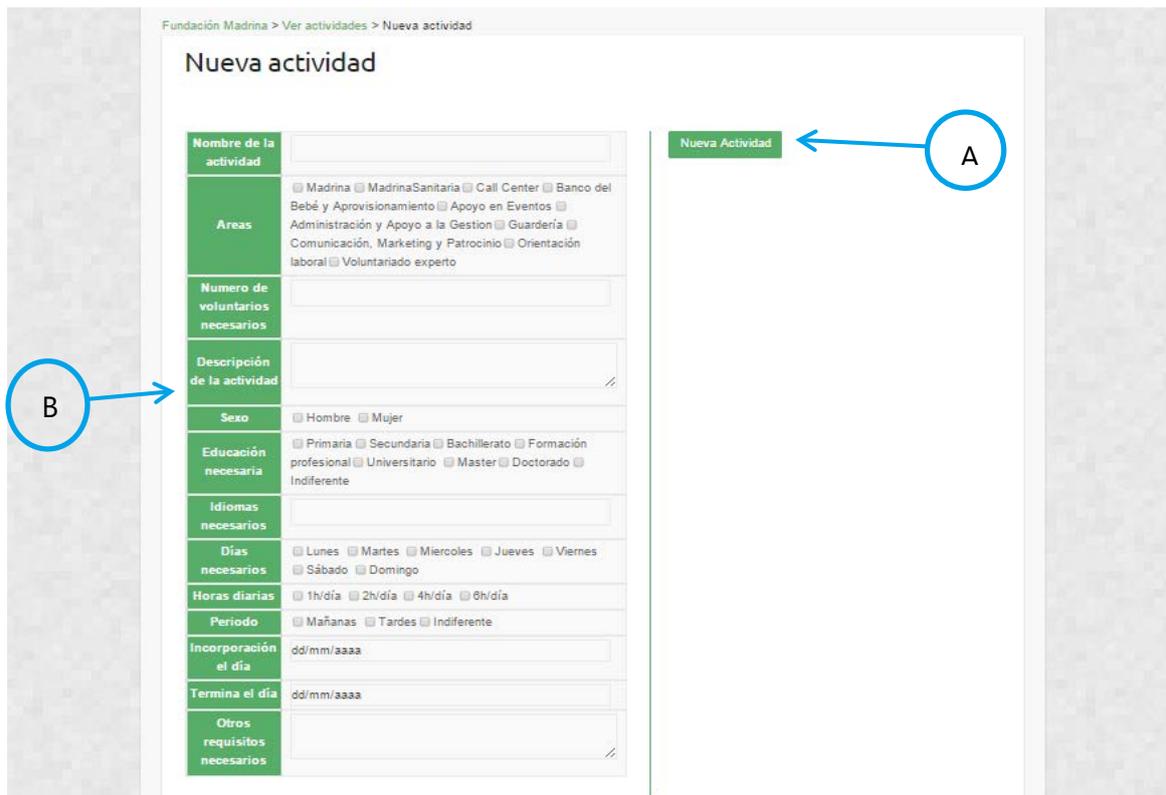
Ilustración 40: Pantalla Ver voluntarios Asignados a una Actividad

La pantalla de ver voluntarios asignados a una actividad, muestra el listado de voluntarios que tiene asignados una actividad. Consta de los siguientes elementos:

VER VOLUNTARIOS ASIGNADOS A UNA ACTIVIDAD			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Tabla asignar actividad	Tabla	-

Tabla 102: Pantalla Ver Voluntarios Asignados a una Actividad

Nueva actividad



Fundación Madrina > Ver actividades > Nueva actividad

Nueva actividad

Nombre de la actividad

Áreas Madrina MadrinaSanitaria Call Center Banco del Bebé y Aprovisionamiento Apoyo en Eventos Administración y Apoyo a la Gestión Guardería Comunicación, Marketing y Patrocinio Orientación laboral Voluntariado experto

Numero de voluntarios necesarios

Descripción de la actividad

Sexo Hombre Mujer

Educación necesaria Primaria Secundaria Bachillerato Formación profesional Universitario Master Doctorado Indiferente

Idiomas necesarios

Días necesarios Lunes Martes Miércoles Jueves Viernes Sábado Domingo

Horas diarias 1h/día 2h/día 4h/día 6h/día

Periodo Mañanas Tardes Indiferente

Incorporación el día dd/mm/aaaa

Termina el día dd/mm/aaaa

Otros requisitos necesarios

Nueva Actividad

Ilustración 41: Pantalla Nueva Actividad

La pantalla de nueva actividad permite al trabajador añadir una nueva actividad en la base de datos. Esta pantalla está formada por los siguientes elementos:

NUEVA ACTIVIDAD			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Nueva actividad	Formulario	Almacena una nueva actividad en la base de datos
B	Tabla nueva actividad	Tabla	-

Tabla 103: Pantalla Nueva Actividad

Mis actividades creadas



Fundación Madrina > Ver actividades > Actividades creadas

Actividades creadas

Nombre	Area	Descripción	Estudios	Sexo	Días	Incorporación	Requisitos	Idiomas	Detalles
Ayuda a recogida de alimentos	Madrina, Madrina sanitaria	Evento para recaudación de alimentos y material sanitario	Secundaria, Bachillerato, Universitario	Hombre, Mujer	Martes, Miércoles, Jueves	2015-03-29	Facilidad de comunicación con otros usuarios	Frances	<input type="button" value="Ver Actividad"/>
Ayuda para recogida de pañales	Madrina	Recogida de pañales	Indiferente	Hombre, Mujer	Martes	2015-11-10	Buenas dotes de comunicacion	Inglés, Árabe, Frances	<input type="button" value="Ver Actividad"/>

Ilustración 42: Pantalla Mis Actividades Creadas

La pantalla de mis actividades creadas muestra un listado con las actividades que ha creado un trabajador. Esta pantalla está formada por los siguientes elementos:

MIS ACTIVIDADES CREADAS			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Ver actividad	Botón	Enlace con la página ver detalles actividad
B	Tabla actividades creadas	Tabla	-

Tabla 104: Pantalla Mis Actividades Creadas

Actividades terminadas



Ilustración 43: Pantalla Actividades Terminadas

La pantalla actividades terminadas, muestra una tabla con las actividades que ya han finalizado, así como un buscador para realizar consultas. Consta de los siguientes elementos:

ACTIVIDADES TERMINADAS			
ID	NOMBRE	TIPO	MÉTODOS IMPLICADOS ONCLICK
A	Buscar	Buscador	Busca por parámetros de actividad
B	Tabla actividades creadas	Tabla	-
C	Ver actividad	Botón	Enlace con la página ver detalles actividad

Tabla 105: Pantalla Actividades Terminadas

6.3 Diseño de clases

Un buen diseño de clases determinará en gran medida la estructura de la aplicación y su sencillez a la hora de realizar cambios en el futuro.

CLASE DE ACCESO A LA INFORMACIÓN	
DATABASE	Implementa funcionalidades relacionada con la persistencia de datos.

Tabla 106: Clase de acceso a la información

CLASE LÓGICA DE VOLUNTARIOS	
VOLUNTARIES	Implementa la funcionalidad de ver, editar y borrar un voluntario.
TABLEUSER	Implementa la construcción de una tabla de visualización de voluntarios.
TABLEACTIVITIES	Implementa la construcción de una tabla con las actividades asignadas.

Tabla 107: Clase lógica de voluntarios

CLASE LÓGICA DE TRABAJADORES	
WORKERS	Implementa la funcionalidad de ver, editar y borrar un trabajador.
TABLEWORKER	Implementa la construcción de una tabla de visualización de trabajadores.
TABLEACTIVITIES	Implementa la construcción de una tabla con la visualización de actividades creadas.

Tabla 108: Clase lógica de trabajadores

CLASE LÓGICA DE ACTIVIDADES	
ACTIVITIES	Implementa la funcionalidad de una actividad.
TABLEACTIVITIES	Implementa la construcción de una tabla de visualización de actividades.
TABLEUSER	Implementa la construcción de una tabla con los usuarios asignados.

Tabla 109: Clase lógica de actividades

CLASE LÓGICA DE ASIGNACIONES	
ACTIVITIESUSER	Implementa la construcción de una tabla que visualiza todos los voluntarios y actividades asignados.

Tabla 110: Clase lógica de asignaciones

Además cada clase contendrá uno o varios métodos que son necesarios para el desempeño de las funciones que realizan:

DataBase	
<i>getUser(id_voluntary)</i>	
DESCRIPCIÓN	Obtiene el usuario con ese identificador.
ARGUMENTOS	Identificador del voluntario.
FUNCIONAMIENTO	Recupera de la base de datos el voluntario que tenga ese identificador.
RETORNO	Devuelve el objeto voluntario con ese identificador.

<i>getActivity(id_activity)</i>	
DESCRIPCIÓN	Obtiene la actividad con ese identificador.
ARGUMENTOS	Identificador de la actividad.
FUNCIONAMIENTO	Recupera de la base de datos la actividad que tenga ese identificador.
RETORNO	Devuelve el objeto actividad con ese identificador.
<i>getActivityUser(id)</i>	
DESCRIPCIÓN	Obtiene la actividad asignada a un voluntario con ese identificador.
ARGUMENTOS	Identificador de la asignación.
FUNCIONAMIENTO	Recupera de la base de datos la actividad y voluntario que tiene esa asignación.
RETORNO	Devuelve el objeto actividad y usuario asignados.
<i>getStudies(email)</i>	
DESCRIPCIÓN	Obtiene los estudios con ese identificador.
ARGUMENTOS	Identificador de los estudios.
FUNCIONAMIENTO	Recupera de la base de datos los estudios con ese identificador.
RETORNO	Devuelve el objeto estudios con ese identificador.
<i>getLanguages(email)</i>	
DESCRIPCIÓN	Obtiene los idiomas con ese identificador.
ARGUMENTOS	Identificador de los idiomas.
FUNCIONAMIENTO	Recupera de la base de datos los idiomas con ese identificador.
RETORNO	Devuelve el objeto idiomas con ese identificador.
<i>getExperience(email)</i>	
DESCRIPCIÓN	Obtiene la experiencia con ese identificador.
ARGUMENTOS	Identificador de experiencia.
FUNCIONAMIENTO	Recupera de la base de datos la experiencia con ese identificador.
RETORNO	Devuelve el objeto experiencia con ese identificador.
<i>getUserValidate(id_voluntary)</i>	
DESCRIPCIÓN	Obtiene el usuario con ese identificador.
ARGUMENTOS	Identificador del voluntario.
FUNCIONAMIENTO	Recupera de la base de datos el voluntario que tenga ese identificador.
RETORNO	Devuelve el objeto voluntario con ese identificador.
<i>setUser(user)</i>	
DESCRIPCIÓN	Almacena un nuevo voluntario en la base de datos.
ARGUMENTOS	Los parámetros del voluntario a almacenar.
FUNCIONAMIENTO	Recoge los datos e inserta un nuevo voluntario con los parámetros deseados.
RETORNO	-
<i>setActivity(activity)</i>	
DESCRIPCIÓN	Almacena una nueva actividad en la base de datos.
ARGUMENTOS	Los parámetros de la actividad a almacenar.
FUNCIONAMIENTO	Recoge los datos e inserta una nueva actividad con los parámetros deseados.
RETORNO	-
<i>setStudies(studies)</i>	
DESCRIPCIÓN	Almacena unos nuevos estudios en la base de datos.
ARGUMENTOS	Los parámetros de los estudios a almacenar.
FUNCIONAMIENTO	Recoge los datos e inserta los nuevos estudios con los parámetros deseados.
RETORNO	-

<i>setLanguages(language)</i>	
DESCRIPCIÓN	Almacena unos nuevos idiomas en la base de datos.
ARGUMENTOS	Los parámetros de los idiomas a almacenar.
FUNCIONAMIENTO	Recoge los datos e inserta los nuevos idiomas con los parámetros deseados
RETORNO	-
<i>setExperience(experience)</i>	
DESCRIPCIÓN	Almacena una nueva experiencia en la base de datos.
ARGUMENTOS	Los parámetros de la experiencia a almacenar.
FUNCIONAMIENTO	Recoge los datos e inserta la nueva experiencia con los parámetros deseados.
RETORNO	-
<i>setUserValidate(language)</i>	
DESCRIPCIÓN	Almacena un nuevo voluntario en la base de datos.
ARGUMENTOS	Los parámetros del voluntario a almacenar.
FUNCIONAMIENTO	Recoge los datos e inserta un nuevo voluntario con los parámetros deseados.
RETORNO	-
<i>setActivityUser(id_activity, id_voluntary)</i>	
DESCRIPCIÓN	Almacena una nueva asignación en la base de datos.
ARGUMENTOS	El identificador de la actividad y del voluntario.
FUNCIONAMIENTO	Inserta en la base de datos un nuevo registro con la actividad y el usuario.
RETORNO	-
<i>getSearchUser(String atributo, String valor)</i>	
DESCRIPCIÓN	Recupera los datos cuyo atributo sea igual al valor dado.
ARGUMENTOS	Nombre del valor que desea comparar y dicho valor.
FUNCIONAMIENTO	Recupera de la base de datos todos los voluntarios si cumplen que el atributo que se ha pasado como parámetro tenga el valor deseado.
RETORNO	Lista de voluntarios.
<i>getSearchActivities(String atributo, String valor)</i>	
DESCRIPCIÓN	Recupera los datos cuyo atributo sea igual al valor dado.
ARGUMENTOS	Nombre del valor que desea comparar y dicho valor.
FUNCIONAMIENTO	Recupera de la base de datos todas las actividades si cumplen que el atributo que se ha pasado como parámetro tenga el valor deseado.
RETORNO	Lista de actividades.
<i>getUserNoValidate()</i>	
DESCRIPCIÓN	Recupera todos los voluntarios que no han sido validados.
ARGUMENTOS	-
FUNCIONAMIENTO	Recupera de la base de datos todos los voluntarios no validados.
RETORNO	Lista de voluntarios no validados.
<i>getActivitiesUser()</i>	
DESCRIPCIÓN	Recupera todas las actividades con voluntarios asignados y dichos voluntarios.
ARGUMENTOS	-
FUNCIONAMIENTO	Recuperar todas las actividades con voluntarios asignados y dichos voluntarios.
RETORNO	Lista de actividades con voluntarios asignados.
<i>removeUser(id_voluntary)</i>	
DESCRIPCIÓN	Eliminar un voluntario.
ARGUMENTOS	Identificador del voluntario a eliminar.

FUNCIONAMIENTO	Elimina el voluntario con el identificador pasado como parámetro de la base de datos.
RETORNO	-
<i>removeActivity(id_activity)</i>	
DESCRIPCIÓN	Eliminar una actividad.
ARGUMENTOS	Identificador de la actividad a eliminar.
FUNCIONAMIENTO	Elimina la actividad de la base de datos con el identificador dado.
RETORNO	-
<i>getWorker(id_worker)</i>	
DESCRIPCIÓN	Obtiene el trabajador con ese identificador.
ARGUMENTOS	Identificador del trabajador.
FUNCIONAMIENTO	Recupera de la base de datos el trabajador que tenga ese identificador.
RETORNO	Devuelve el objeto trabajador con ese identificador.
<i>setWorker(id_worker)</i>	
DESCRIPCIÓN	Almacena un nuevo trabajador en la base de datos.
ARGUMENTOS	Los parámetros del trabajador a almacenar.
FUNCIONAMIENTO	Recoge los datos e inserta un nuevo trabajador con los parámetros dados.
RETORNO	-
<i>removeWorker(id_worker)</i>	
DESCRIPCIÓN	Eliminar un trabajador.
ARGUMENTOS	Identificador del trabajador a eliminar.
FUNCIONAMIENTO	Elimina el trabajador con el identificador pasado como parámetro de la base de datos.
RETORNO	-

Tabla 111: Clase DataBase

Clase Voluntaries	
<i>voluntary()</i>	
DESCRIPCIÓN	Conforma toda la vista del voluntario.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve los datos del voluntario.
<i>modifyVoluntary()</i>	
DESCRIPCIÓN	Conforma toda la vista de modificar del voluntario.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve los datos a modificar del voluntario.
<i>activitiesUser()</i>	
DESCRIPCIÓN	Conforma toda la vista de actividades de un usuario.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve las actividades que tiene el voluntario.
<i>allUser()</i>	
DESCRIPCIÓN	Conforma toda la vista de voluntarios.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve una lista con todos los voluntarios.

Tabla 112: Clase Voluntaries

Clase Activities	
<i>activity()</i>	
DESCRIPCIÓN	Conforma toda la vista de una actividad.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve los datos de la actividad.
<i>modifyActivity()</i>	
DESCRIPCIÓN	Conforma toda la vista de modificar la actividad.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve los datos a modificar de la actividad.
<i>userActivities()</i>	
DESCRIPCIÓN	Conforma toda la vista de usuarios de una actividad.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve los usuarios que tiene una actividad.
<i>allActivities()</i>	
DESCRIPCIÓN	Conforma toda la vista de actividades.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve una lista con todas las actividades.

Tabla 113: Clase Activities

Clase Workers	
<i>worker()</i>	
DESCRIPCIÓN	Conforma toda la vista de un trabajador.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve los datos de un trabajador.
<i>modifyWorker()</i>	
DESCRIPCIÓN	Conforma toda la vista de modificar trabajador.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve los datos a modificar del trabajador.

Tabla 114: Clase Workers

Clase ActivitiesUser	
<i>activityUserAll()</i>	
DESCRIPCIÓN	Conforma toda la vista actividades y voluntarios asignados.
ARGUMENTOS	-
FUNCIONAMIENTO	Hace llamadas a otros métodos para realizar una serie de tareas.
RETORNO	Devuelve los datos de las actividades y trabajadores que están asignados.

Tabla 115: Clase ActivitiesUser

6.4 Base de datos

A continuación se muestra el diseño de la base de datos que contiene 8 tablas, con su esquema relacional y una descripción de cada uno de los campos almacenados.

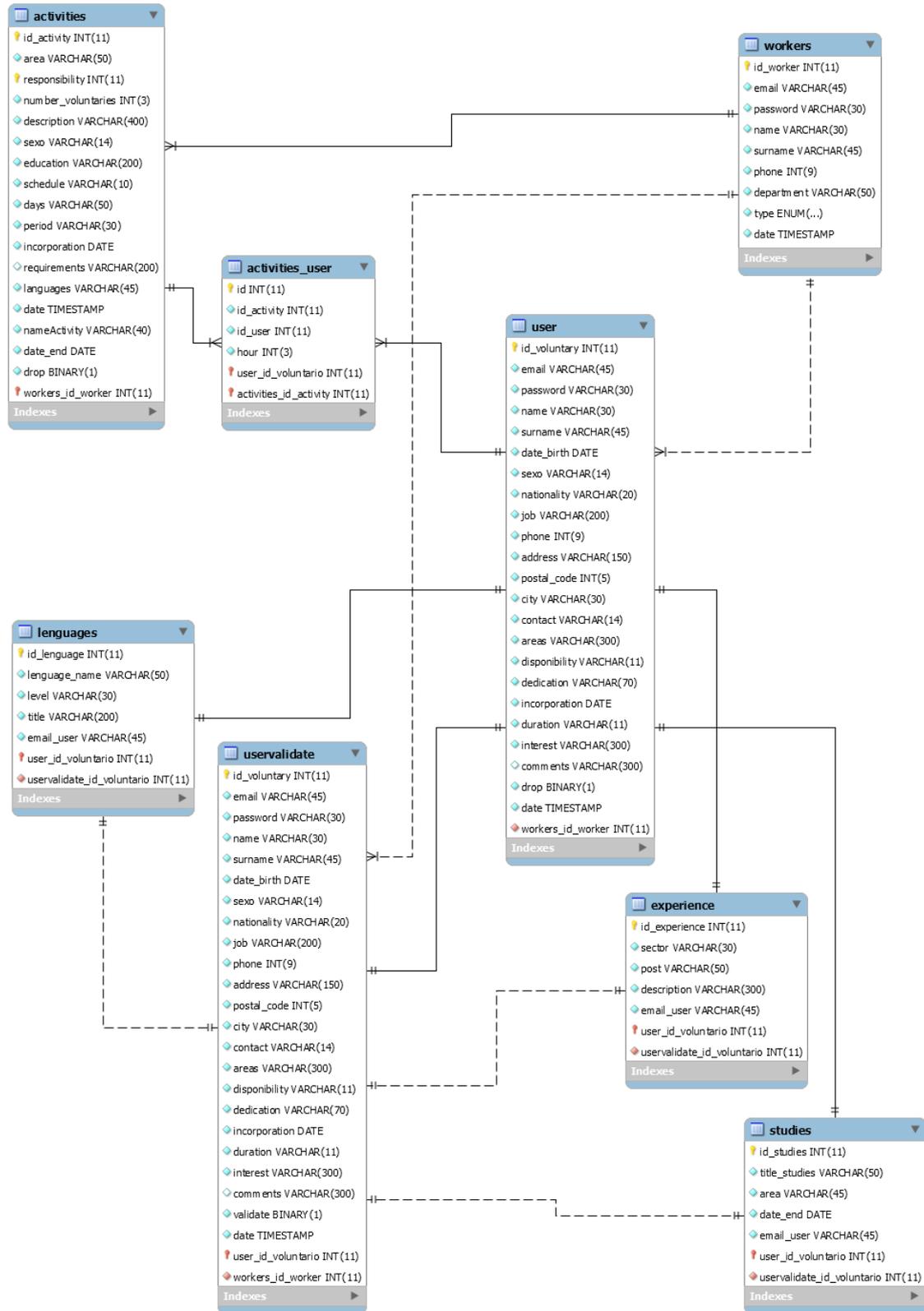


Ilustración 44: Base de datos



Tabla user				
<i>Atributo</i>	<i>Descripción</i>	<i>Tipo Atributo</i>	<i>Tipo Dato</i>	<i>Valores</i>
ID_VOLUNTARY	Identificador único de la tabla user	Clave primaria	Int(11)	0 a 2 ³¹ - 1
EMAIL	Email del usuario	Atributo no nulo	Varchar(45)	Email
PASSWORD	Contraseña del usuario	Atributo no nulo	Varchar(30)	Password
NAME	Nombre del usuario	Atributo no nulo	Varchar(30)	Nombre
SURNAME	Apellidos del usuario	Atributo no nulo	Varchar(45)	Apellidos
DATE_BIRTH	Fecha de cumpleaños	Atributo no nulo	Date	Fecha de cumpleaños
SEXO	Sexo	Atributo no nulo	Varchar(14)	Hombre, Mujer
NATIONALITY	Nacionalidad	Atributo no nulo	Varchar(20)	España, Francia, Marruecos...
JOB	Trabajo que tiene	Atributo no nulo	Varchar(200)	Descripción de los trabajos que ha realizado
PHONE	Número de teléfono	Atributo no nulo	Int(9)	Número de teléfono
ADDRESS	Dirección	Atributo no nulo	Varchar(150)	Dirección
POSTAL_CODE	Código postal	Atributo no nulo	Int(5)	Código postal
CITY	Ciudad	Atributo no nulo	Varchar(30)	Ciudad
CONTACT	Preferencia de contacto	Atributo no nulo	Varchar(14)	Mañana, Tarde, Cualquier hora
AREAS	Áreas en las que quiere ser voluntario	Atributo no nulo	Varchar(300)	Madrina, Madrina Sanitaria, Guardería...
DISPONIBILITY	Disponibilidad	Atributo no nulo	Varchar(11)	Puntual, Continua, Vacaciones
DEDICATION	Dedicación semanal	Atributo no nulo	Varchar(70)	Lunes, Martes, Miércoles, 1h/día...

INCORPORATION	Fecha de incorporación	Atributo no nulo	Date	Fecha de incorporación
DURATION	Duración	Atributo no nulo	Varchar(11)	1 Mes, 2 Meses...
INTEREST	Intereses de ser voluntario	Atributo no nulo	Varchar(300)	Descripción de los intereses de ser voluntario
COMMENTS	Comentarios adicionales	-	Varchar(300)	Comentarios que quiera añadir
DROP	Estado del usuario	Atributo no nulo	Binary(1)	Indica si está borrado o no el usuario
DATE	Fecha de registro	Atributo no nulo	Date	Fecha de registro

Tabla 116: Tabla user

Tabla uservalidate				
<i>Atributo</i>	<i>Descripción</i>	<i>Tipo Atributo</i>	<i>Tipo Dato</i>	<i>Valores</i>
ID_VOLUNTARY	Identificador único de la tabla uservalidate	Clave primaria	Int(11)	0 a $2^{31} - 1$
EMAIL	Email del usuario	Atributo no nulo	Varchar(45)	Email
PASSWORD	Contraseña del usuario	Atributo no nulo	Varchar(30)	Password
NAME	Nombre del usuario	Atributo no nulo	Varchar(30)	Nombre
SURNAME	Apellidos del usuario	Atributo no nulo	Varchar(45)	Apellidos
DATE_BIRTH	Fecha de cumpleaños	Atributo no nulo	Date	Fecha de cumpleaños
SEXO	Sexo	Atributo no nulo	Varchar(14)	Hombre, Mujer
NATIONALITY	Nacionalidad	Atributo no nulo	Varchar(20)	España, Francia, Marruecos...
JOB	Trabajo que tiene	Atributo no nulo	Varchar(200)	Descripción de los trabajos que ha realizado
PHONE	Número de teléfono	Atributo no nulo	Int(9)	Número de teléfono
ADDRESS	Dirección	Atributo no nulo	Varchar(150)	Dirección



POSTAL_CODE	Código postal	Atributo no nulo	Int(5)	Código postal
CITY	Ciudad	Atributo no nulo	Varchar(30)	Ciudad
CONTACT	Preferencia de contacto	Atributo no nulo	Varchar(14)	Mañana, Tarde, Cualquier hora
AREAS	Áreas en las que quiere ser voluntario	Atributo no nulo	Varchar(300)	Madrina, Madrina Sanitaria, Guardería...
DISPONIBILITY	Disponibilidad	Atributo no nulo	Varchar(11)	Puntual, Continua, Vacaciones
DEDICATION	Dedicación semanal	Atributo no nulo	Varchar(70)	Lunes, Martes, Miércoles, 1h/día...
INCORPORATION	Fecha de incorporación	Atributo no nulo	Date	Fecha de incorporación
DURATION	Duración	Atributo no nulo	Varchar(11)	1 Mes, 2 Meses...
INTEREST	Intereses de ser voluntario	Atributo no nulo	Varchar(300)	Descripción de los intereses de ser voluntario
COMMENTS	Comentarios adicionales	-	Varchar(300)	Comentarios que quiera añadir
VALIDATE	Estado del usuario	Atributo no nulo	Binary(1)	Indica si el voluntario ha sido validado
DATE	Fecha de registro	Atributo no nulo	Date	Fecha de registro

Tabla 117: Tabla uservalidate

Tabla studies				
Atributo	Descripción	Tipo Atributo	Tipo Dato	Valores
ID_STUDIES	Identificador único de estudios	Clave primaria	Int(11)	0 a $2^{31} - 1$
TITLE_STUDIES	Contiene los estudios que tiene un usuario	Atributo no nulo	Varchar(50)	Título de estudios
AREA	Área de los estudios	Atributo no nulo	Varchar(45)	Área de los estudios
DATE_END	Fecha de finalización de estudios	Atributo no nulo	Date	Fecha de finalización de estudios
EMAIL_USER	Email del usuario	Atributo no nulo	Varchar(45)	Email del usuario

Tabla 118: Tabla studies

Tabla experience				
Atributo	Descripción	Tipo Atributo	Tipo Dato	Valores
ID_EXPERIENCE	Identificador único de las ciudades.	Clave primaria	Int(11)	0 a $2^{31} - 1$
SECTOR	Contiene el sector en el que tiene experiencia	Atributo no nulo	Varchar(30)	Sector en el que tiene experiencia
POST	Puesto desempeñado	Atributo no nulo	Varchar(50)	Puesto desempeñado
DESCRIPTION	Descripción de la experiencia en las áreas seleccionadas	Atributo no nulo	Varchar(300)	Descripción de la experiencia en las áreas seleccionadas
EMAIL_USER	Email del usuario	Atributo no nulo	Varchar(45)	Email del usuario

Tabla 119: Tabla experience

Tabla languages				
Atributo	Descripción	Tipo Atributo	Tipo Dato	Valores
ID_LANGUAGE	Identificador único de las ciudades.	Clave primaria	Int(11)	0 a $2^{31} - 1$
LANGUAGE_NAME	Contiene los idiomas	Atributo no nulo	Varchar(50)	Idioma
LEVEL	Nivel de los idiomas	Atributo no nulo	Varchar(30)	Nivel de idiomas
TITLE	Títulos de idiomas	Atributo no nulo	Varchar(200)	Título de idiomas
EMAIL_USER	Email del usuario	Atributo no nulo	Varchar(45)	Email del usuario

Tabla 120: Tabla languages

Tabla activities				
<i>Atributo</i>	<i>Descripción</i>	<i>Tipo Atributo</i>	<i>Tipo Dato</i>	<i>Valores</i>
ID_ACTIVITY	Identificador único de las ciudades.	Clave primaria	Int(11)	0 a $2^{31} - 1$
AREA	Contiene el nombre de la ciudad.	Atributo no nulo	Varchar(50)	Ciudades de España
RESPONSIBILITY	Identificador del trabajador responsable de la actividad	Clave ajena	Int(11)	0 a $2^{31} - 1$
NUMBER_VOLUNTARIES	Número de voluntarios necesarios	Atributo no nulo	Int(3)	
DESCRIPTION	Descripción de la actividad	Atributo no nulo	Varchar(400)	Descripción de la actividad
SEXO	Sexo	Atributo no nulo	Varchar(14)	Hombre, Mujer
EDUCATION	Educación necesaria	Atributo no nulo	Varchar(200)	Primaria, Universitarios, Indiferente...
SCHEDULE	Duración diaria de la actividad	Atributo no nulo	Varchar(10)	Duración diaria de la actividad
DAYS	Días que se realiza la actividad	Atributo no nulo	Varchar(50)	Días que se realiza la actividad
PERIOD	Periodo en el que se realizan	Atributo no nulo	Varchar(30)	Mañanas, Tardes, Indiferente
INCORPORATION	Fecha de incorporación	Atributo no nulo	Date	Fecha de incorporación
REQUIREMENTS	Requisitos necesarios	-	Varchar(200)	Requisitos
LANGUAGES	Idiomas necesarios	Atributo no nulo	Varchar(45)	Idiomas necesarios
DATE	Fecha de inicio actividad	Atributo no nulo	Date	Fecha de inicio actividad

NAMEACTIVITY	Nombre de la actividad	Atributo no nulo	Varchar(40)	Nombre de la actividad
DATE_END	Fecha fin de actividad	Atributo no nulo	Date	Fecha fin actividad
DROP	Estado de la actividad	Atributo no nulo	Binary(1)	Indica si ha sido borrada

Tabla 121: Tabla activities

Tabla activities_user				
Atributo	Descripción	Tipo Atributo	Tipo Dato	Valores
ID	Identificador de tabla activities_user	Clave primaria	Int(11)	0 a $2^{31} - 1$
ID_ACTIVITY	Identificador de actividad	Clave ajena	Int(11)	0 a $2^{31} - 1$
ID_USER	Identificador de usuario	Clave ajena	Int(11)	0 a $2^{31} - 1$
HOURL	Horas de la actividad	Atributo no nulo	Int(3)	Horas de la actividad

Tabla 122: Tabla activities_user

Tabla workers				
Atributo	Descripción	Tipo de Atributo	Tipo de Dato	Valores
ID_WORKER	Identificador del trabajador	Clave primaria	Int(11)	0 a $2^{31} - 1$
EMAIL	Email del trabajador	Atributo no nulo	Varchar(45)	Email
PASSWORD	Contraseña del trabajador	Atributo no nulo	Varchar(30)	Contraseña
NAME	Nombre del trabajador	Atributo no nulo	Varchar(30)	Nombre
SURNAME	Apellidos del trabajador	Atributo no nulo	Varchar(45)	Apellidos
PHONE	Teléfono del trabajador	Atributo no nulo	Int(9)	Teléfono
DEPARTMENT	Departamento al que pertenece	Atributo no nulo	Varchar(50)	Departamento al que pertenece
TYPE	Indica si es administrador o no	Atributo no nulo	Enum	Admin, noAdmin
DATE	Fecha de creación del trabajador	Atributo no nulo	Date	Fecha creación del trabajador

Tabla 123: Tabla workers



7 Implementación e implantación del sistema

La implementación e implantación del sistema tiene como objetivo la entrega y aceptación final del sistema, y la realización de todas las actividades necesarias para garantizar su correcta implantación.

7.1 Establecimiento del plan de implantación

El principal beneficio de la elección de una correcta estrategia de implantación radica en la anticipación para calcular los recursos y el esfuerzo necesarios para mantener en funcionamiento el software diseñado. Para este caso, determinamos que el plan de implantación del sistema ha de seguir las siguientes fases:

- Formación necesaria para la implantación, tanto de los usuarios finales del sistema como del equipo encargado de definir las pruebas de implantación y aceptación.
- Preparación de la infraestructura para la incorporación del sistema al entorno de operación.
- Instalación de los componentes y realización de los procedimientos necesarios para la incorporación del sistema al entorno de operación.
- Carga inicial de los datos.
- Realización de las pruebas de implantación y aceptación.
- Formalización de un plan de mantenimiento.

A su vez, es necesario definir una serie de roles a desempeñar durante todo el proceso de implantación del sistema, si bien varios roles pueden ser realmente desempeñados por una misma persona física:

- **Rol [1] – Responsable de implantación:** será el encargado de coordinar y supervisar la correcta implantación del sistema. Realizará un seguimiento de todo el proceso y ofrecerá soporte al resto de miembros.
- **Rol [2] – Administrador de base de datos:** encargado de gestionar la integridad y consistencia de la base de datos, así como de la carga inicial de los mismos.
- **Rol [3] – Administrador del servidor:** encargado del mantenimiento y gestión del servidor sobre el que correrá la aplicación final.

- **Rol [4] – Administrador del sitio web:** encargado de la gestión, seguridad y mantenimiento del sitio web sobre el que se construye la aplicación.

7.2 Formación necesaria para la implantación y mantenimiento del sistema

Cada miembro del equipo que desempeñe un rol distinto deberá tener una formación para poder llevar a cabo la implantación y mantenimiento del sistema. En función de los distintos perfiles y niveles de responsabilidad, se establece un plan de formación de acuerdo al rol de cada individuo que participe en la implantación del sistema. Se proponen los siguientes cursos:

- MySQL: principios de las Bases de Datos relacionales (30 horas).
- WordPress: aprenda WordPress fácilmente (15 horas).
- PHP: PHP avanzado MVC (50 horas).
- HTML5 y CSS3: curso de diseño adaptativo (15 horas).

Como es lógico los roles definidos en el apartado anterior deberán realizar los cuatro cursos establecidos para garantizar un buen desarrollo de la aplicación, garantizando la formación mínima para la implantación final.

En el caso de los usuarios finales de la aplicación, podemos distinguir entre los trabajadores de la fundación y los voluntarios. Como mínimo deberán leer los manuales de uso para facilitar la utilización de la aplicación. Principalmente deberán hacer uso de este manual los trabajadores, puesto que realizar operaciones más complejas y muchos de ellos no tienen muchos conocimientos informáticos.

7.3 Elaboración de manuales

Se ha elaborado una serie de manuales sobre el funcionamiento del sistema destinados a los diferentes roles que harán uso de la aplicación. Dichos manuales están incluidos como anexos al final del documento, y son los siguientes:

- **Manual de trabajador:** contiene la información de cómo realizar las gestiones con los distintos elementos de la aplicación.



- **Manual de voluntario:** contiene la información de cómo realizar las distintas operaciones de este tipo de usuario.

7.4 Incorporación del sistema al entorno de operación

El objetivo de este apartado es dar unas instrucciones claras para incorporar el sistema al entorno de operación en el que funcionará, y en el cual se realizarán las pruebas de implantación y de aceptación del sistema. Es esencial que estas pruebas se lleven a cabo en el entorno real de funcionamiento, para que no existan imprevistos una vez que se inicie la producción.

Como paso previo a la realización de dichas pruebas y de acuerdo al plan de implantación establecido, se verificará que los recursos necesarios están disponibles para que se pueda realizar la instalación de todos los componentes que integran el sistema, así como la creación y puesta a punto de las bases de datos en el entorno de operación. Asimismo, se establecen los procedimientos de explotación y uso de las bases de datos de acuerdo a la normativa existente en dicho entorno.

El objetivo de esta tarea es realizar la instalación de la base del sistema, para la incorporación posterior de los componentes asociados a los sistemas de información implicados en la implantación. El sistema base consta de los siguientes elementos:

- Equipo servidor: Servidor web Apache HTTP Server, y el Sistema Gestor de Base de Datos MySQL.

En el equipo servidor tenemos que realizar la instalación básica de WordPress. Para ello vamos a utilizar *wordpress-4.3.1-es_ES.zip* en la carpeta correspondiente al servidor web (*madrina.org/web1/*).

Como requisito del cliente solicitaron hacer uso del tema predefinido *Travelify*, por lo que tuve que descargarlo e instalarlo en WordPress, descomprimiéndolo en la carpeta */wordpress/wp-content/themes/*, y posteriormente activarlo desde las opciones de administración de WordPress.



Una vez realizada la instalación de WordPress, añadimos la base de datos diseñada al servidor, en la que se almacenaran todos los datos relacionados con usuarios y actividades, en caso de tener una carga inicial de datos, se ejecutará en este punto.

Posteriormente se crearán todas las páginas que permitirán al usuario interactuar con la base de datos y gestionar los distintos elementos del sistema. El manejo de estas páginas se detalla en los manuales de usuario incluidos al final del documento.

7.5 Carga de datos al entorno de operación

Para el correcto funcionamiento del sistema, es necesaria la carga de datos iniciales, durante la instalación del sistema, puesto que tenemos que definir a un trabajador de tipo administrador que se encargue de gestionar y añadir al resto de usuarios. Una vez cargado este usuario no será necesario realizar más carga de datos, puesto que serán los propios trabajadores los encargados de añadir nuevos elementos a la base de datos, incluso cualquier voluntario que quiera registrarse, deberá hacerlo el mismo.

Dicha carga de datos se realiza directamente haciendo uso de la propia aplicación, con ayuda del manual definido.

7.6 Preparación del mantenimiento del sistema

En lo referente al mantenimiento del sistema, se propone la utilización de dos entornos bien diferenciados que permitan garantizar la continuidad y el correcto funcionamiento del sistema al mismo tiempo que agiliza la tarea de corrección de errores e inclusión de nuevas características en caso de que fuese necesario.

7.6.1 Entorno de pruebas

Se realizarán las primeras correcciones cuando surja un problema o se realice una actualización software. Sobre este entorno se realizará un control exhaustivo sobre cada una de las versiones de software debido a que la finalidad del mismo será determinar los problemas asociados a una determinada versión. Es importante realizar un control de las versiones de WordPress con el fin de garantizar el correcto funcionamiento con cada nueva versión del CMS.



Este entorno dispondrá de todas las funcionalidades que compone la aplicación para poder probar los nuevos cambios integrados con el resto de módulos del sistema.

Si las pruebas son superadas, las modificaciones se pasarán al entorno de pre-producción.

7.6.2 Entorno de pre-producción

Se realizarán las pruebas de funcionalidad y además se tendrán en cuenta los parámetros de rendimiento y capacidad que se le exige al sistema real.

Este entorno estará controlado por personal independiente a los entornos previos con el objetivo de que las pruebas realizadas sobre el mismo sean completas e integra y no exista ninguna influencia procedente de las anteriores pruebas realizadas.

Si las pruebas son superadas, los cambios se pasarán al entorno de producción.

Este punto es importante en el desarrollo de este proyecto, puesto que se tendrá el sistema dos meses en preproducción para garantizar el correcto funcionamiento del mismo, garantizando la seguridad de todos los usuarios y el correcto manejo de datos.

7.6.3 Entorno de producción

Es el entorno en el que funciona la aplicación. En dicho entorno se realizará un seguimiento (durante el denominado periodo de adaptación) de las posibles modificaciones o actualizaciones realizadas hasta asegurarse que la nueva modificación sea efectiva y estable.

Si las pruebas son superadas, el entorno de producción será actualizado.

En el siguiente esquema se muestra el flujo de control de incidencias:

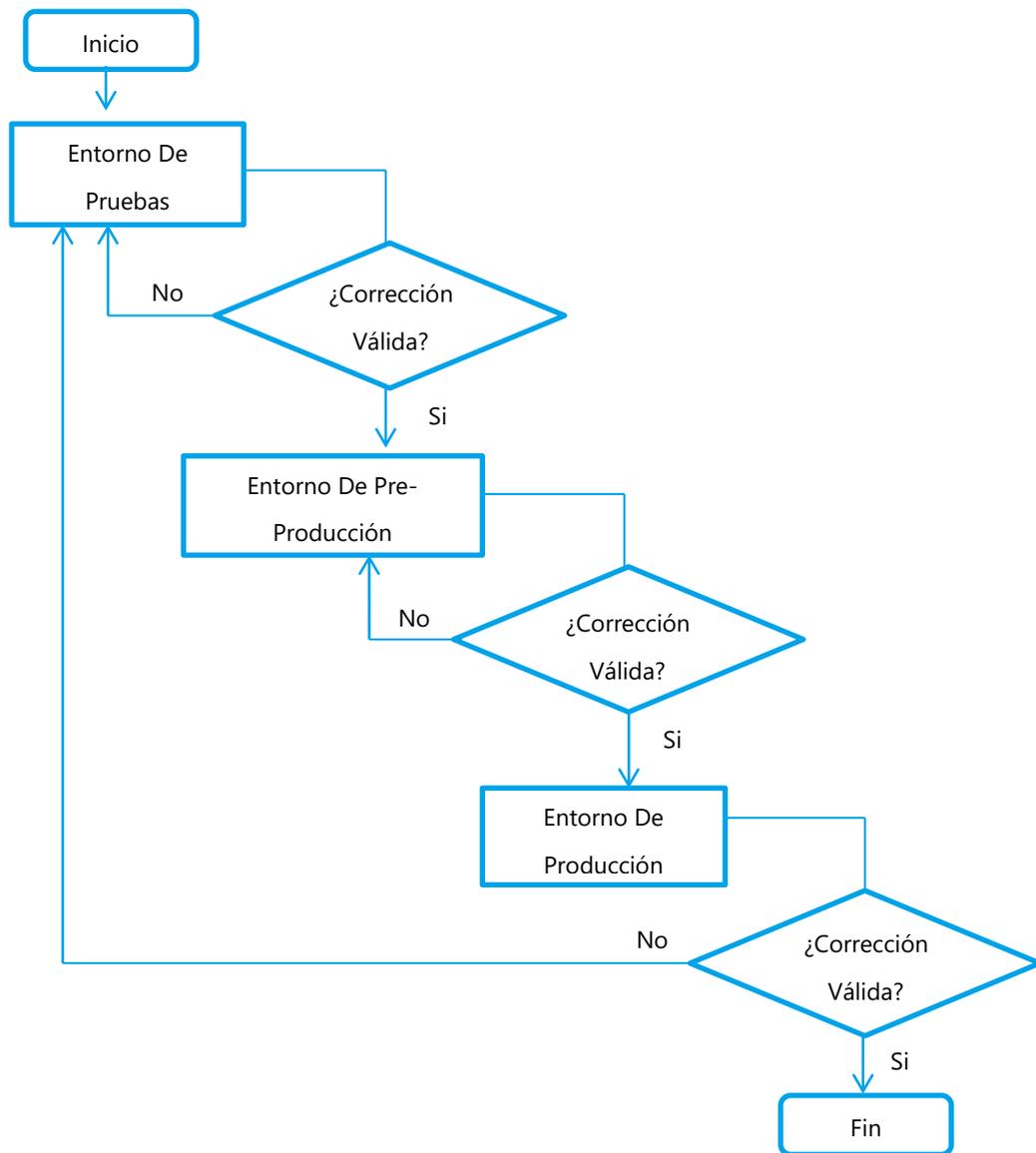


Ilustración 45: Flujo de control

La desarrolladora se compromete a dar soporte con la finalidad de resolver dudas al equipo de mantenimiento de la fundación a través de contacto telefónico o personal en caso de ser necesario. La duración de dicho soporte tendrá una extensión de seis meses desde la finalización del proyecto. Una vez superado este periodo no se garantiza el soporte por parte de la desarrolladora.

Por otro lado, cualquier error o inconsistencia que sufra la aplicación durante el periodo de adaptación a causa de un mal diseño del sistema, será solventado por la desarrolladora de manera totalmente gratuita.

7.7 Presentación y aprobación del sistema

Una vez superadas las pruebas de implantación y aprobación del sistema, se presentará al cliente con el fin de formalizar la aprobación de la aplicación web de la Fundación Madrina.

Se acordará con el cliente una reunión para exponer cada punto del sistema y realizar una presentación del mismo. En dicha reunión deben estar presentes la desarrolladora de la aplicación Sofía Torres y distintos miembros de la fundación: Conrado (gerente) y Manuel (informático).

Existirá a partir de ese momento, un periodo de prueba de 30 días naturales cuyo fin es permitir al cliente comprobar la correcta funcionalidad de la aplicación, tras el cual deberá indicar al desarrollador si se decide aprobar el sistema.

7.8 Paso a producción

El paso a producción se llevará a cabo de la forma planificada, y será el momento en el que se pasa la responsabilidad al informático de la Fundación Madrina. En este proceso diferenciamos dos pasos:

- Análisis y determinación de los componentes a incorporar en producción, de acuerdo al entorno donde se hayan llevado a cabo las pruebas. Tras este estudio se realizará la instalación de los distintos componentes, y una carga inicial de datos.

Se comprobará la correcta instalación del sistema y se determinará una fecha de activación del mismo, y la eliminación del antiguo portal web. Se establecerá como se llevará a cabo la transición de uno a otro.

- Una vez comprobada la correcta instalación del sistema y su correcto funcionamiento, se comunicará al cliente la total disponibilidad del sistema para su uso. A partir de este momento comenzará el servicio de mantenimiento del sistema bajo las condiciones anteriormente indicadas.

8 Evaluación

En este apartado se detalla los pasos a seguir para comprobar que el sistema cumple los objetivos establecidos y comprobar el correcto funcionamiento de la aplicación.

8.1 Plan de pruebas

El establecimiento del plan de pruebas es uno de los principales puntos fuertes de un proyecto software, puesto que garantiza el correcto funcionamiento de la aplicación.

En este apartado se especificaran las pruebas del sistema que se van a llevar a cabo, en ellas encontramos: pruebas unitarias, de integración, de sistema, de implantación y de aceptación del sistema y usabilidad. Para que el producto software pase a producción es indispensable que supere todas y cada una de las pruebas establecidas.

8.1.1 Pruebas Unitarias

Las pruebas unitarias se encargan de comprobar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado. Podemos diferenciar dos tipos de pruebas:

- **Pruebas de caja negra:** se centran en estudiar la entrada y salida del sistema del que se realizan las pruebas, sin importar el funcionamiento interno.
- **Pruebas de caja blanca:** se centran en los detalles procedimentales del software, por lo que su diseño está ligado al código fuente. El testeador escoge distintos valores de entrada para examinar cada uno de los posibles flujos de ejecución del programa y cerciorarse de que se devuelven los valores de salida adecuados.

Dado que las pruebas de caja blanca son importantes cuando la complejidad ciclomática de los métodos es de 2 o mayor y en nuestra aplicación no existe ningún método con esas características, dichas pruebas no se realizarán ya que la complejidad de los métodos es de 1.

Cada una de las pruebas será definida en una tabla con los siguientes campos:



- **Identificador:** sirve para distinguir una prueba de otra inequívocamente. Cada identificador seguirá la siguiente nomenclatura:
 - PU-Xnnn: las pruebas unitarias llevarán este identificador, en el que cada letra indica:
 - PU: indica que la prueba es unitaria.
 - B: determina si son pruebas de caja blanca.
 - N: determina si son pruebas de caja negra.
 - nnn: Indica el número de requisito, este número es un valor comprendido entre 1 y 999.
- **Descripción:** detalla el objetivo de la prueba unitaria en cuestión.
- **Módulo:** especifica el módulo sobre el que se realiza la prueba, basándonos en el Modelo Vista-Controlador.
- **Clase:** nombre de la clase afectada por la prueba unitaria.
- **Método:** nombre del método sobre el que se realiza la prueba, el cual proporciona la funcionalidad a probar.
- **Entrada:** datos de entrada del método para la prueba en cuestión.
- **Salida válida (sólo caja negra):** resultados esperados de la prueba.

PU-N001	
Descripción	Comprobar el correcto funcionamiento del método que recupera datos de la tabla user.
Módulo	Modelo
Clase	DataBase
Método	getUser(id_voluntary)
Entrada	Identificador del usuario a recuperar.
Salida válida	Datos del usuario solicitado.

Tabla 124: Prueba unitaria 001

PU-N002	
Descripción	Comprobar el correcto funcionamiento del método que recupera datos de la tabla activities.
Módulo	Modelo
Clase	DataBase
Método	getActivity(id_activity)
Entrada	Identificador de la actividad a recuperar.
Salida válida	Datos de la actividad solicitada.

Tabla 125: Prueba unitaria 002

PU-N003

Descripción	Comprobar el correcto funcionamiento del método que recupera datos de la tabla de activities_user.
Módulo	Modelo
Clase	DataBase
Método	getActivityUser(id)
Entrada	Identificador de la relación entre user y activities.
Salida válida	Datos del usuario y actividad que están asignados.

Tabla 126: Prueba unitaria 003

PU-N004

Descripción	Comprobar el correcto funcionamiento del método que recupera datos de la tabla studies.
Módulo	Modelo
Clase	DataBase
Método	getStudies(email)
Entrada	Email del usuario que tiene relación con esos estudios.
Salida válida	Estudios que tiene un determinado usuario.

Tabla 127: Prueba unitaria 004

PU-N005

Descripción	Comprobar el correcto funcionamiento del método que recupera datos de la tabla languages.
Módulo	Modelo
Clase	DataBase
Método	getLanguages(email)
Entrada	Email del usuario que tiene relación con esos idiomas.
Salida válida	Idiomas que tiene un determinado usuario.

Tabla 128: Prueba unitaria 005

PU-N006

Descripción	Comprobar el correcto funcionamiento del método que recupera datos de la tabla experience.
Módulo	Modelo
Clase	DataBase
Método	getExperience(email)
Entrada	Email del usuario que tiene relación con esa experiencia.
Salida válida	Experiencia que tiene un determinado usuario.

Tabla 129: Prueba unitaria 006



PU-N007	
Descripción	Comprobar el correcto funcionamiento del método que recupera datos de la tabla uservalidate.
Módulo	Modelo
Clase	DataBase
Método	getUserValidate(id_voluntary)
Entrada	Identificador del usuario a recuperar.
Salida válida	Datos del voluntario a recuperar.

Tabla 130: Prueba unitaria 007

PU-N008	
Descripción	Comprobar el correcto funcionamiento del método que inserta datos en la tabla user.
Módulo	Modelo
Clase	DataBase
Método	setUser(user)
Entrada	Atributos de usuario que se quieren almacenar.
Salida válida	Nuevo objeto insertado en la base de datos.

Tabla 131: Prueba unitaria 008

PU-N009	
Descripción	Comprobar el correcto funcionamiento del método que inserta datos en la tabla activities.
Módulo	Modelo
Clase	DataBase
Método	setActivity(activity)
Entrada	Atributos de actividad que se quieren almacenar.
Salida válida	Nuevo objeto insertado en la base de datos.

Tabla 132: Prueba unitaria 009

PU-N010	
Descripción	Comprobar el correcto funcionamiento del método que inserta datos en la tabla studies.
Módulo	Modelo
Clase	DataBase
Método	setStudies(studies)
Entrada	Atributos de estudios que se quieren almacenar.
Salida válida	Nuevo objeto insertado en la base de datos.

Tabla 133: Prueba unitaria 010

PU-N011

Descripción	Comprobar el correcto funcionamiento del método que inserta datos en la tabla languages.
Módulo	Modelo
Clase	DataBase
Método	setLanguages(language)
Entrada	Atributos de idiomas que se quieren almacenar.
Salida válida	Nuevo objeto insertado en la base de datos.

Tabla 134: Prueba unitaria 011

PU-N012

Descripción	Comprobar el correcto funcionamiento del método que inserta datos en la tabla experience.
Módulo	Modelo
Clase	DataBase
Método	setExperience(experience)
Entrada	Atributos de experiencia que se quieren almacenar.
Salida válida	Nuevo objeto insertado en la base de datos.

Tabla 135: Prueba unitaria 012

PU-N013

Descripción	Comprobar el correcto funcionamiento del método que inserta datos en la tabla uservalidate.
Módulo	Modelo
Clase	DataBase
Método	setUserValidate(uservalidate)
Entrada	Atributos de usuario que se quieren almacenar.
Salida válida	Nuevo objeto insertado en la base de datos.

Tabla 136: Prueba unitaria 013

PU-N014

Descripción	Comprobar el correcto funcionamiento del método que inserta datos en la tabla activities_user.
Módulo	Modelo
Clase	DataBase
Método	setActivityUser(id_activity, id_voluntary)
Entrada	Identificador del objeto actividad y voluntario que se quieren almacenar.
Salida válida	Nueva asignación insertada en la base de datos.

Tabla 137: Prueba unitaria 014



PU-N015	
Descripción	Comprobar el correcto funcionamiento del método que recupera usuarios de la base de datos.
Módulo	Modelo
Clase	DataBase
Método	getSearchUser(atributo, valor)
Entrada	Nombre del atributo y valor por el que se filtrará la búsqueda.
Salida válida	Lista de usuarios que coinciden con los parámetros de búsqueda.

Tabla 138: Prueba unitaria 015

PU-N016	
Descripción	Comprobar el correcto funcionamiento del método que recupera las actividades de la base de datos.
Módulo	Modelo
Clase	DataBase
Método	getSearchActivities(atributo, valor)
Entrada	Nombre del atributo y valor por el que se filtrará la búsqueda.
Salida válida	Lista de actividades que coinciden con los parámetros de búsqueda.

Tabla 139: Prueba unitaria 016

PU-N017	
Descripción	Comprobar el correcto funcionamiento del método que recupera los usuarios que están sin validar.
Módulo	Modelo
Clase	DataBase
Método	getUserNoValidate()
Entrada	-
Salida válida	Todos los voluntarios que estén sin validar.

Tabla 140: Prueba unitaria 017

PU-N018	
Descripción	Comprobar el correcto funcionamiento del método que recupera todas las actividades que tienen voluntarios asignados.
Módulo	Modelo
Clase	DataBase
Método	getActivitiesUser()
Entrada	-
Salida válida	Todas las actividades que tienes usuarios asignados.

Tabla 141: Prueba unitaria 018

PU-N019	
Descripción	Comprobar el correcto funcionamiento del método que elimina un usuario de la tabla user.
Módulo	Modelo
Clase	DataBase
Método	removeUser(id_voluntary)
Entrada	-
Salida válida	Elemento borrado de la base de datos.

Tabla 142: Prueba unitaria 019

PU-N020	
Descripción	Comprobar el correcto funcionamiento del método que elimina una actividad de la tabla activities.
Módulo	Modelo
Clase	DataBase
Método	removeActivity(id_activity)
Entrada	-
Salida válida	Elemento borrado de la base de datos.

Tabla 143: Prueba unitaria 020

PU-N021	
Descripción	Comprobar el correcto funcionamiento del método que recupera datos de la tabla de workers.
Módulo	Modelo
Clase	DataBase
Método	getWorkers(id_worker)
Entrada	Identificador del trabajador a recuperar.
Salida válida	Datos del trabajador a recuperar.

Tabla 144: Prueba unitaria 021

PU-N022	
Descripción	Comprobar el correcto funcionamiento del método que inserta datos de la tabla de workers.
Módulo	Modelo
Clase	DataBase
Método	setWorkers(worker)
Entrada	Atributos de experiencia que se quieren almacenar.
Salida válida	Nuevo objeto insertado en la base de datos.

Tabla 145: Prueba unitaria 022



PU-N023	
Descripción	Comprobar el correcto funcionamiento del método que elimina un trabajador de la tabla workers.
Módulo	Modelo
Clase	DataBase
Método	removeWorker(id_worker)
Entrada	-
Salida válida	Elemento borrado de la base de datos.

Tabla 146: Prueba unitaria 023

PU-N024	
Descripción	Comprobar el correcto funcionamiento que compone la página de mi perfil.
Módulo	Vista
Clase	Voluntaries
Método	voluntary()
Entrada	-
Salida válida	Página de mi perfil.

Tabla 147: Prueba unitaria 024

PU-N025	
Descripción	Comprobar el correcto funcionamiento que compone la página de modificar mi perfil.
Módulo	Vista
Clase	Voluntaries
Método	modifyVoluntary()
Entrada	-
Salida válida	Página de modificar mi perfil.

Tabla 148: Prueba unitaria 025

PU-N026	
Descripción	Comprobar el correcto funcionamiento que compone la página de actividades que tiene asignadas el usuario.
Módulo	Vista
Clase	Voluntaries
Método	activitiesUser()
Entrada	-
Salida válida	Página de actividades asignadas a mi perfil.

Tabla 149: Prueba unitaria 026

PU-N027

Descripción	Comprobar el correcto funcionamiento que compone la página de perfil trabajador.
Módulo	Vista
Clase	Worker
Método	worker()
Entrada	-
Salida válida	Página que muestra el perfil de un trabajador.

Tabla 150: Prueba unitaria 027

PU-N028

Descripción	Comprobar el correcto funcionamiento que compone la página modificar de perfil trabajador.
Módulo	Vista
Clase	Worker
Método	modifyWorker()
Entrada	-
Salida válida	Página modificar el perfil de un trabajador.

Tabla 151: Prueba unitaria 028

PU-N029

Descripción	Comprobar el correcto funcionamiento que compone la página de ver actividad.
Módulo	Vista
Clase	Activity
Método	activity()
Entrada	-
Salida válida	Página que muestra una actividad.

Tabla 152: Prueba unitaria 029

PU-N030

Descripción	Comprobar el correcto funcionamiento que compone la página modificar actividad.
Módulo	Vista
Clase	Activity
Método	modifyActivity()
Entrada	-
Salida válida	Página que muestra una actividad.

Tabla 153: Prueba unitaria 030

PU-N031	
Descripción	Comprobar el correcto funcionamiento que compone la página usuarios que tiene asignados una actividad.
Módulo	Vista
Clase	Activity
Método	userActivites()
Entrada	-
Salida válida	Página que muestra los voluntarios que tiene asignados una actividad.

Tabla 154: Prueba unitaria 031

PU-N032	
Descripción	Comprobar el correcto funcionamiento que compone la página que muestra todas las actividades con los voluntarios que tiene asignados.
Módulo	Vista
Clase	ActivityUser
Método	activitiesUserAll()
Entrada	-
Salida válida	Página que muestra todas las actividades con los voluntarios asignados.

Tabla 155: Prueba unitaria 032

PU-N033	
Descripción	Comprobar el correcto funcionamiento que compone la página que muestra todos los voluntarios.
Módulo	Vista
Clase	User
Método	allUser()
Entrada	-
Salida válida	Página que muestra todos los voluntarios.

Tabla 156: Prueba unitaria 033

PU-N034	
Descripción	Comprobar el correcto funcionamiento que compone la página que muestra todas las actividades.
Módulo	Vista
Clase	Activity
Método	allActivities()
Entrada	-
Salida válida	Página que muestra todas las actividades.

Tabla 157: Prueba unitaria 034

8.1.2 Pruebas de integración

Las pruebas de integración son aquellas que se realizan en el ámbito de desarrollo software una vez que se han aprobado todas las pruebas unitarias.

Las pruebas de integración del sistema, serán definidas en tablas, utilizando los siguientes campos:

- **Identificador:** sirve para distinguir una prueba de otra inequívocamente. Cada identificador seguirá la siguiente nomenclatura:
 - PI-nnn: Las pruebas unitarias llevarán este identificador, en el que cada letra indica:
 - PU: indica que la prueba es de integración.
 - nnn: Indica el número de requisito, este número es un valor comprendido entre 1 y 999.
- **Descripción:** detalla el objetivo de la prueba.
- **Clase:** nombre de la clase afectada por la prueba unitaria.
- **Entrada:** datos de entrada del método para la prueba en cuestión.
- **Secuencia de pasos:** se indica los pasos llevados a cabo para la realización de la prueba.
- **Salida:** resultado de la prueba.
- **Precondiciones:** condiciones a cumplir para ejecutar la prueba.

PI-001	
Descripción	Ir a la página de mi perfil.
Clase	Voluntaries
Entrada	Datos de inicio de sesión (correo y contraseña).
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder a la página de login. 3. Introducir los credenciales de inicio de sesión. 4. Acceder a la página de mi perfil.
Salida	Acceso a datos de perfil.
Precondiciones	La página web debe estar accesible desde el navegador. El usuario debe estar almacenado en la base de datos.

Tabla 158: Prueba de integración 001



PI-002	
Descripción	Modificar perfil de un voluntario.
Clase	Voluntaries
Entrada	Datos del voluntario.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder a la página de login. 3. Introducir los credenciales de inicio de sesión. 4. Acceder a la página de mi perfil. 5. Editar perfil.
Salida	Se modifica el perfil del voluntario.
Precondiciones	La página web debe estar accesible desde el navegador. El usuario debe estar almacenado en la base de datos.

Tabla 159: Prueba de integración 002

PI-003	
Descripción	Eliminar voluntario.
Clase	Voluntaries
Entrada	Datos del voluntario.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder a la página de login. 3. Introducir los credenciales de inicio de sesión. 4. Acceder a la página de mi perfil. 5. Borrar voluntario.
Salida	Se elimina el voluntario de la base de datos.
Precondiciones	La página web debe estar accesible desde el navegador. El usuario debe estar almacenado en la base de datos.

Tabla 160: Prueba de integración 003

PI-004	
Descripción	Ir a la página de ver actividades asignadas a un voluntario.
Clase	Voluntaries
Entrada	Sin datos.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder a la página de login. 3. Introducir los credenciales de inicio de sesión. 4. Accede a ver actividades asignadas.
Salida	Aparecerá una tabla con la lista de actividades asignadas.
Precondiciones	La página web debe estar accesible desde el navegador. El usuario debe estar almacenado en la base de datos.

Tabla 161: Prueba de integración 004

PI-005	
Descripción	Añadir un voluntario.
Clase	Voluntaries
Entrada	Datos del voluntario.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder a la página de hazte voluntario.
Salida	Se añade el nuevo voluntario a la base de datos user.
Precondiciones	La página web debe estar accesible desde el navegador.

Tabla 162: Prueba de integración 005

PI-006	
Descripción	Ir a la página de perfil de trabajador.
Clase	Workers
Entrada	Datos de inicio de sesión (correo y contraseña).
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder a la página de login. 3. Introducir los credenciales de inicio de sesión. 4. Acceder a la página perfil trabajador.
Salida	Acceso a datos de perfil del trabajador.
Precondiciones	La página web debe estar accesible desde el navegador. El usuario debe estar almacenado en la base de datos.

Tabla 163: Prueba de integración 006

PI-007	
Descripción	Modificar trabajador.
Clase	Workers
Entrada	Datos del trabajador.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder a la página de login. 3. Introducir los credenciales de inicio de sesión. 4. Acceder a la página de perfil trabajador. 5. Modificar trabajador.
Salida	Se modifican los datos del trabajador de la base de datos.
Precondiciones	La página web debe estar accesible desde el navegador. El trabajador debe estar almacenado en la base de datos.

Tabla 164: Prueba de integración 007

PI-008	
Descripción	Eliminar trabajador.
Clase	Workers
Entrada	Datos del trabajador.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder a la página de login. 3. Introducir los credenciales de inicio de sesión. 4. Acceder a la página de perfil trabajador. 5. Borrar trabajador.
Salida	Se elimina el trabajador de la base de datos.
Precondiciones	La página web debe estar accesible desde el navegador. El trabajador debe estar almacenado en la base de datos.

Tabla 165: Prueba de integración 008

PI-009	
Descripción	Añadir trabajador.
Clase	Workers
Entrada	Datos del trabajador.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder al sistema como trabajador. 3. Ir a la página añadir trabajador.
Salida	Se añade un trabajador de la base de datos.
Precondiciones	La página web debe estar accesible desde el navegador.

Tabla 166: Prueba de integración 009

PI-010	
Descripción	Añadir actividad.
Clase	Activities
Entrada	Datos de la actividad.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder al sistema como trabajador. 3. Ir a la página añadir actividad.
Salida	Se añade la actividad a la base de datos.
Precondiciones	La página web debe estar accesible desde el navegador.

Tabla 167: Prueba de integración 010

PI-011	
Descripción	Ir a la página de una actividad.
Clase	Activities
Entrada	Datos de la actividad.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder al sistema como trabajador. 3. Ir a la página de ver actividad.
Salida	Acceso datos de la actividad.
Precondiciones	La página web debe estar accesible desde el navegador. Debe existir la actividad en el sistema.

Tabla 168: Prueba de integración 011

PI-012	
Descripción	Modificar actividad.
Clase	Activities
Entrada	Datos de la actividad.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder al sistema como trabajador. 3. Ir a la página de modificar actividad.
Salida	Se modifican los datos de la actividad en la base de datos.
Precondiciones	La página web debe estar accesible desde el navegador. Debe existir la actividad en el sistema.

Tabla 169: Prueba de integración 012

PI-013	
Descripción	Eliminar actividad.
Clase	Activities
Entrada	Sin datos.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder al sistema como trabajador. 3. Ir a la página de eliminar actividad.
Salida	Se elimina la actividad de la base de datos.
Precondiciones	La página web debe estar accesible desde el navegador. Debe existir la actividad en el sistema.

Tabla 170: Prueba de integración 013

PI-014	
Descripción	Ver usuarios que tiene asignados una actividad.
Clase	Activities
Entrada	Sin datos.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder al sistema como trabajador. 3. Accede a ver usuarios que tiene asignados.
Salida	Aparecerá una tabla con la lista de voluntarios asignadas.
Precondiciones	La página web debe estar accesible desde el navegador. La actividad debe estar almacenada en la base de datos.

Tabla 171: Prueba de integración 014

PI-015	
Descripción	Ir a la página de validar voluntario.
Clase	Voluntaries
Entrada	Datos del voluntario.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Entrar a la web de la Fundación Madrina. 2. Acceder al sistema como trabajador. 3. Accede a validar voluntario.
Salida	Aparecerán los datos del voluntario a validar.
Precondiciones	La página web debe estar accesible desde el navegador. El voluntario debe estar almacenada en la base de datos.

Tabla 172: Prueba de integración 015

8.1.3 Pruebas del sistema

Las pruebas del sistema tienen como objetivo comprobar que el sistema en su conjunto funciona de forma correcta. En nuestro caso se ha decidido realizar un test de satisfacción de usuarios realizado a los clientes de la aplicación (Fundación Madrina) y otros usuarios externos (voluntarios). Estos test están definidos en los siguientes apartados.

8.1.4 Pruebas de implantación del sistema

El principal objetivo de las pruebas de implantación es verificar el correcto funcionamiento del sistema. Las pruebas se componen de las siguientes especificaciones:

- **Recuperación:** en caso de fallo, se comprueba si se recupera el sistema de forma adecuada.
- **Seguridad:** se comprueba si el sistema ofrece protección de acuerdo a sus objetivos.

- **Rendimiento:** se comprueba que el rendimiento del sistema es el adecuado, en cuanto al consumo de recursos, tiempo de ejecución y respuesta.
- **Comunicación:** se comprueba que la comunicación es correcta, es fluida.

Las pruebas del sistema, serán definidas en tablas, utilizando los siguientes campos:

- **Identificador:** sirve para distinguir una prueba de otra inequívocamente. Cada identificador seguirá la siguiente nomenclatura:
 - PIS-*nnn*: Las pruebas unitarias llevarán este identificador, en el que cada letra indica:
 - PIS: indica que la prueba es de implantación del sistema.
 - nnn: Indica el número de requisito, este número es un valor comprendido entre 1 y 999.
- **Descripción:** detalla el objetivo de la prueba.
- **Entrada:** datos de entrada del método para la prueba en cuestión.
- **Secuencia de pasos:** se indica los pasos llevados a cabo para la realización de la prueba.
- **Salida:** resultado de la prueba.
- **Precondiciones:** condiciones a cumplir para ejecutar la prueba.

PIS-001	
Descripción	Comprobar que la aplicación está instalada correctamente en el servidor.
Entrada	Se comprueba cuando se inicia el sistema.
Secuencia de pasos	1. Se accede a la web de la Fundación Madrina.
Salida	Se inicia correctamente, y responde peticiones.
Precondiciones	Servidor iniciado.

Tabla 173: Prueba de implantación del sistema 001

PIS-002	
Descripción	Comprobar la conexión con el servidor que contiene la Base de Datos.
Entrada	Se comprueba cuando se inicia el sistema y accede a la base de datos.
Secuencia de pasos	1. Se accede a la web de la Fundación Madrina. 2. Se comprueba que se puede iniciar sesión.
Salida	Permite iniciar sesión comprobando que existe el usuario.
Precondiciones	Servidor iniciado y servidor con la base de datos iniciado.

Tabla 174: Prueba de implantación del sistema 002

PIS-003	
Descripción	Comprobar la longitud de campos de formularios para evitar ataques.
Entrada	Introducir una cadena de caracteres superior a la permitida.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Se accede a la web de la Fundación Madrina. 2. Se introduce una longitud no permitida.
Salida	Muestra mensaje de longitud inválida.
Precondiciones	-

Tabla 175: Prueba de implantación del sistema 003

PIS-004	
Descripción	Comprobar el tipo de caracteres introducidos en los formularios.
Entrada	Introducir una cadena de caracteres con tipo no permitido.
Secuencia de pasos	<ol style="list-style-type: none"> 1. Se accede a la web de la Fundación Madrina. 2. Se introduce un tipo de carácter no permitido.
Salida	Muestra mensaje tipo de carácter no permitido.
Precondiciones	-

Tabla 176: Prueba de implantación del sistema 004

8.1.5 Pruebas de aceptación del sistema

El propósito de estas pruebas es comprobar si el producto software se puede calificar como aceptable y si está listo para su paso a producción. En nuestro caso se ha decidido realizar un test de satisfacción de usuarios realizado a los clientes de la aplicación (Fundación Madrina) y otros usuarios externos (voluntarios). Estos test están definidos en los siguientes apartados.

8.1.6 Pruebas de usabilidad del sistema

El propósito de las pruebas de usabilidad del sistema es comprobar si el producto software es usable y está listo para su paso a producción. Para analizar la usabilidad del sistema hemos utilizado tres herramientas, que han ayudado a optimizar el sistema en fase de pre-producción, ya puesto que se encontraron una serie de elementos que mejoraban la usabilidad del sistema.

8.1.6.1 Google Analytics

Google Analytics es una herramienta analítica web que ofrece información agrupada del tráfico que llega a los sitios web según la audiencia, la adquisición, el comportamiento y las conversiones que se llevan a cabo en el sitio web. Se pueden obtener informes como el seguimiento de usuarios, tasas de rebote, duración de las sesiones, lugares de acceso, etc.

Gracias a esta aplicación podemos analizar desde que dispositivos tenemos acceso, en este caso observamos que más del 80% de los accesos se hacen desde un pc, ya sea Windows o Mac. Esta cifra tan alta se debe a que la web antigua no era accesible desde un dispositivo móvil, entonces los usuarios no están acostumbrados y tardará un tiempo en darse a conocer la nueva versión, para ampliar el número de usuarios desde el dispositivo móvil.

Las estadísticas de uso, no son reales al 100%, ya que se han realizado en fase de pre-producción, aun así podemos hacer una estimación de la usabilidad que va a tener en el futuro.

8.1.6.2 Web Master Tools

Esta herramienta permite comprobar el estado de la indexación de los sitios en internet por el buscador y optimizar su visibilidad.

Para poder hacer uso de esta herramienta, hemos tenido que registrar la URL de nuestra aplicación web, a continuación se genera un fichero que tenemos que subir a nuestro servidor, para garantizar la autenticidad del usuario. Una vez subido el fichero tenemos que dejar pasar un mínimo de 24 horas para que analice toda la aplicación.

Si analizamos los resultados obtenidos, vemos que no nos muestra ningún error, esto indica que nuestra aplicación es usable. Vemos que no tiene errores DNS, conecta correctamente con el servidor, que tenemos el fichero robots.txt configurado correctamente para acceder a los distintos elementos de la aplicación.

Los datos de las páginas están correctamente estructurados, y el único error que encontramos es que no tenemos indicado el autor de las páginas para mejorar la usabilidad. Una vez analizado este error, se han incluido el autor en todas las páginas y días después hemos visto como se ha solucionado el problema.

Gracias a esta aplicación hemos analizado la estructura de toda la aplicación y posicionamiento SEO.



8.1.6.3 Test Taw

Esta aplicación analiza la aplicación web, indicando el nivel de accesibilidad, tecnologías usadas, problemas que encuentra, advertencias, etc.

Para analizar el nivel de accesibilidad encontramos una serie de puntos de verificación que ayudan a detectar posibles errores, cada punto de verificación está asignado a uno de los tres niveles de prioridad:

- **Prioridad 1:** son aquellos puntos que un desarrollador web tiene que cumplir ya que, de otra manera, ciertos grupos de usuarios no podrían acceder a la información del sitio Web.
- **Prioridad 2:** son aquellos puntos que un desarrollador web debería cumplir ya que, si no fuese así, sería muy difícil acceder a la información para ciertos grupos de usuarios.
- **Prioridad 3:** son aquellos puntos que un desarrollador web debería cumplir ya que, de otra forma, algunos usuarios experimentarían ciertas dificultades para acceder a la información.

En función a estos puntos de verificación se establecen los niveles de conformidad:

- **Nivel de Conformidad A,** todos los puntos de verificación de prioridad 1 se satisfacen.
- **Nivel de Conformidad AA,** todos los puntos de verificación de prioridad 1 y 2 se satisfacen.
- **Nivel de Conformidad AAA,** todos los puntos de verificación de prioridad 1,2 y 3 se satisfacen.

Observamos que el resultado del análisis del test es AA, por lo que tenemos una aplicación con una accesibilidad óptima, de esta manera observamos que la usabilidad que va a tener la aplicación es adecuada al usuario.

Si seguimos analizando el resultado del test había unos problemas que se han solucionado (salvo en los casos en que esta solución empeorara el funcionamiento o la interacción con el usuario).

- **Imágenes sin etiqueta "alt"**: para solucionar el error incluimos las etiquetas "alt" a cada una de las imágenes, para obtener mejores resultados.
- **Etiquetas de encabezado juntas**: poner dos etiquetas de encabezado juntas, indican que no tiene una buena estructura el sitio, por lo que cambie las etiquetas por otras para que la estructura fuera correcta.
- **Página "bien formada"**: encontramos una serie de errores que indican que se puede mejorar la robustez de la aplicación, ya que encuentra código JavaScript en las páginas HTML. Estos errores no se pueden solucionar ya que es defecto del tema elegido para el desarrollo de la aplicación.
- **Enlaces sin contenido**: muestra unos enlaces predefinidos en la plantilla de la aplicación que enlazan con las redes sociales, la manera de enlazar no es la mejor, ya que reduce la operabilidad de la aplicación. Al ser elementos por defecto de la plantilla no se han podido solucionar.
- **Advertencia de enlaces seguidos**: en el código HTML de la aplicación encontramos muchos enlaces seguidos, algunos de ellos hemos podido optimizarlos, en cambio otros no, ya que son elementos por defecto del tema.

8.1.6.4 *Google developers*

Con esta aplicación analizamos la web a nivel móvil y ordenador:

- **Ordenador**: nos muestra una serie de sugerencias que ayudan a optimizar la aplicación web: optimizar imágenes, habilitar compresión y especificar caché del navegador. No podemos realizar ninguna de estas optimizaciones ya que no tenemos acceso a la configuración del servidor apache.
- **Móvil**: tenemos dos test, el de velocidad y experiencia de usuario. En el test de experiencia de usuario es satisfactorio al 98%, sin embargo el de velocidad ocurre como con el test del ordenador que indica que mejoremos la compresión de imágenes, habilitar la compresión, no

podemos realizarlo por no disponer de permisos para la configuración del servidor Apache.

Una vez realizado el análisis de estas cuatro aplicaciones podemos llegar a la conclusión de que es accesible el sitio web diseñado, y cumple con los estándares establecidos.

8.2 Test de satisfacción de usuarios

Una vez garantizado el correcto funcionamiento de la aplicación a través del plan de pruebas de cada uno de los elementos del sistema, se realiza un test de satisfacción de usuarios con el objetivo de proporcionar a la desarrolladora un *feedback* del funcionamiento y facilidad del sistema desarrollado. De esta manera podemos garantizar la usabilidad del sistema, posibles mejoras de la interfaz y nuevas funcionalidades que se pueden añadir al sistema en un futuro.

8.2.1 Satisfacción de voluntarios

El test consiste en una serie de preguntas que se van a realizar a 10 usuarios con diferentes conocimientos informáticos, estudios, sexo y edad.

Para realizar el test los diferentes usuarios deben realizar las siguientes operaciones: Navegar por la aplicación web – Registrarse – Logearse – Modificar datos de perfil – Cerrar sesión.

El test que se va a realizar consta de las siguientes preguntas:

1. ¿Qué le parece la nueva Web de la Fundación Madrina?
2. ¿Mejoraría la interfaz de la nueva aplicación?
3. Sobre el registro de voluntarios. ¿Ha tenido alguna dificultad para registrarse como voluntario? Indica cuales.
4. Sobre el acceso como voluntario a la aplicación. ¿Ha tenido problemas para acceder al sistema como voluntario?
5. Sobre la modificación de datos de voluntario. ¿Ha podido modificar los datos personales sin problema? Indique cualquier dificultad encontrada.

6. Sobre la navegación de la aplicación una vez *logueado*. ¿Le pareció sencilla e intuitiva la aplicación una vez que accede como voluntario? ¿Qué mejoraría?
7. Sobre la navegación por las diferentes páginas de la aplicación sin estar *logueado*. ¿Le pareció sencilla la navegación a través de las diferentes páginas de la aplicación? ¿Cree que está repartida la información de la mejor manera? ¿Qué aspectos mejoraría?
8. Indique cualquier otro aspecto a mejorar o problema que haya encontrado.

Los resultados obtenidos son los siguientes:

Test 01			
Estudios, edad, sexo		Mujer – 49 años – Higienista bucodental.	
1	Una web sencilla y funcional.	2	Metería más imágenes en las distintas páginas de la web.
3	No.	4	No.
5	Sí, no he tenido problema.	6	Sí.
7	Sí, la información está ordenada correctamente.	8	Cargan un poco despacio las distintas páginas.

Tabla 177: Test usuario 01

Test 02			
Estudios, edad, sexo		Mujer – 18 años – Estudiante Ingeniería Química.	
1	Útil y sencilla de manejar.	2	Incluiría otros colores en la web.
3	No.	4	No.
5	Sí.	6	Sí, incluiría más funciones al usuario voluntario en un futuro.
7	Sí, es una aplicación web bastante sencilla de manejar.	8	Creo que facilita las tareas a la fundación.

Tabla 178: Test usuario 02

Test 03			
Estudios, edad, sexo		Hombre – 55 años – Auxiliar administrativo	
1	Organiza mejor la información y los contenidos que la web antigua.	2	No, me gusta.
3	No.	4	No.
5	Sí, sin problema.	6	Sí, es muy sencillo su manejo.
7	Sí, es sencilla.	8	Personalmente me ha gustado mucho.

Tabla 179: Test usuario 03

Test 04			
Estudios, edad, sexo		Hombre – 21 años – ESO.	
1	Bien, es funcional.	2	Quitaría de la cabecera dos de los logos, dejando solo el principal.
3	No.	4	No.
5	Sí.	6	Sí, tiene un aspecto bastante sencillo de manejar.
7	Sí.	8	Ninguno.

Tabla 180: Test usuario 04

Test 05			
Estudios, edad, sexo		Hombre – 43 años – Abogado.	
1	Me gusta que sea adaptativa, para poder utilizarla desde mi dispositivo móvil.	2	Cambiaría el color verde.
3	No.	4	No.
5	Sí, he modificado la dirección que había introducido mal el número.	6	Sí, es sencilla de manejar.
7	Sí, la distribución me parece óptima.	8	Añadiría nuevas funcionalidades.

Tabla 181: Test usuario 05

Test 06			
Estudios, edad, sexo		Mujer – 68 años – Profesora educación infantil.	
1	Sencilla de utilizar.	2	No.
3	No.	4	No.
5	Sí.	6	Me ha gustado mucho, no cambiaría nada.
7	Sí, su manejo es bastante sencillo.	8	Ninguno.

Tabla 182: Test usuario 06

Test 07			
Estudios, edad, sexo		Hombre – 28 años – Profesor Educación física ESO.	
1	Tiene un diseño claro para su utilización.	2	La nueva interfaz, me gusta mucho más que la antigua, tiene un diseño moderno.
3	No.	4	No.
5	Sí.	6	Sí, es muy fácil de manejar.
7	Sí.	8	Ninguno.

Tabla 183: Test usuario 07

Test 08			
Estudios, edad, sexo		Mujer – 22 años – Ingeniera Informática.	
1	Una web sencilla que se adapta a todos los dispositivos.	2	No.
3	No.	4	No.
5	Sí.	6	Muy útil, además es bastante sencilla de manejar.
7	Sí.	8	No.

Tabla 184: Test usuario 08

Test 09			
Estudios, edad, sexo		Hombre – 19 años – Estudiante Psicología.	
1	Es una aplicación sencilla, que añade nuevas funcionalidades respecto la antigua.	2	No.
3	No.	4	Ninguno.
5	Sí, he podido modificarlos sin ningún tipo de problema.	6	Sí, es sencillo su manejo.
7	Sí, la información está clara y organizada.	8	No he tenido ningún problema.

Tabla 185: Test usuario 09

Test 10			
Estudios, edad, sexo		Mujer – 38 años – Enfermera.	
1	Me parece una web sencilla y bastante completa.	2	Incluiría más imágenes.
3	No.	4	Ningún problema para acceder al sistema.
5	He realizado dos modificaciones en los datos y no he tenido ningún problema.	6	Sí, es sencilla de manejar.
7	Sí, la organización es clara e intuitiva.	8	No tengo ninguna mejora que aportar.

Tabla 186: Test usuario 10

En definitiva, tal y como podemos apreciar en las respuestas proporcionadas por voluntarios parece que la aplicación web funciona sin problema, mejora la web antigua y los usuarios quedan bastante satisfechos.

8.2.2 Satisfacción de miembros de la Fundación

El test consiste en una serie de preguntas que se van a realizar a 4 miembros de la fundación madrina.

Para realizar el test los diferentes usuarios deben realizar las siguientes operaciones: Navegar por la aplicación web – *Logearse* – Modificar datos de perfil



- Ver los voluntarios registrados, eliminados y sin validar – Ver los detalles de un voluntario – Modificar un voluntario – Ver actividades, actividades acabadas, mis actividades creadas – Añadir una actividad nueva – Ver detalles de una actividad
- Modificar una actividad – Asignar una actividad a un usuario – Cerrar sesión.

El test que se va a realizar consta de las siguientes preguntas:

1. ¿Qué le parece la nueva Web de la Fundación Madrina?
2. ¿Mejoraría la interfaz de la nueva aplicación?
3. Sobre el acceso como trabajador a la aplicación. ¿Ha tenido problemas para acceder al sistema como trabajador?
4. Sobre la modificación de datos de trabajador. ¿Ha podido modificar los datos personales sin problema? Si ha encontrado alguna dificultad indíquela.
5. Sobre la navegación de la aplicación una vez *logueado*. ¿Le pareció sencilla e intuitiva la aplicación una vez que accede como trabajador? ¿Qué mejoraría?
6. Sobre la navegación por las diferentes páginas de la aplicación sin estar *logueado*. ¿Le pareció sencilla la navegación a través de las diferentes páginas de la aplicación? ¿Cree que está repartida la información de la mejor manera? ¿Qué aspectos mejoraría?
7. Sobre el acceso a voluntarios. ¿Ha tenido problemas para ver el listado de voluntarios, acceder a uno en particular o modificar algún dato? Explique cuáles.
8. Sobre el acceso a las actividades. ¿Ha tenido algún problema para ver el listado de actividades, acceder a una en particular o modificar algún dato? Explique cuáles.
9. Sobre la asignación de voluntarios a una actividad. ¿Ha encontrado alguna dificultad para asignar algún voluntario a una actividad? Indique cuáles.
10. Sobre crear una nueva actividad. ¿Ha encontrado alguna dificultad para añadir una nueva actividad a la base de datos? Indíquela.
11. Indique cualquier otro aspecto a mejorar o problema que haya encontrado.

Test 01			
Puesto	Informático.	1	Una web con un diseño actual, adaptativa y agradable de utilizar.
2	No, cumple con todos los requisitos.	3	No, he accedido sin problemas.
4	Sí, ha sido fácil de realizar esta tarea.	5	Sí, la información está organizada de forma clara y concisa en el menú.
6	Sí, es fácil navegar por las páginas. La organización me parece correcta.	7	No, gracias al manual de usuario he podido realizar esta tarea sin problema.
8	No, como bien he mencionado en la respuesta anterior, gracias al manual he podido realizar esta tarea.	9	Es sencillo asignar las tareas.
10	No he tenido problema.	11	En un futuro añadir nuevas funcionalidades como la gestión de afectados.

Tabla 187: Test trabajador 01

Test 02			
Puesto	Director.	1	Bonita, tiene los colores que escogí, así como su distribución es correcta.
2	No, es tal y como he pedido.	3	No, con ayuda del manual de usuario no he tenido problema para acceder.
4	Sí, los he modificado sin problema.	5	Sí, el menú es muy claro.
6	Es sencilla, cumple con los objetivos que le dijimos.	7	No, es sencillo de realizar.
8	No.	9	No, he podido asignar varias tareas a un voluntario sin problemas.
10	No, he creado varias actividades sin problema.	11	Me gusta mucho, en un futuro se pueden implementar mejoras para completar su funcionamiento.

Tabla 188: Test trabajador 02

Test 03			
Puesto	Call Center.	1	Bonita y sencilla.
2	No, me gusta como es.	3	He accedido sin ningún problema como trabajador de la fundación.
4	Sí.	5	Sí, no es compleja de usar.
6	Es sencilla, la organización elegida me gusta, ya que es intuitiva.	7	No, he podido ver los voluntarios y modificarlos sin ningún tipo de problemas.
8	He podido ver las actividades correctamente.	9	Esta fue la tarea que más me costó realizar, ya que no sabía que había limitación en el número de asignaciones de tareas a un voluntario.
10	No.	11	En general me gusta mucho.

Tabla 189: Test trabajador 03

Test 04			
Puesto	Sanitaria.	1	Una web bonita y bien estructurada.
2	No, me gusta.	3	No, en cuanto al login no he tenido ningún problema.
4	He modificado el número de teléfono y sin problema.	5	Si, como bien he dicho antes está bien estructurada toda la información.
6	Si.	7	No, he accedido a los datos sin problema.
8	No.	9	Realicé dos asignaciones y todo funcionó correctamente.
10	No, he añadido tres actividades sin encontrar ninguna dificultad.	11	Gestionar a los afectados en un futuro.

Tabla 190: Test trabajador 04

En definitiva analizando las respuestas de los cuatro miembros de la fundación que realizaron el test, parece que la aplicación funciona sin problemas y cumple con las expectativas del cliente tanto en el aspecto de funcionalidad como de estabilidad.

Una vez realizados los test podemos determinar que la aplicación está lista para su paso a producción y que ha terminado el proceso de desarrollo, teniendo la versión 1.0 de la aplicación para la Fundación Madrina.

9 Conclusiones y líneas futuras

9.1 Conclusiones

El principal problema para la realización del proyecto ha venido dado por la falta de personal especializado en informática entre los miembros de la fundación que se encargan de mantener el portal web. Hay que tener en cuenta que una ONG típicamente tiene poco personal, sueldo escaso y perfiles frecuentemente no especializados en cada rama. El personal de la Fundación Madrina es excelente en la gestión de situaciones personales y las tareas asociadas a asegurar la supervivencia financiera de la institución. Sin embargo la gestión web se percibe como algo accesorio, donde el personal tiene que dedicar parte de su escaso tiempo a la mejora de una web futura. Esto ha complicado sobremanera la comunicación con la institución, demorándose innecesariamente las entregas de material y permisos. Si bien entendemos que es algo natural dado el contexto debería tenerse en cuenta al analizar los tiempos y costes que estos ocasionan.

A continuación se muestra el GANTT con la planificación estimada y planificación real:

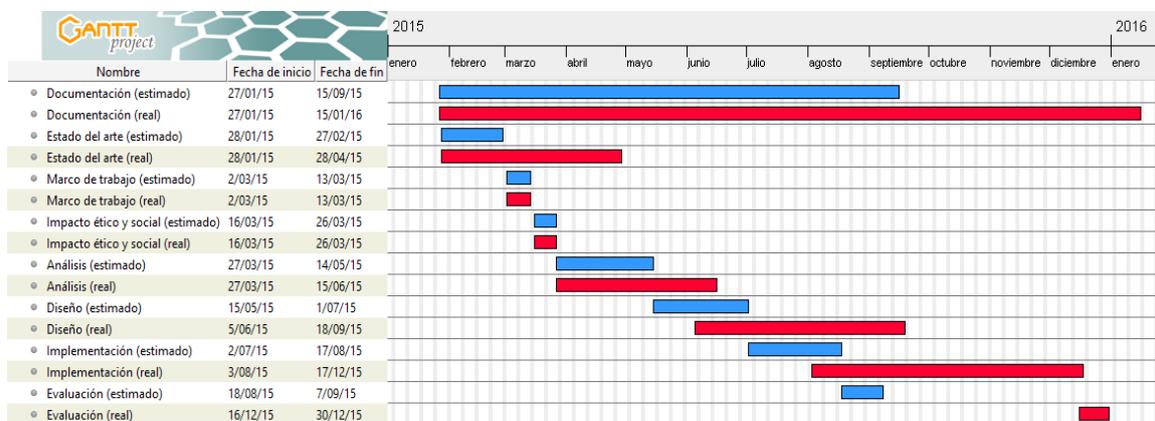


Ilustración 46: GANTT estimado-real

9.1.1 Producto

Si analizamos los objetivos que estableció el cliente al inicio del proyecto, podemos decir que han sido cumplidos de manera satisfactoria, ya que la aplicación web ha mejorado su aspecto, manejo y adaptación a cualquier dispositivo, respecto a la aplicación web inicial. Además la aplicación web inicial no tenía ningún tipo de conexión con base de datos por lo que no gestionaba ningún tipo de datos, sino que los gestionaban los miembros de la ONG con un



Excel. Gracias a este proyecto se gestiona esta información de manera clara y mucho más rápida. Anteriormente si un usuario quería hacerse voluntario de la fundación tenía que acudir personalmente o llamar vía telefonía, ahora tan solo tiene que completar un cuestionario, y tras su validación será un voluntario de la fundación.

Al comienzo del desarrollo de la aplicación trabajaba con la versión de WordPress 4.2.1, ahora mismo está actualizado a la versión WordPress 4.3.1, este proceso de actualización no requirió ningún cambio del sistema, por lo que se demuestra que la aplicación es compatible con las diferentes versiones y se actualiza de manera clara y sencilla.

9.1.2 Proceso

Para el desarrollo de este proyecto he tenido que aprender tecnologías que no conocía como PHP y profundizar los pocos conocimientos que tenía sobre la administración de servidores y metodología ágil SCRUM. A su vez he empleado muchos de los conocimientos adquiridos durante la carrera, principalmente los adquiridos en las asignaturas de Ingeniería del Software, Dirección de Proyectos de desarrollo software para generar la documentación del proyecto; Ficheros y bases de datos para la gestión y desarrollo de la base de datos; Interfaces de usuario y Tecnologías informáticas para la web para el desarrollo de la página web; Principios de desarrollo del software para la implementación de pruebas.

9.1.3 Personales

Tras la realización de este proyecto puedo decir que estoy muy orgullosa de haberlo llevado a cabo. Ha sido muy gratificante poder ayudar a una Fundación con pocos recursos en actualizarse a las nuevas tecnologías y poder actualizar su portal web, ya que inicialmente tenían una web bastante deprecada. Tras este proyecto la web está totalmente actualizada, además gestiona todo el trabajo interno de los miembros de la fundación, la gestión de los voluntarios de la ONG y las actividades a realizar.

Estoy segura de que esta aplicación la utilizarán durante mucho tiempo e iremos añadiendo en medida de lo posible todas las funcionalidades que puedan mejorar cualquier gestión de la fundación.

Este proyecto me ha ayudado a demostrar mi capacidad en convertirme en ingeniera además de ver que soy capaz de realizar un proyecto, y aprender nuevas tecnologías por mí misma.

9.2 Líneas futuras

Por último, se proponen algunas ideas sobre nuevas características que sería interesante añadir a la aplicación para ampliar su funcionalidad y mejorar los servicios ya implementados:

- **Gestión de usuarios afectados.** Se propone implementar la gestión de los afectados que acuden a por ayudar a la fundación, de manera similar a la que se han gestionado los voluntarios en el proyecto.
Esta gestión la están realizando miembros de la Universidad Europea de Madrid en paralelo al trabajo que he realizado, no obstante por diferentes motivos ajenos a mi implementación no han cumplido con los plazos establecidos, por lo que no ha sido posible presentar el proyecto con esta parte integrada.
- **Una aplicación Android/iOS.** Se propone el desarrollo de una aplicación móvil con las mismas funcionalidades que he desarrollado, de esta manera se facilitará aún más el manejo de este tipo de datos a usuarios con bajos conocimientos tecnológicos. Esta propuesta podría ser un futuro trabajo fin de grado de algún compañero/a.
- **Sistema de estadísticas completas.** El sistema permite sacar estadísticas con facilidad de accesos que ha tenido la aplicación, ver el número de voluntarios registrados en un momento, ver todas las actividades que están en curso, etc.
No obstante se propone implementar un sistema de estadísticas que saque informes mensuales o incluso semanales con toda esta información que hasta el momento se obtiene accediendo a las distintas áreas de la plataforma.
- **Gestión de preguntas frecuentes (FAQ).** Sería de gran utilidad incluir en el sistema una sección de preguntas frecuentes incluyendo las dudas que surgen



con facilidad a los usuarios. Se podría crear un módulo de administración de estas preguntas permitiendo a los usuarios de la aplicación que puedan interactuar entre ellos para resolver las dudas, en caso de que no encuentren una solución en las preguntas frecuentes. Los miembros de la fundación también podrán resolver cualquier tipo de duda en todo momento.

- **Sistema de alertas de inicio de actividades.** Sería de gran interés implementar un sistema que notifique por una alerta en el terminal móvil o email el inicio de cualquier actividad que debe realizar. De esta manera evitamos que se le olvide acudir a algunas de las citas que se han establecido.



10 Development of an Application for Non-Governmental Organizations

10.1 Introduction

The motivation for this End-of-Degree Project (EDP) is the *Fundación Madrina* need for renewing and updating its website which has become absolutely obsolete. It is not intuitive and the unclear category organization makes navigation difficult for users.

The main reason for this website abandonment is the lack of human resources owning knowledge and skills in website design and development. Despite their lack of knowledge in this field, these workers try to maintain active this obsolete website.

Taking into account this fact, *Fundación Madrina* would need an expert in this field to develop a new web application which should be more modern and accessible to people and include new functionalities as the management of its volunteers, workers and activities, so that they would work in a more efficient and easier way and, consequently, they would obtain better results.

I decided to do the project "Development of a web application for NGOs" as my End-of-Degree Project because it was very gratifying for me to help and organisation with few resources to adapt to the new technologies and to make their daily administrative and management work easier.

To sum up, this project means the culmination of my stage as Computer Engineering student and, at the same time, the results of my work as volunteer by helping this foundation to improve its web application.

10.2 Objectives

The objective of this project is to develop a web application for *Fundación Madrina*, as its current website does not reach the accessibility standards the new one should contemplate the conformance with these standards to ensure its contents can be accessed by everyone and to make easier the navigation.



In the same way, it is proposed to develop a Database that will be managed through the application, for instance, if someone wants to register as volunteer, she/he only will have to fulfil a registration form through the website and will have the option to access to his/her profile at any moment. Moreover, the foundation employees will have an account and a profile to manage the volunteers' information, to manage the activities planning, as well as to assign different tasks to every volunteer.

At the same time, this Database will be integrated with another database which is being developed by students from *Universidad Europea* in order to manage the persons in risk of exclusion supported by this foundation and also the products store.

Our Database is intended to manage the volunteership, activities and personnel aspects, whereas the students of *Universidad Europea* will add in the coming future the functionalities previously mentioned. Taking this into account, it will be developed a multi-platform database to avoid its migration to another operating system in the case that the NGO wants to change the current system.

The different tools used to develop this project have been selected according to the foundation demands and therefore they are focused on making easier the contents updating, the tasks performed by the volunteers, the management of affected beneficiaries and of the website. This aim is not easy to achieve, since the most important is to satisfy the foundation requirements by providing it with a user-friendly application easy to use.

The selection of the different components that conform the new website meet the foundation expectations for developing a new one following a structure similar to the former web, but with a more user-friendly interface and adding new functionalities as the volunteers' registration and access. The reason to conserve a similar structure is to avoid that former users find difficulties when accessing.

The functionality of users' registration and access will be completely new so that the persons who want to become volunteers will have the possibility to register themselves through the website, as a consequence it will saved a lot of time and work to the foundation members.



10.3 Abstract

This work will show my skills related to the design and development of a software engineering project as the culmination of my university studies.

This project is based on the development of a web application for the NGO *Fundación Madrina* that will make easier the management of the organisation personnel, volunteers and activities. With this project I intend to meet *Fundación Madrina* need for renewing and updating its current website which has become absolutely obsolete. The final project result will be the development of an application that will update this website by including improvements and adding new functionalities. This application will save a lot of time and work to the foundation personnel because just by accessing to they will be able to check the number of volunteers currently available, the tasks they are performing and also to add new tasks and to assign them to every volunteer. Nowadays, these tasks are performed manually by means of an Excel file without updating, so that it is difficult for them to know at any moment the availability of each volunteer. Summarising, thanks to this application, which has a more modern and user-friendly interface and, what is the most important, will have a responsive design adapted to all kind of devices (tablets, smart phones), the personnel will perform their tasks more quickly and efficiently.

To carry out this project, the first step was to consider if it was better to use open source or proprietary software. Taking into consideration that an End-of-Degree Project and the client is a NGO, it was decided to use open source software due to the lack of economic resources. In a second stage, the most popular CMS applications were analysed:

- **WordPress** is a free open-source content management system (CMS) focused to blog creation. It is based on PHP for servers running MySQL and Apache under licence and modifying code. WordPress is the most popular and powerful blogging system in use on the Web today.
- **Joomla** is a management system (CMS) for publishing dynamic and interactive web contents. Through its administration panel it is very easy to create and modify websites. It is also open source and written in PHP as WordPress. It is estimated to be the second most used content management system on the Internet, after WordPress.



- **Drupal** is a free open-source and modular content management system (CMS). Its configuration is very easy and it is possible to publish articles, images and other kind of files. It can be considered as a dynamic system since it is possible to store configuration, contents and modifications in a database instead of using a file system.

Once the three main CMS were analysed, it was decided to use WordPress because it was a foundation requirement. They wanted to update the application by themselves.

The next project stage was established in a framework by following Agile SCRUM methodologies. There are three main of roles according to this methodology:

- **Product owner:** foundation member who will define the requirements and will evaluate the project development to ensure the right evolution and results.
- **Scrum master:** project tutor responsible for checking the established project milestones and terms as well as the goals established in each sprint.
- **Team:** this is my role and I have been responsible for the project development.

The project was divided in several *Sprints*, each of them with an approximate duration of 4 weeks. In each sprint different tasks of our *product backlog* were established. After each sprint a meeting was held to analyse its goals and to add new functionalities to our *product backlog* and to fix the tasks of each new *sprint*, known as *sprint backlog*.

The project planning has been strongly defined according to the client requirements and those of my End-of-Degree Project with a value of 12 credits; each of them is equivalent to 25 work hours, that is, a total of 300 hours approximately. Taking into account this fact, it has been planned a 10-hour-work week approximately so that the whole project duration will be more 33 weeks. The project timing has been calculated considering some work interruptions in summer, some possible delays on the client's part and my time limitations since I had a full-time contract during the project development.

The following Gantt diagram shows the project planning. As it can be observed, the project was expected to start the 27th January of 2015 and to finish the 15th September of the same year.

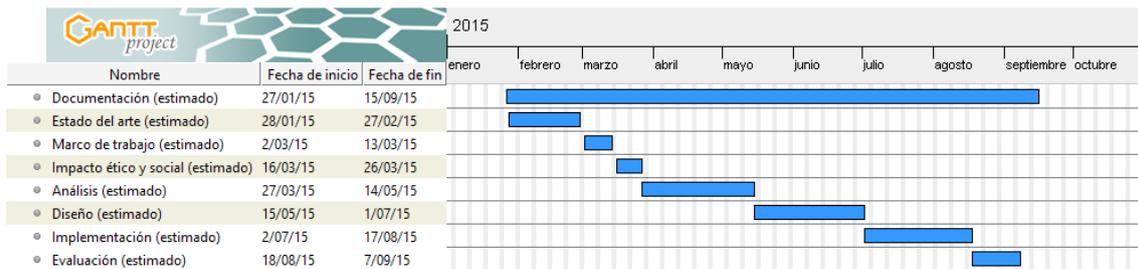


Illustration 47: GANT

Finally, once the appropriate calculations were made, it was estimated a total project cost of 21.578,66 € (TWENTY ONE THOUSAND FIVE HUNDRED SEVENTY-EIGHT EUROS AND SIXTY-SIX CENTS) without VAT. It was also stipulated a payment of the 30% in advance, with monthly payments of 10%. Once the final application was delivered, a final payment was made to complete the total project budget.

For further information, the payment scheme is attached:

- Initial payment (30%): 6473,60€*
- First, second, third and fourth monthly payment (10% each of them): 2157,87€*
- Final payment (30%): 6473,60€*

* VAT not included

PROJECT BUDGET
Direct Costs
15.794,66 €
Indirect Costs (8%)
17.058,23 €
Contingency Margin (10%)
18.764,05 €
Profit Margin (15%)
21.578,66 €

Table 191: Project Budget

Once the framework was established, a study was performed dealing with similar NGO websites. The next step was to carry out an exhaustive analysis concerning with functionalities, users interface, performance, operations, resources, testing, acceptance testing, damage, security and maintainability. Once all of them were analysed, the main use cases of the application were studied.

At the design stage, the system architecture was defined. It was decided to implement a Model View Controller architecture divided in the following subsystems:

- **Model:** it directly manages the data, logic and rules of the application and contains the data which represent relevant domain objects.
- **View:** users interact with this system part which manages the interface and communicates with the model and the controller, apart from showing the final application to users. Multiple views of the same information are possible.
- **Controller:** this part of the system accepts input and converts it to commands for the model or view. These are its functions: activities, volunteers and employees management, access to information and notifications.

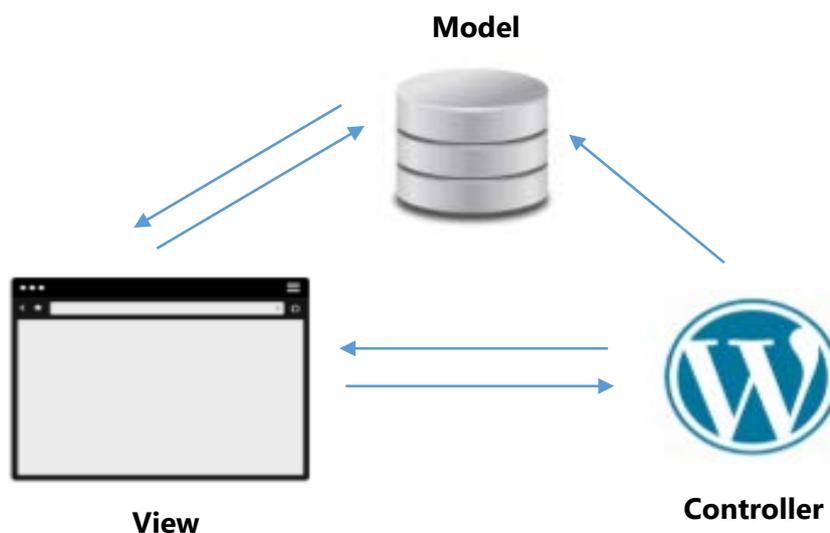


Illustration 48: Model View Controller

Over this stage, the standards and rules of design and construction were specified, as well as the exceptions that could be found in the system. It was also designed the interface operation and taxonomy, as well as the classes in which the application was going to be structured. To finish this project phase, the database composed of 8 tables was designed.

The system implementation and implantation had as aim the delivery and final acceptance of the system and the performance of all the necessary tasks to ensure its appropriate implantation.

The system implementation plan was defined and structured according the following phases:

- Training required for the implantation aimed at the system end users as well as at the team responsible for defining the implantation and acceptance tests.

- Preparation of the infrastructure for the system integration into the operating environment.
- Components setup and development of the necessary proceedings for the system integration into the operating environment.
- Initial data uploading.
- Implantation and acceptance tests realisation.
- Maintainability plan definition.

Two System Operation Manuals have been elaborated according to the different application users:

- **Employee's Manual:** this manual includes information on how to manage and work with the different applications elements.
- **Volunteer's Manual:** embracing all the information on the different tasks that can be performed by this kind of user through the application.

As it is well known, it is necessary to prepare in every project its system maintainability. First of all, it is of utmost importance to establish a test environment in which an exhaustive control of each testing is performed in order to guarantee the appropriate application operation. In the case all the tests are right, the next stage is to prepare the pre-production testing environment. At this stage, functional testing is carried out during two months to ensure the correct system operation. Finally, the production environment is prepared to monitor the possible modifications or updates performed until the new modification is effective and stable. If the testing results are successfully passed, the production environment will be updated. The following scheme shows the incidences control flow:

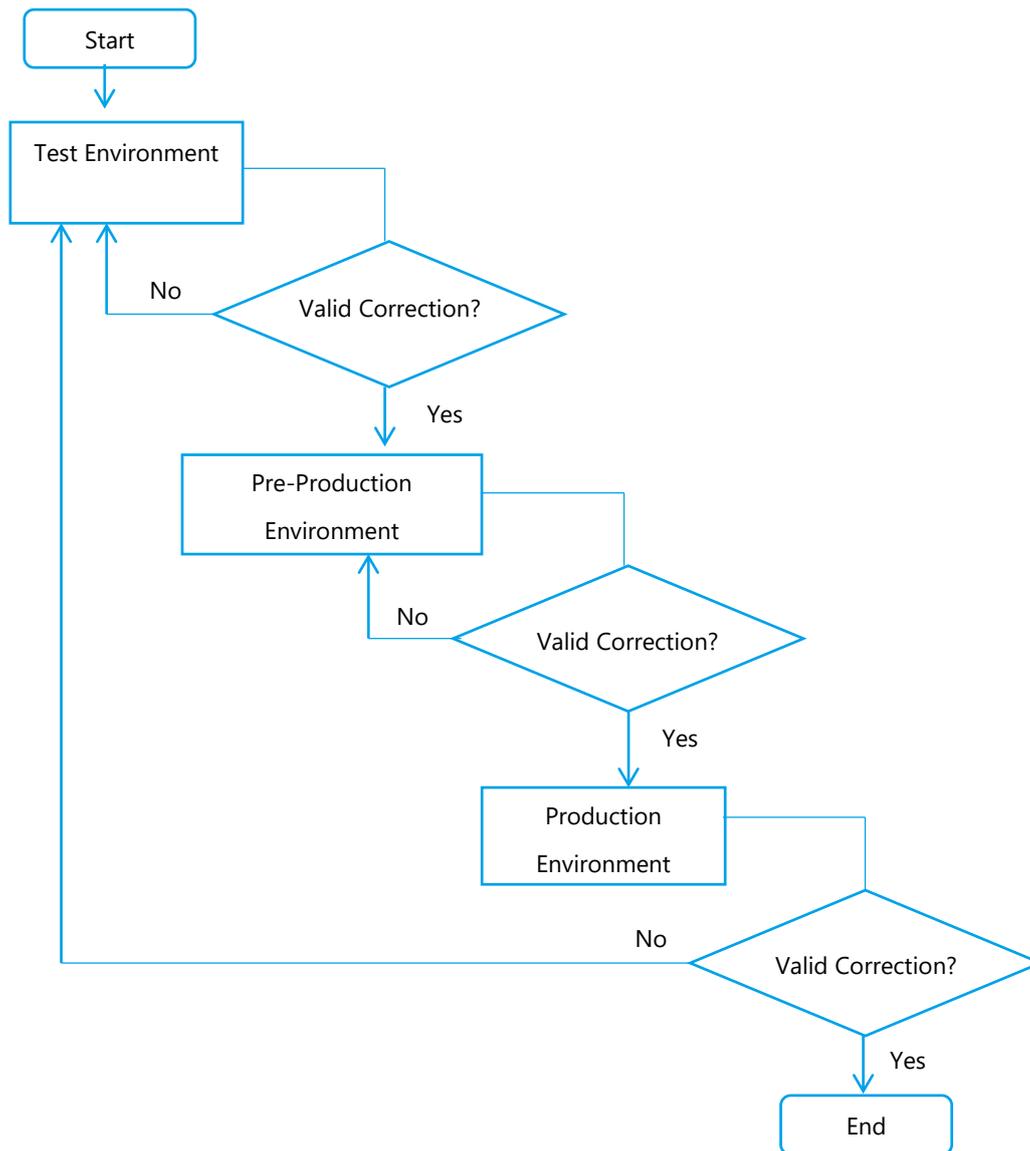


Illustration 49: Control Flow

On one hand, it has been established a commitment with *Foundation Madrina* to give face to face or telephone assessment to its application maintenance team, if necessary. This support will last six months since the project end, but after this period it is not guaranteed.

On the other hand, any application error or inconsistencies that could emerge during the adaptation period due to a bad system design will be solved by the developers for free.

The tests performed to guarantee the correct application operation and to advance towards the pre-production stage, which is the current project phase, are the following:

- **Unit testing:** to check the correct operation of the code model. We can find two kinds of testing: black-box or white-box testing.
- **Integration Testing:** these tests are performed in the framework of the software development, once all the unit tests have been approved.
- **System Testing:** this is to evaluate the system's compliance with its specified requirements. It conducts testing of whole software product to identify the errors in software.
- **System Implementation Testing:** its main objective is to verify the right system operation.
- **System Acceptance Testing:** its aim is to check if the software can be considered as acceptable and if it is ready for production.
- **System Usability Testing:** thanks to these tests we can check software product usability in order to proceed to the production phase.

To finish, satisfaction surveys were made to 10 users (volunteers) and 4 foundation employees. Once they were fulfilled, we could observe that the application works without problems and satisfies the client's expectations in both functionality and stability aspects.

10.4 Conclusions and Future Work

10.4.1 Conclusions

The main problem is due to the lack of foundation personnel skilled in IT in order to manage and maintain the website. It must be taken into account that NGOs rarely have lots of employees highly qualified and specialised in each field of knowledge because their wages are low.

The personnel of *Fundación Madrina* are very efficient in the management of personal situations and tasks related to ensure the financial survival of this organization. However, the web management is not considered a relevant task so that it is difficult for them to accept that they must devote part of their work time to improve the future web. This fact has made difficult the communication with the organization and has delayed unnecessarily the deliveries of material and the permissions to develop the project. Even if we accept this problem as normal due

to the context in which we work, this should be taken into consideration when analysing the delays entailed and the increase of costs resulting from it.

The following GANTT diagram shows the estimated project planning and the real one:

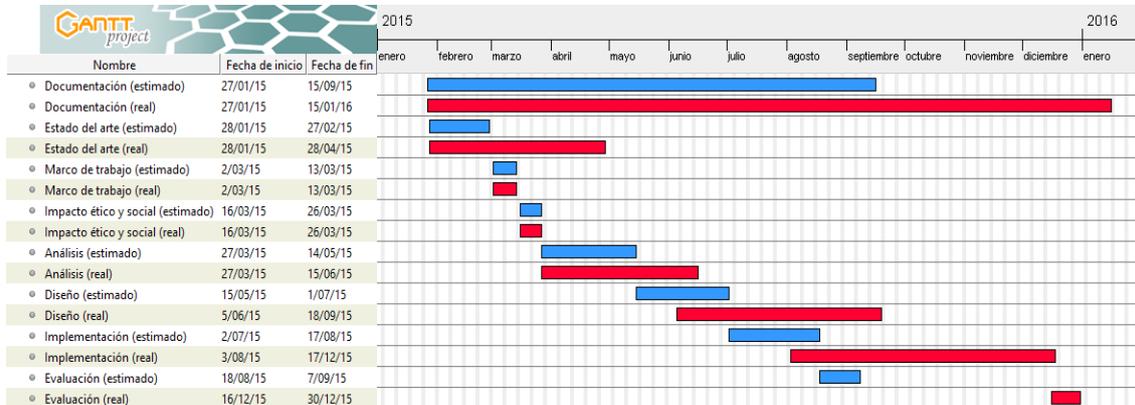


Ilustración 50: GANTT estimated-real

10.4.1.1 Product

If we analyse the requirements established by our client at the project starting, we can affirm that they have been accomplished in a satisfactory way, since the application interface, management and responsive design have been improved in comparison with the previous version. Thus the previous version was not connected to a database so that it was impossible for their employees to manage data; they just managed an Excel file.

Thanks to this project, the information is now managed in clearer and faster way. Formerly, if user wanted to become foundation volunteers they had to do it face to face or by phone, nowadays they just have to fulfil a web form to achieve it.

I started to work in the application development with the WordPress version 4.2.1, nowadays the application has been upgraded to the WordPress 4.3.1 version. This process of updating did not require any system change and this is a proof that the application is compatible with the different versions and can be updated in a simple and easy way.

10.4.1.2 Process

In order to develop this project I have had to learn unknown technologies for me such as PHP and too deep in the fields of server management and SCRUM Agile Methodology. At the same time, I have had the opportunity to apply most of the knowledge and competences acquired over my university education, mainly those gained through the subjects of Software Engineering, Management of Software Development Projects in order to generate the project documentation, files and databases to manage and develop the project database and Information Computer Technologies to develop the new foundation website and, finally, Fundamentals of Software Development for testing implementation.

10.4.1.3 Personnel

After having finished this Project, I felt very proud because it was very gratifying for me to help and organisation with few resources to adapt to the new technologies and to renew its web, which was much deprecated.

Nowadays, thanks to this project, its web is totally renewed and updated, thus it serves to manage the tasks of the foundation employees and the planning of the NGO volunteers and activities.

I am sure that this application will be very useful for a long time since we will progressively add, if possible, all the functionalities that can improve any foundation management process.

Finally, I would like to highlight that this project has help me to show my skills and knowledge as engineer apart from demonstrating that I am able to manage and develop a project and to learn and apply new technologies all by myself.

10.4.2 Future Work

To conclude, I propose some new features which would be very interesting to integrate in the application in order to extend its functionality and to improve the services already implemented:



- **Management of users.** It is proposed to implement the management of the information concerning the foundation users or beneficiaries in the same way we have already implemented the management of volunteers.
- This part of the project is being developed by *Universidad Europea de Madrid* in parallel with the work I have performed, but it has been impossible for them to meet the established project deadlines, so that this part has not been finally integrated in the project.
- **Android/IOS application.** It is suggested the development of a mobile application with the same functionalities I have already implemented, so that it will make easier the information access for users with low technology skills. This proposal could be the subject of the End-of-Degree Project of any student.
- **Complete Statistics System.** The current system allows to extract application access statistics easily and to know the number of volunteers registered the activities which are being currently developed, etc. Nevertheless, we propose to implement a statistics system which allows extracting monthly even weekly reports including all this information that, till now, was being obtained by accessing to the different platform areas.
- **Management of Frequently Asked Questions (FAQ).** It would be very useful to include a FAQ section in the system in order to solve the users' doubts. A FAQ administration module could be integrated that would allow the interaction among users in order to solve those doubts not included in the FAQ section. The foundation employees could also solve any kind of doubt at any moment.
- **Activities Alert System.** It would be of utmost importance to implement an alert system to notify the starting of any activity via mobile device or email and, consequently, we would avoid forgetting the established dates.



Bibliografía

- [1] Diez ventajas del software libre y propietario [en línea]. José [Consulta: 10 de Febrero 2015]. URL: <http://www.abadiadigital.com/diez-ventajas-del-software-libre-y-propietario/>
- [2] *Drupal – Open Source CMS* [en línea] (s.f.) [Consulta: 12 de Febrero 2015]. URL: <https://www.drupal.org/>
- [3] *WordPress vs Joomla! vs Drupal* [en línea] (s.f.) [Consulta: 12 de Febrero 2015]. URL: <http://mongemalo.es/wordpress-joomla-drupal-mejor-gestor-contenidos-cms/>
- [4] *Joomla! The CMS Trusted by Millions for their Websites* [en línea] (s.f.) [Consulta: 12 de Febrero 2015]. URL: <http://www.joomla.org/>
- [5] *Joomla! Extension Directory* [en línea]. *Schedule School* [Consulta: 12 de Febrero 2015]. URL: <http://extensions.joomla.org/browse/most-reviewed>
- [6] WordPress [en línea] (s.f.) [Consulta: 12 de Febrero 2015] URL: <https://wordpress.org/>
- [7] Manual PHP [en línea]. Mehdi Achour, Friedhelm Betz, Antony Dovgal, Nuno Lopes [Consulta: 20 de Febrero 2015]. URL: <http://php.net/manual/es/>
- [8] Historia de la Web [en línea] (s.f.) [Consulta: 20 de Febrero 2015]. URL: http://www.mclibre.org/consultar/htmlcss/otros/otros_historia.html
- [9] Introducción a CSS [en línea]. Javier Eguíluz Pérez [Consulta: 22 de Febrero 2015]. URL: http://dis.um.es/~lopezquesada/documentos/IES_1213/LMSGI/curso/UT5/libroswebcss/www.librosweb.es/css/index.htm
- [10] Ventajas de CSS [en línea]. Albert Balo Pereira [Consulta: 23 de Febrero 2015]. URL: <http://cv.uoc.edu/~abalo/Practica1/ventajas.html>
- [11] Aulaclick – Introducción a las bases de datos [en línea] (s.f.) [Consulta: 25 de Febrero 2015]. URL: http://www.aulaclip.es/sqlserver/b_1_1_4.htm



[12] Fundamento y diseño de base de datos [en línea]. Diana Fernanda Triviño Berbeo [Consulta: 25 de Febrero 2015]. URL:

<http://www.monografias.com/trabajos82/fundamento-y-diseno-base-datos/fundamento-y-diseno-base-datos2.shtml>

[13] Bases de datos en MySQL [documento en línea]. L. A. Casillas Santillán, M. Gibert Ginestà, O. Pérez Mora [Consulta: 25 de Febrero 2015]. URL:

http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02151.pdf

[14] *Wordpress Plugins* [en línea] (s.f.). [Consulta: 15 de Marzo 2015]. URL:

<https://wordpress.org/plugins/>

[15] Calcula y compara tu sueldo en España [en línea] tusalario.es [Consulta: 10 de Mayo 2015]. URL: <http://www.tusalario.es/main/carrera/funcion-y-sueldo/analistas-de-sistemas>

[16] Retenciones IRPF 2015 [en línea]. Agencia Tributaria [Consulta: 11 de Mayo 2015].

URL: <https://www2.agenciatributaria.gob.es/wcl/PRET-R150/index.zul>

[17] Bases de cotización seguridad social [en línea]. Tesorería General de la Seguridad Social [Consulta: 11 de Mayo 2015]. URL: [http://www.seg-](http://www.seg-social.es/Internet_1/Trabajadores/CotizacionRecaudaci10777/Regimenes/RegimenGeneraldeLaS10957/InformacionGeneral/index.htm)

[social.es/Internet_1/Trabajadores/CotizacionRecaudaci10777/Regimenes/RegimenGeneraldeLaS10957/InformacionGeneral/index.htm](http://www.seg-social.es/Internet_1/Trabajadores/CotizacionRecaudaci10777/Regimenes/RegimenGeneraldeLaS10957/InformacionGeneral/index.htm)

[18] Métrica versión 3 [documento]. Universidad Carlos III de Madrid. [Consulta: 15 de Junio 2015].

Glosario de términos

API: Interfaz de Programación de Aplicaciones.

C: lenguaje de programación.

CMS: *Content Management System* o Sistema de Gestión de Contenidos.

Codec: programa o dispositivo hardware capaz de codificar o decodificar una señal o flujo de datos digitales,

Copyright: derechos de autor.

C++: lenguaje de programación que extiende de C.

DRM: *Digital Rights Management* o gestión digital de derechos.

Eiffel: lenguaje de programación orientado a objetos que sigue el estándar ISO.

Estándar de facto: patrón o norma que se caracteriza por no haber sido consensuada ni legitimada por un organismo de estandarización al efecto.

Feedback: también conocido como retroalimentación.

GPL: Licencia Pública General de GNU.

HTML: lenguaje de programación web.

ISO: Organización Internacional de Normalización.

Java: lenguaje de programación orientado a objetos.

Linux: término empleado para referirse al a combinación de núcleo o kernel libre similar a Unix con sistema operativo GNU.

Logeado: traducción inglesa empleada para determinar que un usuario ha accedido al sistema con su usuario y contraseña.

Logearse: traducción inglesa empleada para determinar la acción de iniciar sesión.

Login: traducción inglesa empleada para denominar el proceso de iniciar una sesión mediante la introducción de un usuario y una contraseña.



Logout: traducción inglesa empleada para determinar la acción de cerrar sesión.

MS SQL Server: servidor de base de datos de Microsoft.

MySQL: sistema de base de datos.

ODBC: *Open DataBase Connectivity*, es un estándar de acceso a las bases de datos.

Open Source: código libre.

Perl: lenguaje de programación similar a C.

PHP: lenguaje de programación web del lado del servidor.

PostgreSQL: sistema de base de datos.

Python: lenguaje de programación interpretado.

Ruby: lenguaje de programación interpretado, reflexivo y orientado a objetos.

Royalties: pago que se efectúa al titular de derechos de autor.

SEO: *Search Engine Optimization* o posicionamiento en buscadores.

SCRUM: tipo de metodología ágil.

SGBS: sistema gestor de bases de datos.

Spyware: programa espía.

TCL: *Tool Command Language* o lenguaje de herramientas de comando.

TCP/IP: protocolo de red.

Unix: sistema operativo portable, multitarea y multiusuario.

XML: *eXtensible Markup Language* o lenguaje de marcas extensible.

Anexo I: Manual de voluntario

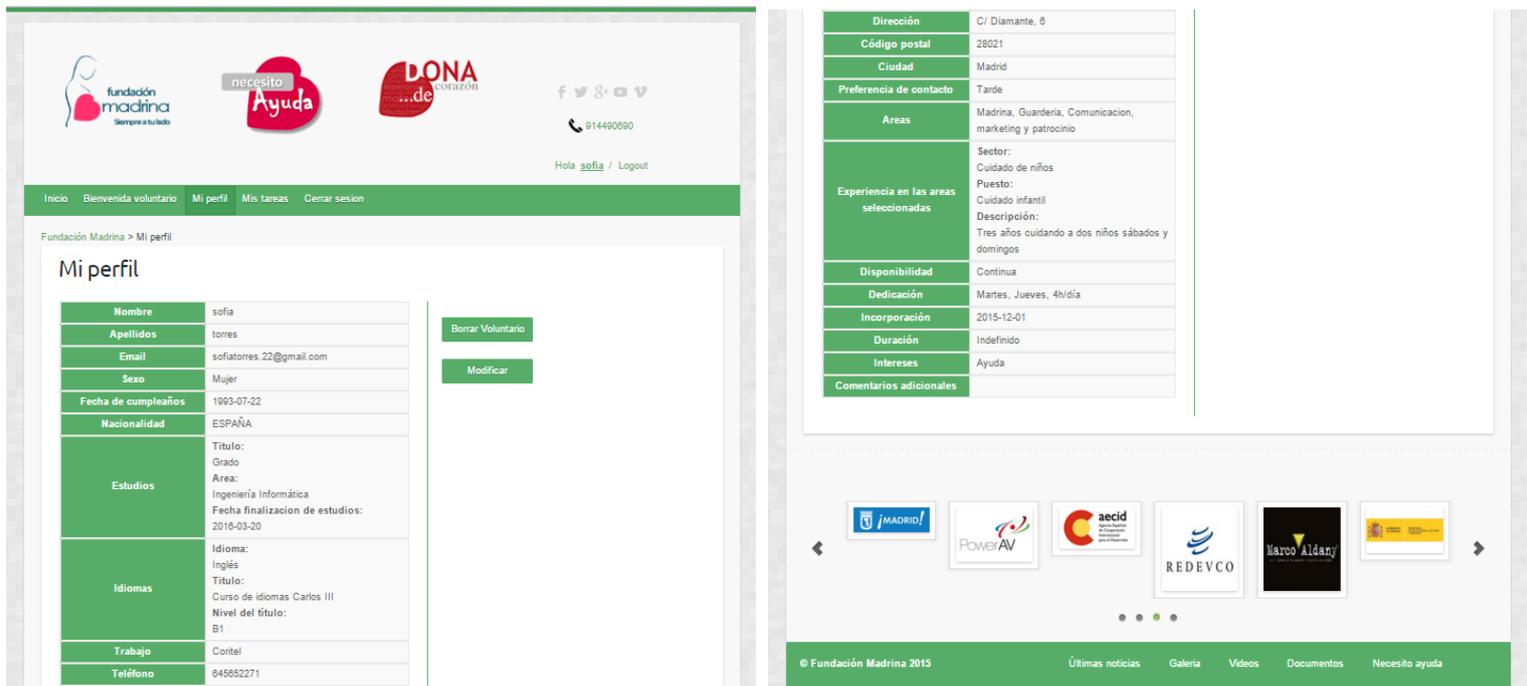
El siguiente manual va dirigido a los voluntarios de la fundación. Se entiende que este usuario no tiene conocimientos avanzados de informática, por lo tanto se explicará detalladamente el uso de la aplicación web.

Registro como voluntario

Para registrarse como voluntario hay que completar un formulario, todos los campos son obligatorios, excepto comentarios. El programa identifica si falta algún campo por completar, si la longitud no es permitida, o si los caracteres no son válidos.

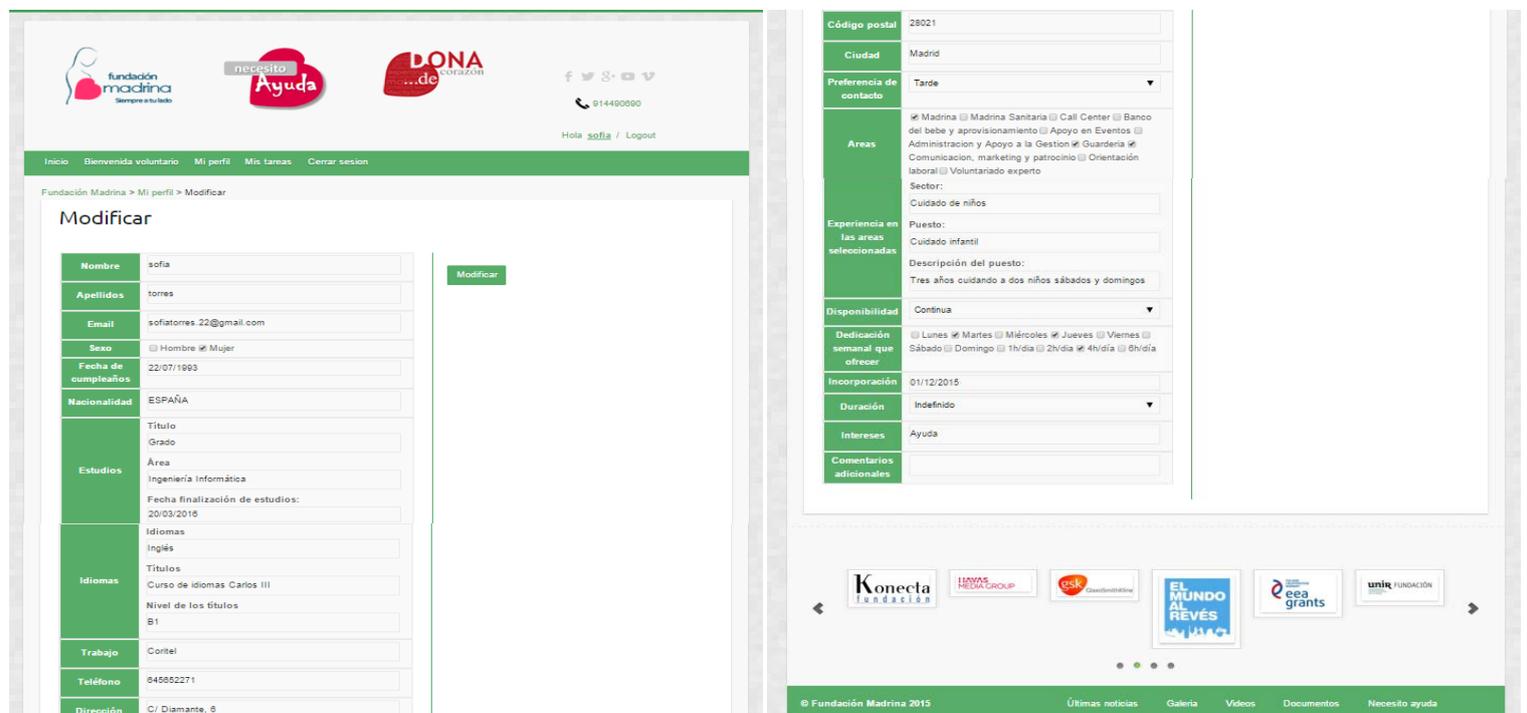
Ver perfil

Para ver los datos de tu perfil, tienes que acceder a la página mi perfil, en ella encontraras una tabla con todos los datos que metiste en el formulario de registro. En la parte derecha de la pantalla encontrará dos botones: modificar y borrar.



Dirección	C/ Diamante, 8
Código postal	28021
Ciudad	Madrid
Preferencia de contacto	Tarde
Áreas	Madrina, Guardería, Comunicación, marketing y patrocinio
Experiencia en las áreas seleccionadas	Sector: Cuidado de niños Puesto: Cuidado infantil Descripción: Tres años cuidando a dos niños sábados y domingos
Disponibilidad	Continua
Dedicación	Martes, Jueves, 4h/día
Incorporación	2015-12-01
Duración	Indefinido
Intereses	Ayuda
Comentarios adicionales	

- Si pulsa sobre **modificar**, se abre una nueva página similar a la de registro, pero con los campos completos con los datos que hay almacenados en la base de datos. En el formulario debe cambiar cualquier dato que desee y se procederá a una validación de todos los campos comprobando que estén completos todos los campos obligatorios, las longitudes sean correctas y los caracteres introducidos son los adecuados.



Código postal	28021
Ciudad	Madrid
Preferencia de contacto	Tarde
Áreas	<input checked="" type="checkbox"/> Madrina <input type="checkbox"/> Madrina Sanitaria <input type="checkbox"/> Call Center <input type="checkbox"/> Banco del bebe y aprovisionamiento <input type="checkbox"/> Apoyo en Eventos <input type="checkbox"/> Administración y Apoyo a la Gestión <input type="checkbox"/> Guardería <input type="checkbox"/> Comunicación, marketing y patrocinio <input type="checkbox"/> Orientación laboral <input type="checkbox"/> Voluntariado experto
Experiencia en las áreas seleccionadas	Sector: Cuidado de niños Puesto: Cuidado infantil Descripción del puesto: Tres años cuidando a dos niños sábados y domingos
Disponibilidad	Continua
Dedicación semanal que ofrecer	<input type="checkbox"/> Lunes <input type="checkbox"/> Martes <input type="checkbox"/> Miércoles <input type="checkbox"/> Jueves <input type="checkbox"/> Viernes <input type="checkbox"/> Sábado <input type="checkbox"/> Domingo <input type="checkbox"/> 1h/día <input type="checkbox"/> 2h/día <input type="checkbox"/> 4h/día <input type="checkbox"/> 8h/día
Incorporación	01/12/2015
Duración	Indefinido
Intereses	Ayuda
Comentarios adicionales	

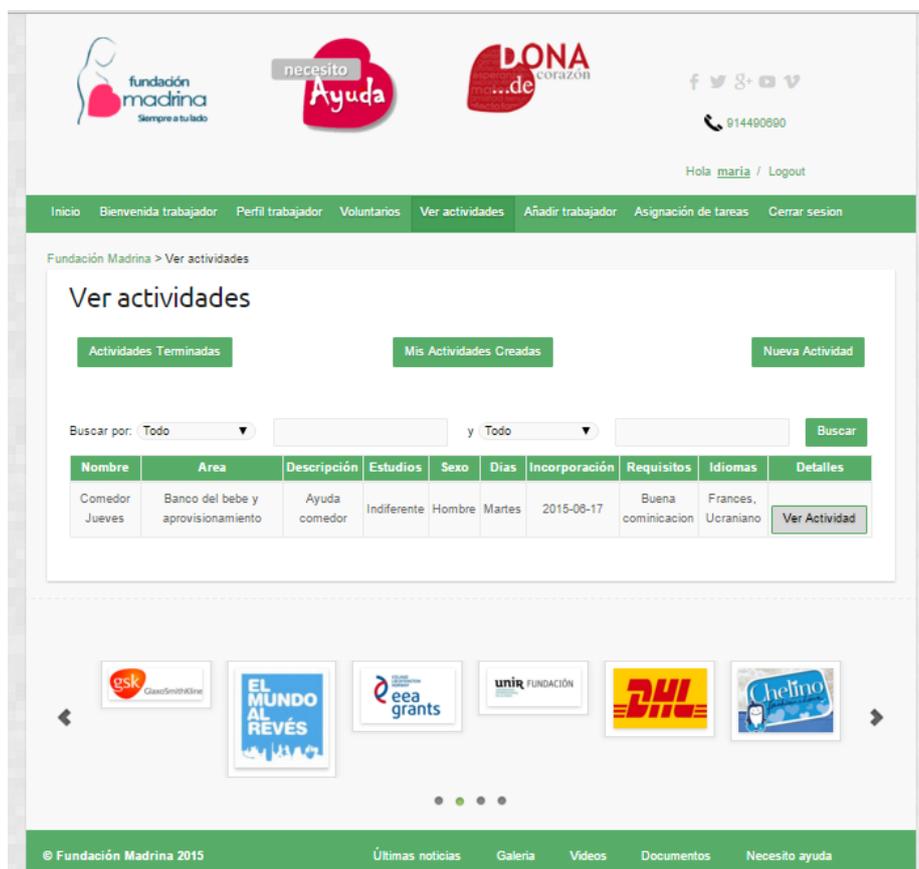
- Si pulsa sobre **borrar voluntario**, eliminará su usuario de la base de datos.

Anexo II: Manual de trabajador

El siguiente manual va dirigido a los trabajadores de la fundación. Se entiende que estos usuarios no tienen conocimientos avanzados de informática, por lo tanto se explicará detalladamente el uso de la aplicación web.

Ver datos

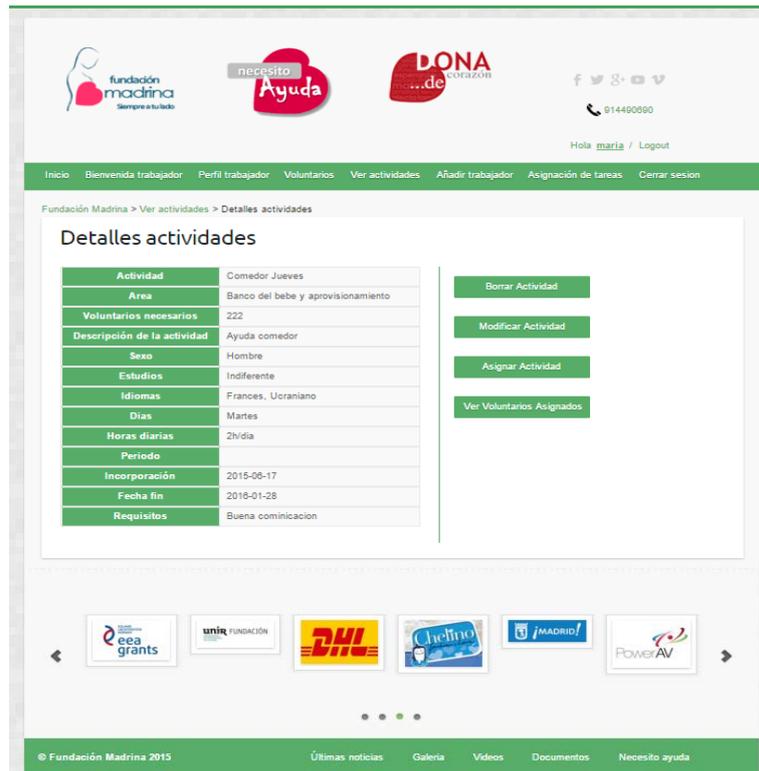
Para ver los datos de una actividad, hay que acceder a la página ver actividades, en ella encontraras una tabla con un listado de actividades. Para ver los detalles de una de ellas hay que pinchar sobre el botón de la última columna de la tabla que pone Ver actividad.



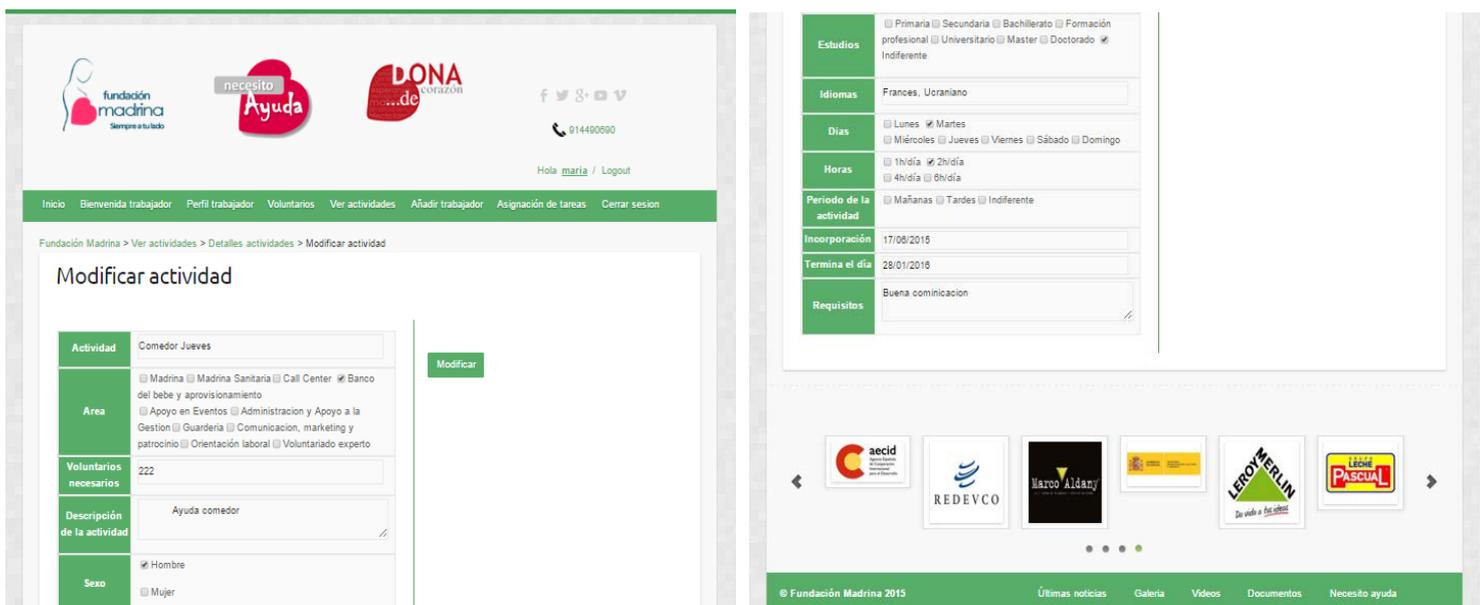
The screenshot shows the 'Ver actividades' page. At the top, there are logos for 'fundación madrina', 'necesito Ayuda', and 'DONA ...de corazón'. Below the logos is a navigation menu with items: Inicio, Bienvenida trabajador, Perfil trabajador, Voluntarios, Ver actividades (highlighted), Añadir trabajador, Asignación de tareas, and Cerrar sesión. The main content area has a title 'Ver actividades' and three buttons: 'Actividades Terminadas', 'Mis Actividades Creadas', and 'Nueva Actividad'. Below these is a search bar with 'Buscar por: Todo' and a 'Buscar' button. A table lists activities with columns: Nombre, Area, Descripción, Estudios, Sexo, Dias, Incorporación, Requisitos, Idiomas, and Detalles. The first row shows an activity named 'Comedor Jueves' in the 'Banco del bebe y aprovisionamiento' area, with a 'Ver Actividad' button in the 'Detalles' column. At the bottom, there is a footer with '© Fundación Madrina 2015' and links for 'Últimas noticias', 'Galería', 'Videos', 'Documentos', and 'Necesito ayuda'.

Nombre	Area	Descripción	Estudios	Sexo	Dias	Incorporación	Requisitos	Idiomas	Detalles
Comedor Jueves	Banco del bebe y aprovisionamiento	Ayuda comedor	Indiferente	Hombre	Martes	2015-06-17	Buena comunicacion	Frances, Ucrainiano	Ver Actividad

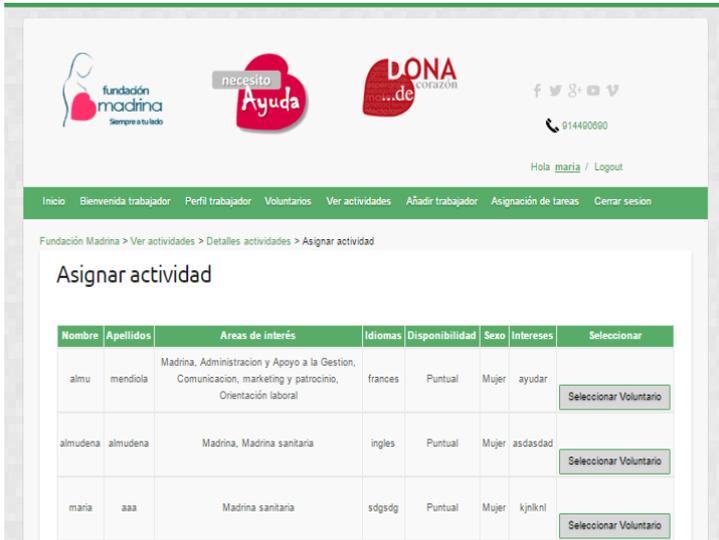
Si accedemos a los detalles de la actividad, en la parte izquierda de la pantalla encontramos una tabla con los detalles de la actividad, y en la parte derecha cuatro botones.



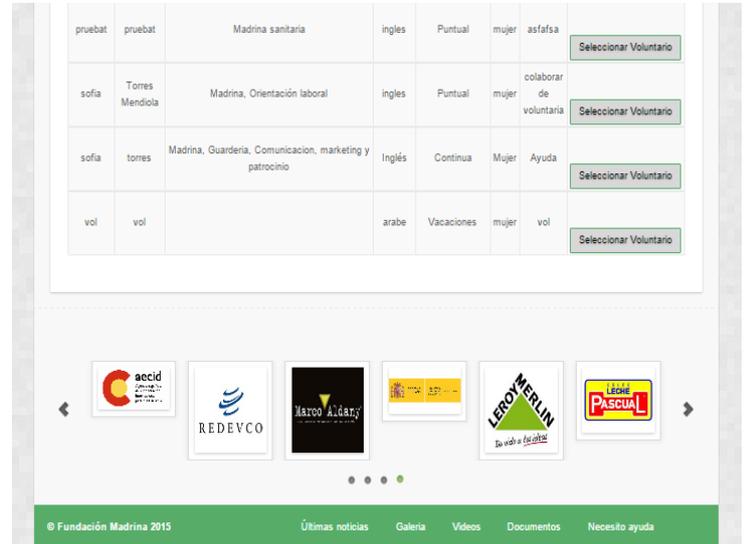
- Si pinchamos en **modificar actividad**, accedemos a un formulario que recoge los datos de la actividad almacenados en la base de datos, en dicho formulario podemos realizar los cambios que deseemos. Al guardar los datos, se procederá a una validación de los campos del formulario, en los que se comprueba que estén todos los campos completos, tengan una longitud adecuada y caracteres permitidos.



- Si pinchamos sobre el botón **borrar actividad**, eliminamos la actividad de la base de datos
- Si pinchamos en el botón **asignar actividad**, accedemos a una nueva página que muestra una lista con todos los voluntarios, buscamos el voluntario deseado y pinchamos en asignar tarea.



Nombre	Apellidos	Areas de interés	Idiomas	Disponibilidad	Sexo	Intereses	Seleccionar
almu	mendiola	Madrina, Administración y Apoyo a la Gestión, Comunicación, marketing y patrocinio, Orientación laboral	frances	Puntual	Mujer	ayudar	Seleccionar Voluntario
almudena	almudena	Madrina, Madrina sanitaria	ingles	Puntual	Mujer	asdasdad	Seleccionar Voluntario
maria	aaa	Madrina sanitaria	sógsóg	Puntual	Mujer	kjnlkl	Seleccionar Voluntario



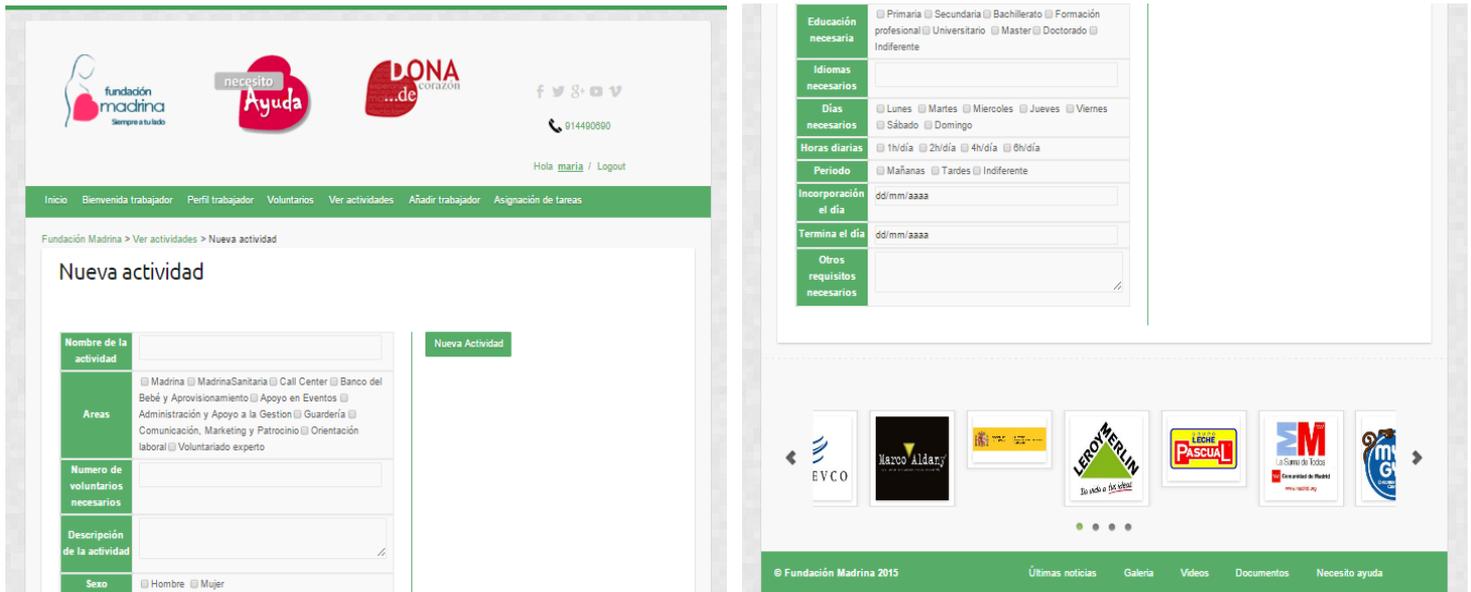
prebat	prebat	Madrina sanitaria	ingles	Puntual	mujer	asfafa	Seleccionar Voluntario
sofia	Torres Mendiola	Madrina, Orientación laboral	ingles	Puntual	mujer	colaborar de voluntaria	Seleccionar Voluntario
sofia	torres	Madrina, Guardería, Comunicación, marketing y patrocinio	Ingles	Continua	Mujer	Ayuda	Seleccionar Voluntario
vol	vol		arabe	Vacaciones	mujer	vol	Seleccionar Voluntario

Cuando pinchamos en seleccionar voluntario, se comprueba el cómputo de horas asignadas que tiene este voluntario. Si las horas semanales superan 8 horas diarias o 40 horas semanales, no se podrán asignar más actividades a este voluntario porque incumpliríamos la jornada máxima de trabajo establecida en el estatuto de los trabajadores. A su vez se comprueba que una tarea no se pueda asignar más de una vez a un mismo voluntario.

- Si pinchamos en el botón de **ver voluntarios asignados**, nos mostrará una lista con los voluntarios asignados a la actividad seleccionada.

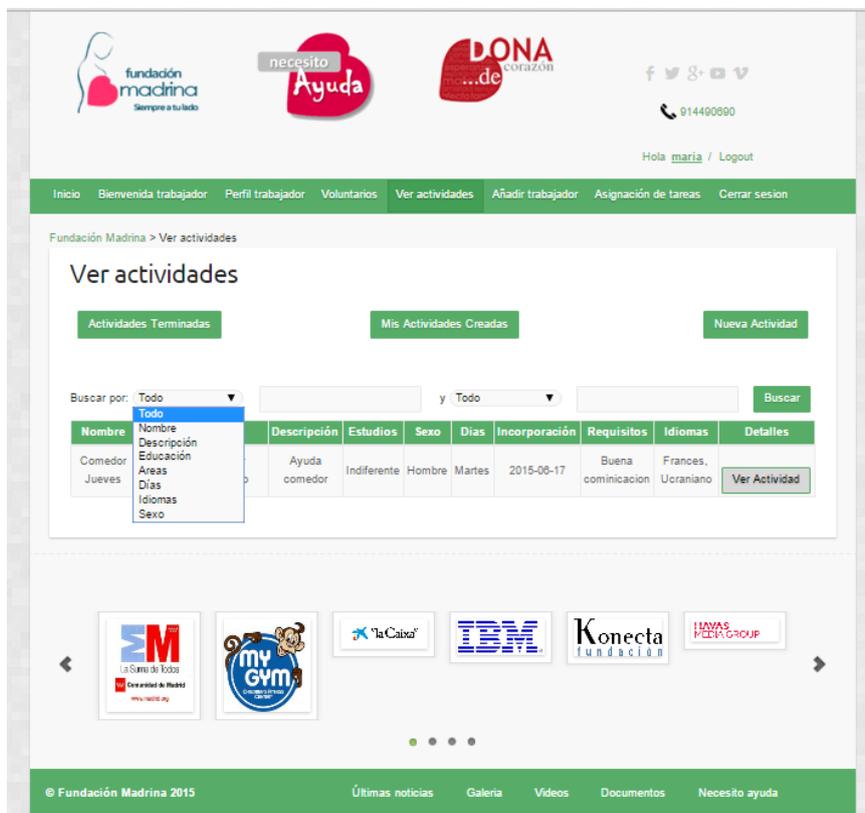
Insertar actividad

Para insertar una nueva actividad en la base de datos, hay que completar el formulario de insertar actividad. En el formulario hay que completar todos los campos que sean obligatorios. El sistema comprobará que estén completos todos los campos necesarios, además validará los datos introducidos comprobando que la longitud y caracteres introducidos cumplen el formato establecido. En caso de que no sea el correcto se mostrará un mensaje por pantalla.



Buscar actividad

Se pueden realizar búsquedas de una actividad determinada en la página de ver actividades. Para ello tenemos un buscador que permite buscar por una serie de parámetros, en los que introduciremos su valor en un campo de texto. Este buscador realizará la consulta en la base de datos mostrando todos los registros que coincidan con estos valores.



Otras funciones

En la aplicación web también encontramos un formulario de insertar voluntario, que funciona igual que insertar actividad.

Podemos ver un apartado de voluntarios, que estará formado por una lista con todos los voluntarios (igual que actividades). Podemos ver los detalles de un voluntario, desde los detalles podremos borrar y modificar los datos (igual que en detalles de actividad).

En el apartado de perfil, el trabajador podrá ver sus datos de usuario almacenados. En esta página encontramos dos botones uno de modificar el perfil y otro de borrar para darse de baja como trabajador de la fundación.

Por último encontramos un apartado de ver actividades asignadas, que muestra un listado con cada actividad y voluntario asignados.

- Fin del documento -