



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Diseño y desarrollo de un portal web para el trabajo colaborativo

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Lidia Contreras Ochando

Tutor: Germán Vidal Oriola

2013/2014

Resumen

En este trabajo se plantea una solución para los profesionales que tienen proyectos o desean participar en ellos. Este portal permite gestionar los proyectos, las tareas, los archivos y sus colaboradores de un modo sencillo y accesible desde cualquier dispositivo con navegador web.

Palabras clave: proyectos, tareas, portal web, web colaborativa, web adaptativa.

Abstract

In this work, we develop a solution for professionals willing to manage and/or collaborate in projects. The system allows one to create and manage projects, tasks, files and collaborators, in a simple and accessible way from any device using a web browser.

Keywords: projects, tasks, web portal, collaborative web, responsive web.

Tabla de contenidos

1	Introducción	7
1.1	Contexto.....	7
1.2	Objetivos del proyecto	7
1.3	Estructura del documento	7
2	Especificación de requisitos.....	9
2.1	Introducción	9
2.1.1	Propósito	9
2.1.2	Ámbito	9
2.1.3	Definiciones, acrónimos y abreviaturas	9
2.1.4	Visión global	9
2.2	Descripción general	10
2.2.1	Perspectiva del producto	10
2.2.2	Características de usuario	10
2.2.3	Operaciones permitidas.....	10
2.2.4	Restricciones generales	11
2.2.5	Supuestos y dependencias.....	11
2.3	Requisitos específicos	11
2.3.1	Requisitos de interfaces externas	11
2.3.2	Requisitos funcionales.....	12
2.3.3	Restricciones de diseño	14
2.3.4	Atributos.....	14
3	Análisis.....	15
3.1	UML.....	15
3.1.1	Diagrama de clases	15
3.1.2	Diagramas de casos de uso	18
4	Diseño e implementación	22
4.1	Arquitectura del sistema.....	22
4.1.1	Nivel de presentación	23
4.1.2	Nivel de aplicación.....	37
4.1.3	Nivel de persistencia.....	38
4.2	Tecnologías utilizadas.....	38

4.2.1	HTML	39
4.2.2	CSS.....	39
4.2.3	PHP.....	40
4.2.4	JavaScript	40
4.2.5	Mysql	40
4.2.6	Wordpress	40
4.2.7	AJAX.....	41
4.2.8	JSON.....	41
4.3	Herramientas utilizadas	42
4.4	Implementación detallada	42
4.4.1	Capa de presentación	42
4.4.2	Capa de aplicación.....	47
4.4.3	Capa de persistencia	70
5	Conclusiones	71
5.1	Posibles ampliaciones	71
6	Bibliografía	72
7	Anexos.....	73
7.1	Anexo I: Casos de uso	73
7.2	Anexo II: Diseño conceptual de la base de datos.....	80
7.3	Anexo III: Dominios base de datos.....	83
7.4	Anexo IV: Fichero de creación de la base de datos.....	85
7.5	Anexo V: Manual de usuario.....	90
7.5.1	Registro.....	90
7.5.2	Olvido de contraseña	91
7.5.3	Iniciar sesión	92
7.5.4	Editar perfil	93
7.5.5	Página pública de usuario.....	93
7.5.6	Búsqueda de profesionales	94
7.5.7	Crear proyecto nuevo.....	95
7.5.8	Buscar proyectos.....	97
7.5.9	Página pública de proyecto.....	98
7.5.10	Proyectos del usuario	100
7.5.11	Intranet de proyecto	101
7.5.12	Intranet de tarea	102
7.5.13	Valorar a un usuario	104
7.6	Anexo VI: Índice de ilustraciones	105



1 Introducción

Este documento es la memoria del Trabajo Final de Grado en Ingeniería Informática, cursado en la Escuela Técnica Superior de Ingeniería informática de la Universidad Politécnica de Valencia.

Durante el resto del documento podremos ver la evolución que ha sufrido el proyecto para su desarrollo, desde el planteamiento y análisis de requisitos, hasta su implementación y diseño de la interfaz visible para el usuario final.

1.1 Contexto

En un momento en el que encontrar un empleo o mantener una empresa es una tarea complicada, se buscan nuevos métodos para realizar trabajos o buscar colaboradores donde el gasto sea mínimo.

Esta web pretende ser un punto de reunión para todas esas personas, tanto profesionales como clientes, que tienen ideas para proyectos o quieren realizar trabajos puntuales, sin estar obligados a tener una empresa o contratar personal, de modo que sea sencillo colaborar entre ellas aportando cada una los conocimientos o recursos necesarios para las tareas a llevar a cabo.

1.2 Objetivos del proyecto

La finalidad del proyecto es crear un portal web que sirva de punto de encuentro para clientes y profesionales, de modo que puedan publicar proyectos y colaborar en ellos de un modo sencillo.

El portal tratará de cubrir los siguientes objetivos esenciales:

- Permitir crear cuentas nuevas a los usuarios que deseen participar en él.
- Albergar y mostrar a los visitantes tanto los proyectos creados como los perfiles de los profesionales dados de alta en el sistema.
- Acceder a una intranet donde modificar datos personales, crear proyectos, inscribirse y/o gestionar tareas o colaborar en ellas.
- Estar disponible en internet de modo que sea accesible desde cualquier dispositivo.

1.3 Estructura del documento

Este documento presenta una estructura en etapas, común en el desarrollo de un proyecto software. Estas etapas son las siguientes:

- Especificación de requisitos: En esta fase se reúnen todos los requisitos funcionales que debe tener el producto final.
- Análisis: El Análisis describe la estructura y funcionalidad del producto mediante diagramas que permitan comprender el sistema. Se incluyen diagramas de clases UML y casos de uso, que describen los principales comportamientos del portal.
- Diseño e implementación: En esta fase se describen los diferentes niveles que componen la arquitectura de la aplicación, así como las diferentes tecnologías

utilizadas. Se describen detalladamente la implementación y el código asociado.

- Conclusiones: En este apartado se señalan las conclusiones finales obtenidas y los objetivos cumplidos.
- Bibliografía: Por último se detalla la bibliografía consultada para el desarrollo del proyecto y/o la memoria.
- Anexos: En los anexos se incluyen los casos de uso, el diseño conceptual de la base de datos, manual de usuario y cualquier documento que por su extensión se ha decidido incluir al final de la memoria.

2 Especificación de requisitos

2.1 Introducción

2.1.1 Propósito

La especificación de requisitos tiene como finalidad conocer lo que se demanda y espera obtener de la aplicación, muy importante para el desarrollo de la misma.

2.1.2 Ámbito

Los dueños de la web han decidido crear un portal donde puedan congregarse clientes, empresas y autónomos para crear o colaborar en proyectos puntuales.

2.1.3 Definiciones, acrónimos y abreviaturas

- Proyecto: Trabajo o conjunto de trabajos a realizar, propuesto por un usuario, en el que pueden colaborar profesionales registrados que se hayan inscrito. Se compone de una o varias tareas que se deberán llevar a cabo para su finalización.
- Tarea: Cada uno de los procesos que hay que realizar para terminar un proyecto.
- Cliente: El cliente es el perfil más básico de la web, ya que solo puede proponer proyectos, nunca colaborar.
- Profesional: Los profesionales pueden proponer proyectos o colaborar en otras tareas. Disponen de un perfil más elaborado con currículum/descripción. Pueden ser empresa o *freelance*.
- *Freelance*: Un *freelance* es aquel profesional que trabaja por cuenta propia (autónomo).
- Anónimo: Toda persona que accede a la web sin identificarse.
- Valoración: Nota que se da o se obtiene al terminar, abandonar o cambiar al colaborador de una tarea.
- Recursos: Aquellos archivos vinculados a una tarea.
- Candidato: Cualquier usuario de tipo profesional que se inscriba en una tarea con la intención de colaborar en ella.
- Colaborador: Aquel usuario, candidato de una tarea, al que le sea asignada la misma.
- Propietario: Persona que propone el proyecto. Puede ser cualquier usuario registrado en la aplicación

2.1.4 Visión global

El producto a desarrollar es un portal web para gestionar las tareas y los usuarios colaboradores en proyectos.

Los visitantes de la web podrán listar tanto los proyectos con búsqueda de colaboradores activa, como los profesionales dados de alta.

Para acceder a la intranet, será requisito indispensable haberse dado de alta previamente.

2.2 Descripción general

2.2.1 Perspectiva del producto

La aplicación que desarrollamos es totalmente independiente.

El sitio web podrá ser visitado por cualquier usuario, independiente de la plataforma y del navegador que utilice.

2.2.2 Características de usuario

Podemos distinguir entre varios tipos de usuario que pueden acceder a la web:

- Anónimo: Cualquier persona no identificada que visita la web.
- Usuario registrado: Aquella persona que accede a la intranet mediante un usuario y una clave.
 - Cliente: Aquel usuario que no tiene intención de colaborar en otros proyectos, pero si proponerlos para su realización.
 - Profesional: Aquel usuario que pretende tanto colaborar en proyectos como proponerlos.
 - Empresa: Perfil profesional que pertenece a una empresa o sociedad.
 - *Freelance*: Perfil profesional que pertenece a un trabajador por cuenta propia o autónomo.
- Administrador: Persona encargada del mantenimiento de la web.

2.2.3 Operaciones permitidas

En este apartado se enumerarán las operaciones permitidas para cada uno de los usuarios.

- Anónimo
 - Listar profesionales
 - Listar proyectos
 - Registrarse
- Cliente
 - Acceder a la zona de usuario mediante nombre de usuario y contraseña.
 - Listar profesionales.
 - Listar proyectos.
 - Modificar sus datos.
 - Listar sus proyectos.
 - Crear nuevo proyecto.
 - Acceder a la intranet de un proyecto como propietario.
 - Acceder a la intranet de las tareas como propietario.
 - Gestionar candidatos y colaboradores en tareas como propietario.
 - Subir y descargar archivos de las tareas como propietario.
 - Valorar a otro usuario colaborador.
- Profesional
 - Acceder a la zona de usuario mediante nombre de usuario y contraseña.
 - Listar profesionales.
 - Listar proyectos.
 - Modificar sus datos.
 - Listar sus proyectos.

- Crear nuevo proyecto.
- Acceder a la intranet de un proyecto como propietario o colaborador.
- Acceder a la intranet de las tareas como propietario o colaborador.
- Gestionar candidatos y colaboradores en tareas como propietario.
- Subir y descargar archivos de las tareas como propietario o colaborador.
- Inscribirse en una tarea como candidato.
- Valorar a otro usuario como colaborador o propietario.
- Administrador
 - Listar/consultar todos los datos de la base de datos.
 - Añadir usuarios/proyectos/tareas.
 - Modificar usuarios/proyectos/tareas.
 - Borrar usuarios/proyectos/tareas.
 - Acceder al escritorio de Wordpress.

2.2.4 Restricciones generales

Para acceder a la web no se requiere ningún hardware específico, solamente acceso a Internet y un navegador con JavaScript habilitado, ya que el servidor es el que se encarga de gestionar las transacciones sobre la base de datos y de interpretar el código PHP de las páginas.

Solo el administrador, al identificarse, tiene posibilidad de modificar cualquier dato de la web o la base de datos. Los demás usuarios registrados solo podrán modificar sus propios datos personales.

2.2.5 Supuestos y dependencias

Deberá haber un correcto funcionamiento de la base de datos y se deberán realizar copias de seguridad periódicamente.

Igualmente se actualizará Wordpress siempre que haya nuevas versiones para asegurar la seguridad y fiabilidad del sistema.

2.3 Requisitos específicos

2.3.1 Requisitos de interfaces externas

2.3.1.1 Interfaces de usuario

La aplicación se visualizará en la pantalla del usuario, de forma sencilla e intuitiva.

2.3.1.2 Interfaces hardware

En nuestra aplicación no se necesita ningún interfaz hardware específico, puede ser visitada desde cualquier dispositivo que pueda conectarse a internet y navegar.

2.3.1.3 Interfaces software

La aplicación estará desarrollada en sobre Wordpress, utilizando PHP5 y jQuery como tecnologías principales. Será creada en Windows 7, con Dreamweaver CS6 sobre un servidor Apache con una base de datos MySQL.



El portal será *responsive*¹ o adaptativo de modo que pueda funcionar sobre cualquier navegador y cualquier sistema operativo.

2.3.2 Requisitos funcionales

2.3.2.1 Almacenamiento y mantenimiento de los datos

1. Operaciones permitidas para usuario anónimo:
 - 1.1. Listar profesionales
 - Introducción:** Visualizar por pantalla una lista con los profesionales registrados.
 - Entradas:** Filtro de búsqueda.
 - Proceso:** Consultar datos que coincidan con la búsqueda.
 - Salida:** Visualización de los resultados de la búsqueda como una lista.
 - 1.2. Listar proyectos
 - Introducción:** Visualizar por pantalla una lista con los proyectos creados que tengan en ese momento proceso activo de búsqueda de colaboradores.
 - Entradas:** Filtro de búsqueda.
 - Proceso:** Consultar datos que coincidan con la búsqueda.
 - Salida:** Visualización de los resultados de la búsqueda como una lista.
 - 1.3. Registrarse
 - Introducción:** Registrar un nuevo usuario en la web.
 - Entradas:** Datos de registro.
 - Proceso:** Rellenar formulario de registro e insertar los nuevos datos en la base de datos.
 - Salida:** Resultado del registro.
2. Operaciones permitidas para usuario cliente (incluye las operaciones permitidas al usuario anónimo):
 - 2.1. Acceder a la zona de usuario mediante nombre de usuario y contraseña.
 - Introducción:** Acceder a la zona privada mediante nombre de usuario y contraseña.
 - Entradas:** Nombre de usuario y contraseña.
 - Proceso:** Rellenar formulario de inicio de sesión y consultar si los datos son correctos.
 - Salida:** Resultado del inicio de sesión.
 - 2.2. Modificar sus datos.
 - Introducción:** Modificar los datos personales del usuario.
 - Entradas:** Datos del usuario, nuevos datos.
 - Proceso:** Modificar en la base de datos aquellos datos que hayan sido modificados.
 - Salida:** Visualización de los datos modificados.
 - 2.3. Listar sus proyectos.
 - Introducción:** Visualizar por pantalla una lista con los profesionales registrados.
 - Entradas:** Filtro de búsqueda.
 - Proceso:** Consultar datos que coincidan con la búsqueda.
 - Salida:** Visualización los resultados de la búsqueda como una lista.
 - 2.4. Crear nuevo proyecto.
 - Introducción:** Crear un proyecto nuevo en la web.
 - Entradas:** Datos del proyecto y las tareas.
 - Proceso:** Rellenar formulario de creación de nuevo proyecto, insertar nuevo proyecto y sus tareas en la base de datos.
 - Salida:** Resultado de la creación del proyecto.
 - 2.5. Acceder a la intranet de un proyecto como propietario.
 - Introducción:** Acceder a la intranet de un proyecto del que se es propietario.

¹ El diseño web *responsive* o diseño web adaptativo es una técnica de diseño web que prioriza la correcta visualización de una misma página en distintos dispositivos: Desde ordenadores de escritorio a tablets o móviles. (4odeFiebre)

- Entradas:** Identificador del proyecto, identificador del usuario.
Proceso: Buscar el proyecto y comprobar que el usuario es el propietario.
Salida: Intranet del proyecto.
- 2.6. Acceder a la intranet de las tareas como propietario.
Introducción: Acceder a la intranet de una tarea de la que se es propietario.
Entradas: Identificador del proyecto, identificador de la tarea, identificador del usuario.
Proceso: Buscar el proyecto y comprobar que el usuario es el propietario.
Salida: Intranet de la tarea.
- 2.7. Gestionar candidatos y colaboradores en tareas como propietario.
Introducción: Cambiar el estado de la candidatura de los candidatos inscritos en la tarea.
Entradas: Identificador de la tarea, identificador del candidato.
Proceso: Modificar en la base de datos el estado de la candidatura de un candidato inscrito.
Salida: Resultado del cambio de estado.
- 2.8. Subir y descargar archivos de las tareas como propietario o colaborador.
Introducción: Añadir o descargar recursos de la tarea.
Entradas: Archivo.
Proceso: Subir al servidor el archivo nuevo o descargar de él un archivo ya existente.
Salida: Lista de archivos de la tarea.
- 2.9. Valorar a otro usuario como colaborador.
Introducción: Poner nota a un usuario.
Entradas: Identificador del usuario, identificador de la tarea.
Proceso: Añadir una nota al usuario en la base de datos.
Salida: Resultado de la valoración.
3. Operaciones permitidas para usuario profesional (incluye las operaciones permitidas al usuario cliente):
- 3.1. Inscribirse en una tarea como candidato.
Introducción: Inscribirse en una tarea con búsqueda de candidatos abierta.
Entradas: Identificador de la tarea, identificador del usuario.
Proceso: Añadir al usuario a la lista de candidatos de la tarea.
Salida: Resultado de la inscripción.
- 3.2. Valorar a otro usuario como colaborador o propietario.
Introducción: Poner nota a un usuario.
Entradas: Identificador del usuario, identificador de la tarea.
Proceso: Añadir una nota al usuario en la base de datos.
Salida: Resultado de la valoración.
4. Operaciones permitidas para usuario administrador:
- 4.1. Listar/consultar todos los datos de la base de datos.
Introducción: Visualizar por pantalla los datos.
Entradas: Filtro de búsqueda/consulta a la base de datos.
Proceso: Consultar datos que coincidan con la búsqueda.
Salida: Visualización de los resultados.
- 4.2. Añadir usuarios/proyectos/tareas.
Introducción: Añadir datos a la base de datos.
Entradas: Nuevos datos.
Proceso: Insertar datos nuevos en la base de datos.
Salida: Visualización de los resultados.
- 4.3. Modificar usuarios/proyectos/tareas.
Introducción: Modificar datos de la base de datos.
Entradas: Datos, nuevos datos.
Proceso: Modificar datos ya existentes en la base de datos.
Salida: Visualización de los resultados.
- 4.4. Borrar usuarios/proyectos/tareas.
Introducción: Borrar datos de la base de datos.
Entradas: Datos a borrar.

Proceso: Eliminar de la base de datos los datos seleccionados.

Salida: Visualización de los resultados.

4.5. Acceder al escritorio de Wordpress.

Introducción: Acceder mediante usuario y contraseña de administrador al escritorio de Wordpress.

Entradas: Nombre de usuario y contraseña.

Proceso: Iniciar sesión como administrador.

Salida: Escritorio de Wordpress.

2.3.3 Restricciones de diseño

Se realizará un diseño general para cualquier situación que se adaptará según la interfaz del usuario (*responsive*).

2.3.4 Atributos

2.3.4.1 Seguridad

La aplicación no mostrará de manera pública los datos personales de sus usuarios, tales como documento de identidad, para protegerlos de posibles fraudes con sus datos personales.

El usuario que desee utilizar los servicios de la Web tendrá identificarse dentro de la página con su nombre de usuario y clave.

El servidor protegerá todas las transacciones que se puedan realizar tanto del usuario hacia el servidor como el servidor al usuario.

2.3.4.2 Mantenimiento

El mantenimiento se llevará a cabo por la persona asignada y autorizada previamente, con conocimientos sobre el sistema.

Se realizarán las pertinentes copias de seguridad de la base de datos periódicamente.

3 Análisis

El siguiente apartado describe la fase de análisis del proyecto. En esta etapa se analiza la aplicación que vamos a crear y se describe su estructura y funcionalidad mediante diagramas UML, con los que entender su funcionalidad nos será más sencillo.

Al final de esta fase tendremos un modelo con los diferentes actores que interactúan con los objetos del sistema a través de diversas relaciones y las acciones que cada uno de ellos podrá llevar a cabo.

3.1 UML

UML es un lenguaje de modelado que utilizaremos para describir nuestro sistema, para detallar y construir los métodos y atributos que estarán presentes en él. Realizaremos varios diagramas para poder entender mejor nuestra aplicación

3.1.1 Diagrama de clases

Un diagrama de clases describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.

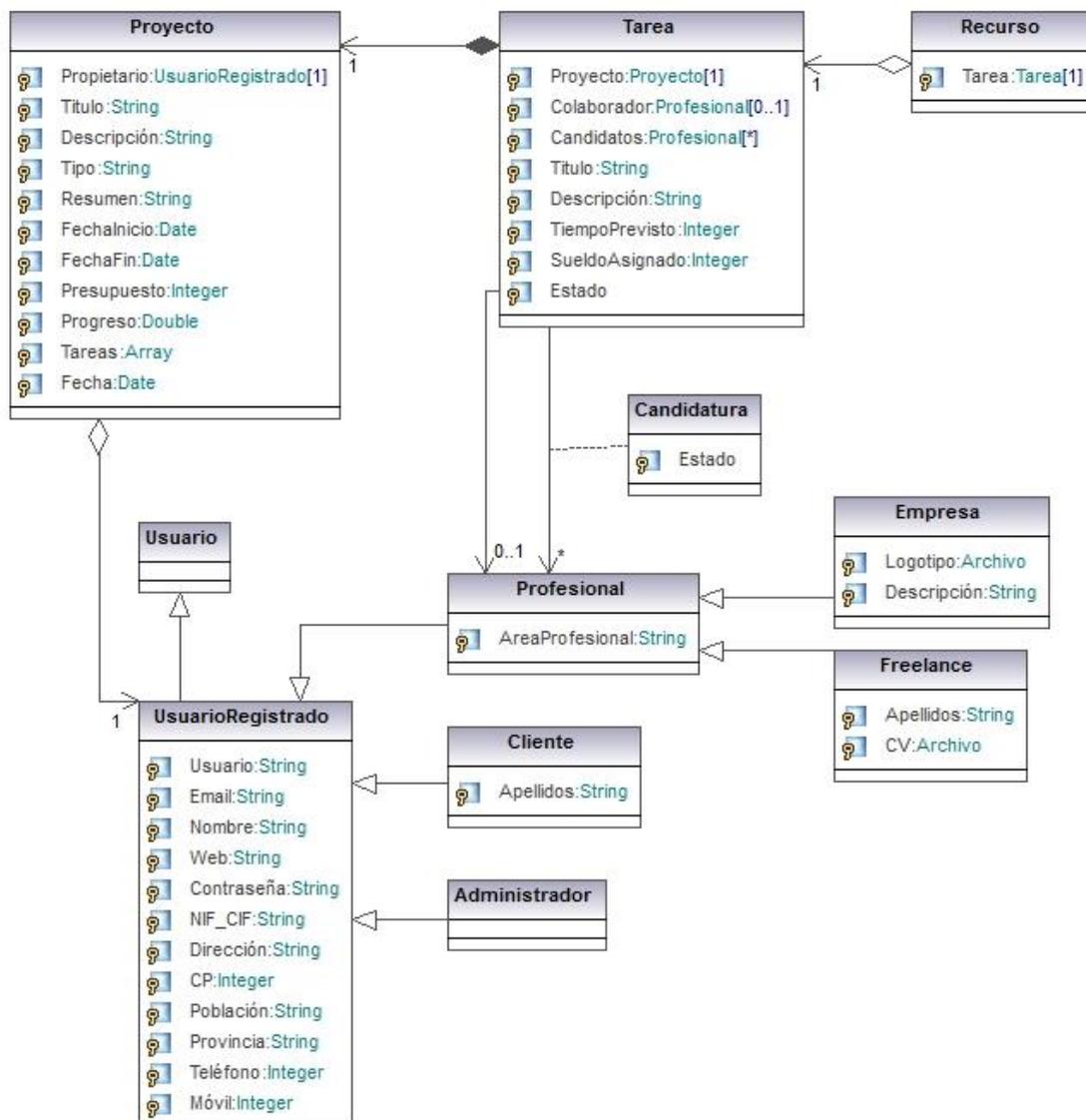


Ilustración 3-1 – Diagrama de clases



Un usuario se considerará anónimo mientras no se identifique en la aplicación. Cuando lo haga pasará a ser usuario registrado.

El administrador es aquel usuario registrado con capacidad y permisos totales para modificar cualquier elemento de la aplicación.

De los usuarios registrados conocemos sus datos personales, así como su nombre de usuario y contraseña.

Un usuario registrado podrá ser administrador, cliente o profesional. El usuario profesional incluirá, además de los datos como usuario, el área profesional.

Un profesional será empresa o *freelance*. De estos tipos tendremos logotipo y descripción en caso de ser empresa, o apellidos y currículum vitae en caso del *freelance*.

Los usuarios registrados podrán ser propietarios de proyectos. De estos conoceremos su título, fecha, presupuesto, tipo y todos aquellos datos que el usuario propietario haya completado en su creación.

Cada proyecto tendrá una o varias tareas de las que sabremos su título, descripción, sueldo asignado y tiempo previsto.

Un proyecto tiene las siguientes características:

- Propietario: Persona que propone el proyecto. Puede ser cualquier usuario registrado en la aplicación. Gestiona el proyecto y elige a los colaboradores asignados a las tareas.
- Título: Título del proyecto
- Resumen: Descripción breve del proyecto (Sale en la búsqueda de proyectos).
- Descripción: Descripción del proyecto.
- Fecha de inicio prevista: Cuándo se debería comenzar el proyecto.
- Fecha de fin prevista: Cuándo se prevé que debería estar terminado el proyecto.
- Presupuesto: Capital máximo del que se dispone para ejecutar el proyecto.
- Tipo de proyecto: Tipo de trabajo que se realizará en el proyecto (por ejemplo, página web).
- Progreso: Porcentaje del trabajo ya realizado. Depende del número de tareas finalizadas.
- Tareas: Procesos que hay que realizar para terminar el proyecto.
 - Título: Título de la tarea.
 - Descripción: Describe lo que hay que realizar.
 - Plazo previsto para la tarea: Tiempo en el que se debería poder terminar la tarea en horas.
 - Clase de tarea: Tipo de tarea que se va a realizar (por ejemplo, formulario de contacto).
 - Sueldo asignado: Parte del presupuesto que se asignará al colaborador de esta tarea.
 - Colaborador: A cada tarea se le asignará un colaborador que será el que la lleve a cabo.
 - Recursos: Archivos o documentos que se adjunten en la tarea que podrían servir para realizarla.

- Estado: El estado lo puede cambiar el propietario y puede ser:
 - Buscando colaboradores: Cuando tenga un proceso de búsqueda abierto.
 - Seleccionando colaboradores: Cuando el proceso de búsqueda esté cerrado y aún no se haya elegido un colaborador.
 - En proceso: Cuando el colaborador asignado esté trabajando en la tarea.
 - Abandonada: Si el propietario decide cambiarla de estado. Este estado no se puede modificar.
 - Pausada: Si el colaborador asignado es eliminado de la tarea, o si el propietario decide cambiarla de estado.
 - Finalizada: Si la tarea ya está realizada. Este estado no se puede modificar.
- Candidatos: Lista de usuarios inscritos a la tarea, cuyo estado puede variar entre:
 - Presentada: La candidatura ha sido presentada a la tarea por parte del usuario.
 - Finalista: El propietario le ha seleccionado como uno de los finalistas para colaborar en la tarea.
 - Asignada: El propietario le ha asignado como el colaborador asociado a la tarea. Este usuario podrá entrar en la intranet de la tarea mientras sea el colaborador de la misma.
 - Descartada: El propietario ha desestimado su candidatura.

3.1.2 Diagramas de casos de uso

Los diagramas de casos de uso describen el comportamiento de la aplicación y enfatizan lo que debe suceder en el sistema modelado.

En estos diagramas podemos observar varios actores, aquellos que podrán actuar con el sistema y se describe de este modo qué podrán hacer.

En nuestro sistema podemos distinguir varios actores:

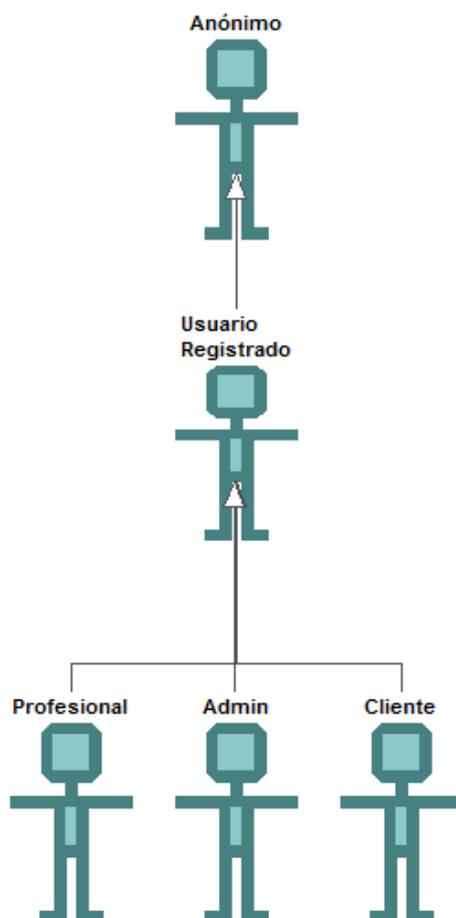


Ilustración 3-2 - Diagrama de casos de uso

Anónimo es toda persona que accede a la aplicación sin identificarse, este actor podrá navegar por la web, pero no acceder a la intranet. Si podrá, en cambio, cualquier usuario registrado.

Este actor puede a su vez ser Administrador, Profesional o Cliente. Según el papel que desempeñe tendrá unos privilegios u otros. El administrador es aquel con acceso total, que puede modificar por completo la aplicación, es el que se encarga de su mantenimiento.

El cliente y profesional tendrán las mismas funciones, la única diferencia será que el cliente no podrá inscribirse en tareas, mientras que el profesional sí.

A continuación se muestran los casos de uso según cada actor.

3.1.2.1 Caso de uso: Anónimo

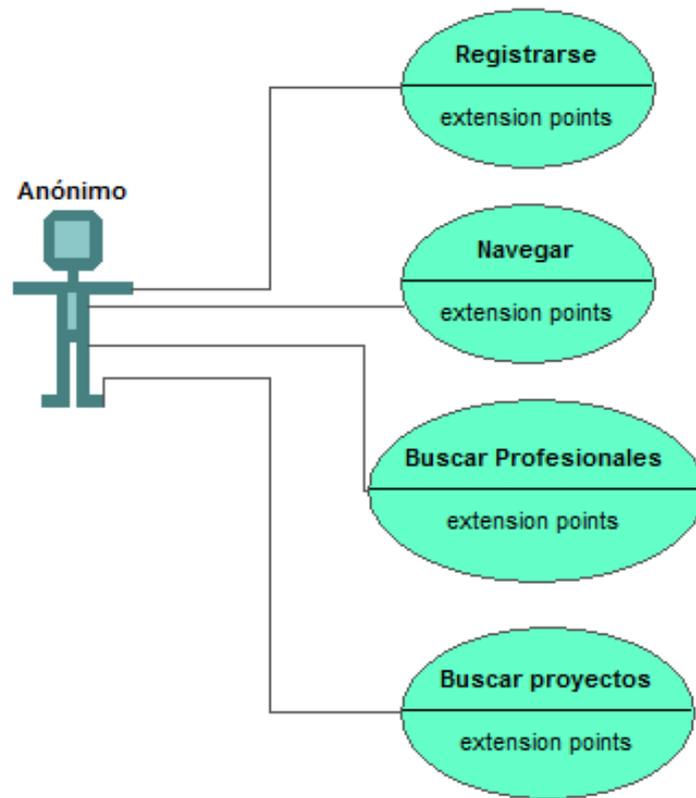


Ilustración 3-3 - Caso de uso: Anónimo

El usuario anónimo solo puede navegar por la web y registrarse como usuario registrado.

3.1.2.2 Caso de uso: Usuario registrado

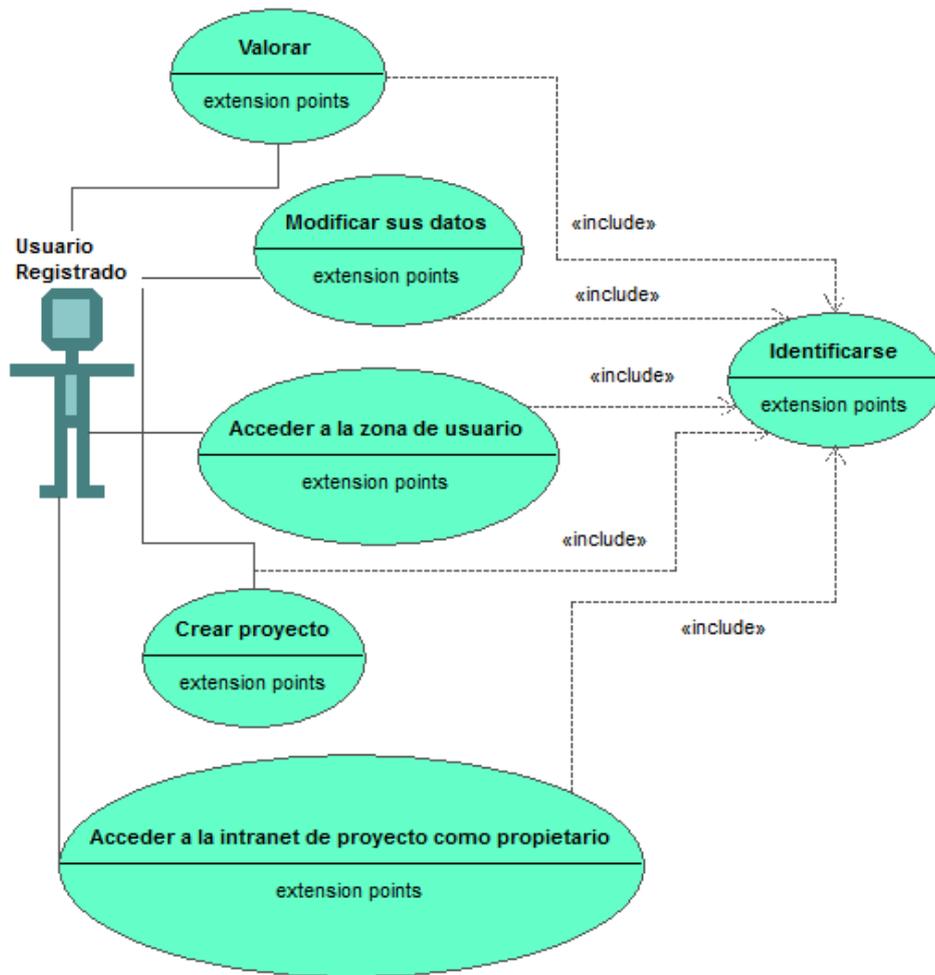


Ilustración 3-4 - Caso de uso: Usuario registrado

El usuario registrado puede identificarse para poder acceder a la zona de usuario y de este modo crear proyectos, participar en ellos, modificar sus datos o valorar usuarios.

3.1.2.3 Caso de uso: Profesional

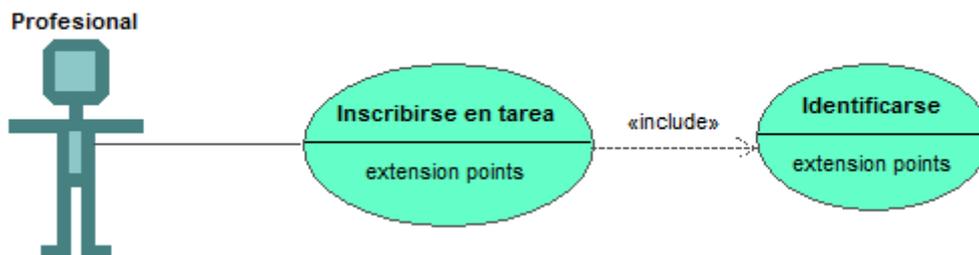


Ilustración 3-5 - Caso de uso: Profesional

El usuario de tipo profesional puede realizar las mismas tareas que el usuario registrado y además puede inscribirse en tareas de otros usuarios para colaborar en ellas.

3.1.2.4 Caso de uso: Administrador

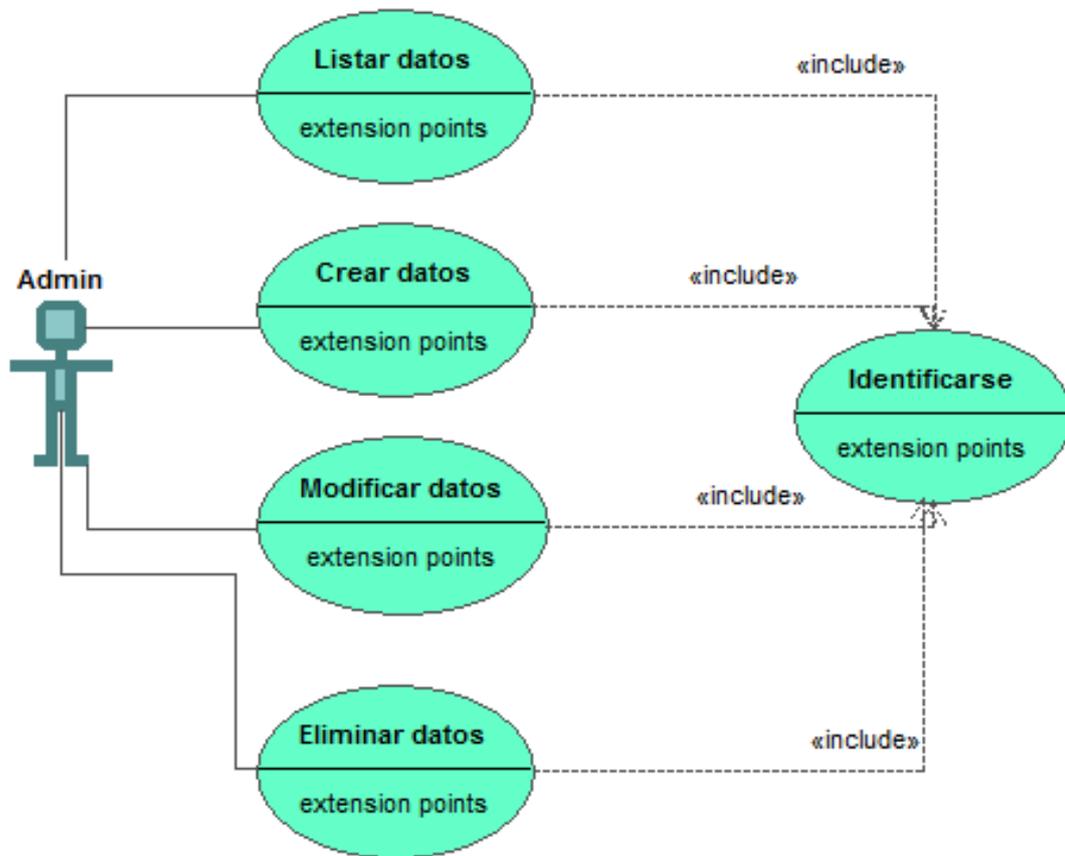


Ilustración 3-6 - Caso de uso: Administrador

El Administrador tiene permisos totales para modificar, añadir o borrar cualquier elemento de la aplicación.

4 Diseño e implementación

4.1 Arquitectura del sistema

Nuestra aplicación está basada en el modelo Cliente/Servidor. Los clientes realizan peticiones al Servidor Web que le ofrece la respuesta, de este modo la capacidad de proceso está repartida entre los clientes y los servidores.

Mediante esta arquitectura los accesos, recursos y la integridad de los datos son controlados por el servidor, de modo que un cliente no puede acceder a aquellos datos a los que no esté autorizado.

La arquitectura de nuestra web estará estructurada de la siguiente manera:

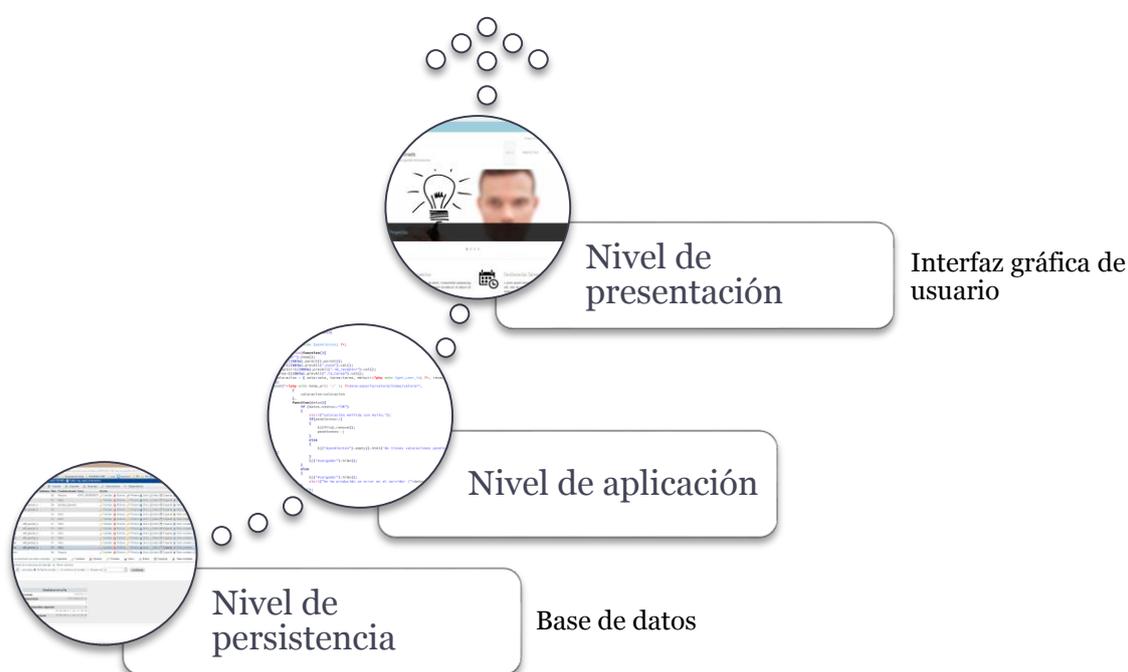


Ilustración 4-1 - Niveles de arquitectura

- **Nivel de presentación:** Esta capa de nuestra arquitectura se encarga de la representación de la información para el usuario final, interactuando con él y comunicándose únicamente con el nivel de aplicación.
- **Nivel de aplicación:** Es donde se ubica el código de los programas, que se ejecutan, recibiendo las peticiones del usuario y enviándole las respuestas tras el proceso.
- **Nivel de persistencia:** En esta capa se encuentran los datos guardados en bases de datos y procesados por el nivel de aplicación.

4.1.1 Nivel de presentación

El nivel de Presentación está formado por todos los documentos que envía el servidor al cliente y que se muestran al usuario final de la aplicación.

En nuestra aplicación hay dos zonas diferentes. La primera para usuarios en general y la segunda, la intranet, solo para usuarios registrados.

4.1.1.1 Interfaz gráfica

4.1.1.1.1 Página principal

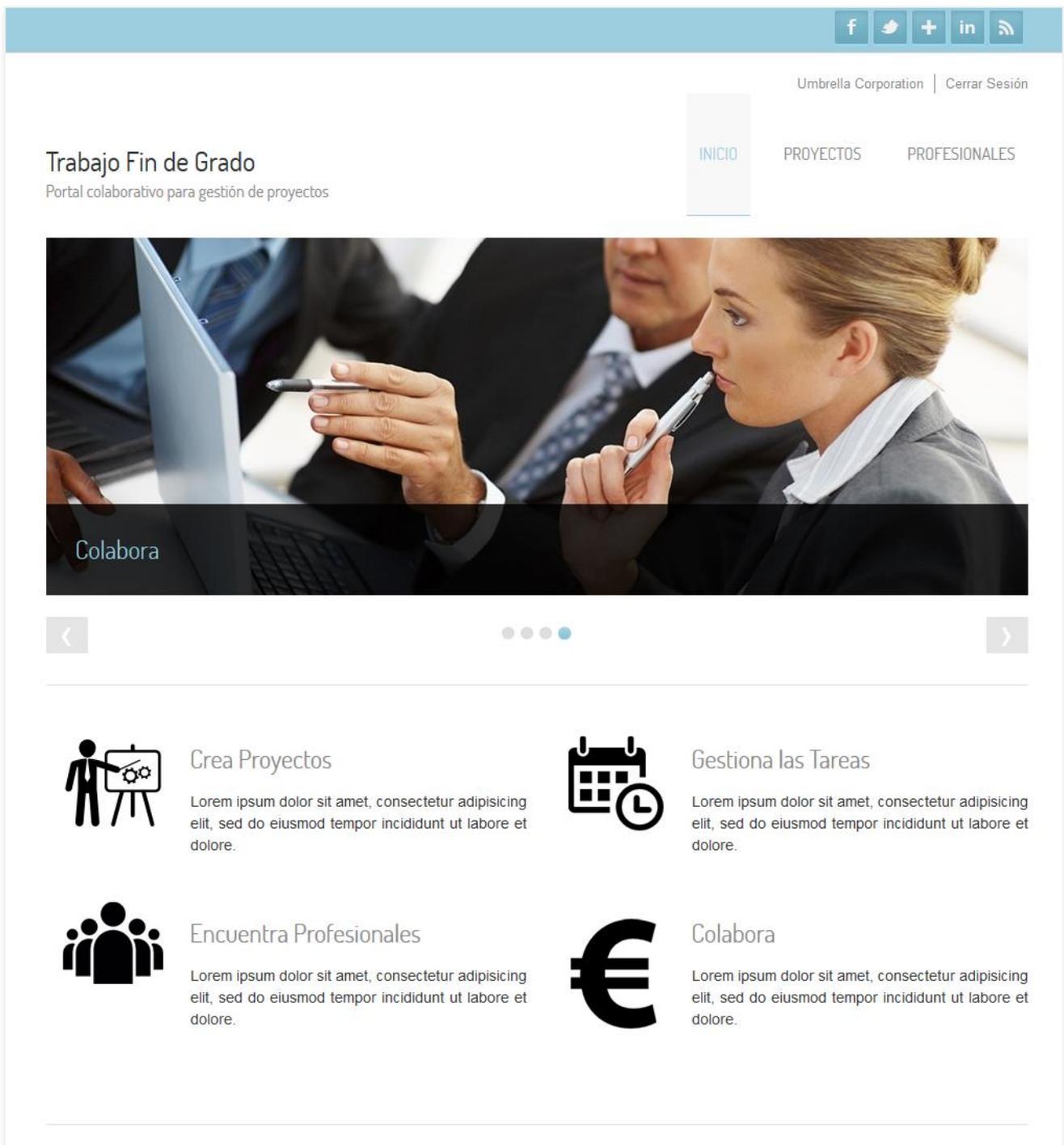


Ilustración 4-2 - Interfaz gráfica: Pantalla principal

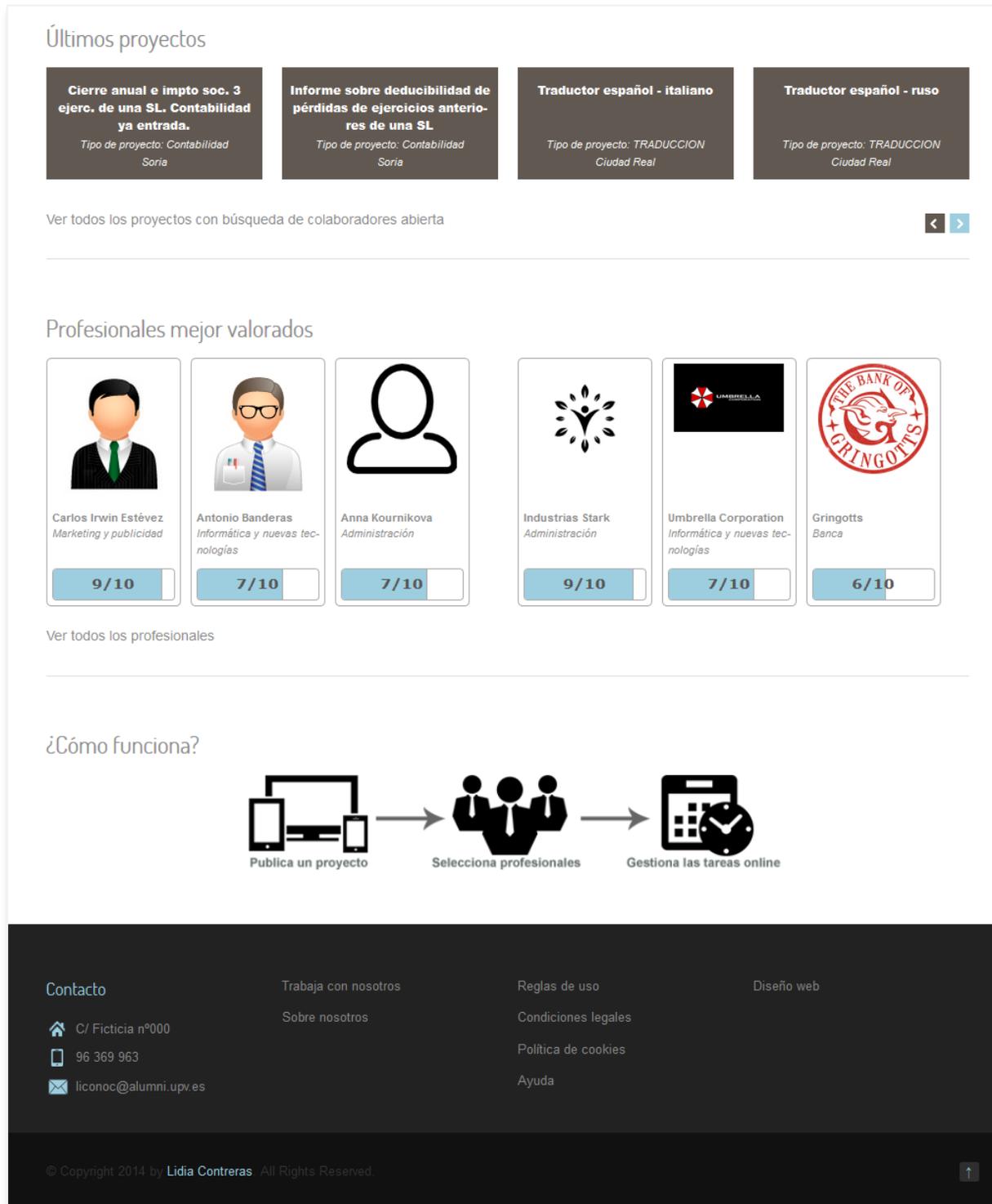


Ilustración 4-3- Interfaz gráfica: Pantalla principal

En la página principal tenemos acceso a todas las zonas de la web, incluidas las búsquedas de proyectos y profesionales.

Existe un menú superior desde el que se accede a la página de inicio de sesión y registro o a la zona de usuario, según el visitante sea anónimo o no.

Se incluye una lista de últimos proyectos creados con búsqueda de candidatos activa, así como los seis profesionales mejor valorados (tres *freelance* y tres empresas).

4.1.1.1.2 Proyectos

Busca proyectos

Nombre

Cierre anual e impto soc. 3 ejerc. de una SL. Contabilidad ya entrada.
Busco un profesional para poner al día una SL.
Tipo: Contabilidad
Soria

Informe sobre deducibilidad de pérdidas de ejercicios anteriores de una SL
Hay que elaborar un informe acerca de la deducibilidad o no de las pérdidas a nivel fiscal.
Tipo: Contabilidad
Soria

Traductor español - italiano
Consultora de formación busca traductor de español a italiano para la traducción de contenidos formativos.
Tipo: TRADUCCION
Ciudad Real

Traductor español - ruso
Consultora de formación busca traductor de español a ruso para la traducción de contenidos formativos.
Tipo: TRADUCCION
Ciudad Real

Oracle DBA
Para un proyecto en Madrid de 7 meses, buscamos un DBA en ORACLE, con al menos 2 ó 3 años de experiencia.
Tipo: Oracle DBA
Cuenca, Cuenca

.NET Complemento IE.
Se desea desarrollar un Browser Helper Object (BHO)
Tipo: Programación
Cuenca, Cuenca

Diseño Imagen / Web
Queremos mejorar nuestra visibilidad y mejorar nuestra imagen al exterior.
Tipo: Diseño gráfico
Zaragoza

EXPERTO EN GMAIL PARA REDACCIÓN DE CONTENIDOS
Estamos seleccionando, un experto en GMAIL que sea capaz de desarrollar el contenido relativo a GMAIL, sus funcionalidades y herramientas, que sirva de base para producir un curso on-line.
Tipo: Profesor
Zaragoza

Creación backend PHP
Creación de un backend PHP para la publicación y suscripción a ofertas de trabajo.
Tipo: Programación
Ávila

Implantación Open ERP, desarrollo módulo a medida y migración
estamos buscando un profesional freelance con experiencia con OpenErp para la instalación y desarrollo de módulos para un proyecto puntual de Open ERP Muy importante tener experiencia en el módulo de
Tipo: Programación
Teruel

Ilustración 4-4- Interfaz gráfica: Proyectos

La página para búsqueda de proyectos incluye un formulario en el que se pueden seleccionar las opciones de búsqueda, esto es: por nombre, tipo de proyecto y localización.

4.1.1.1.3 Profesionales

Busca profesionales

Empresa Todo

Umbrella Corporation
Área: Informática y nuevas tecnologías
Albacete



Central Nuclear de Springfield
Área: Ingeniería
Madrid



Corporación ACME
Área: Comercial y ventas
Valencia



Industrias Stark
Área: Administración
Málaga

Gringotts
Área: Banca
Alicante



Ilustración 4-5- Interfaz gráfica: Profesionales

La búsqueda de profesionales distingue entre empresas y *freelance* y permite filtrar por nombre, población, provincia y área profesional.

4.1.1.1.4 Inicio de sesión



¿Eres Nuevo? Regístrate

Busco colaboradores
Publica proyectos, encuentra profesionales que los lleven a cabo y gestiona las tareas online.

Soy una empresa
Encuentra profesionales que participen en los proyectos de tu empresa o colabora en otros proyectos.

Soy freelance/autónomo
Encuentra proyectos de tu área profesional o publica los tuyos propios para que otros profesionales colaboren contigo.

Usuario

Contraseña

Recuérdame

Entrar →

[¿No recuerdas tu contraseña?](#)

Ilustración 4-6 - Interfaz gráfica: Inicio de sesión

La página de inicio de sesión incluye los campos de usuario y contraseña para acceder a la intranet, un enlace para el posible olvido de la contraseña y los accesos a los formularios de registro de nuevos usuarios.

4.1.1.1.5 Registro de usuario

Regístrate en este sitio

Nombre de usuario

Correo electrónico

Nombre

Apellidos

Web

Contraseña

Password confirmation

Seguridad de la contraseña

Un consejo: La contraseña debe tener, al menos, siete caracteres de longitud. Para hacerla más fuerte, utiliza mayúsculas y minúsculas, números y símbolos como ! " ? \$ % ^ &).

Datos Personales

NIF/CIF

Dirección

Código Postal

Población

Provincia

Teléfono Fijo

Teléfono Móvil



Código

Registrarse

- Acceder
- Contraseña perdida

Ilustración 4-7- Interfaz gráfica: Registro de usuario

El formulario de registro distingue entre cliente, empresa y *freelance*. De este modo según el tipo de usuario los campos obligatorios para cada uno de ellos cambian. Por ejemplo, el usuario de tipo empresa estará obligado a rellenar su descripción, mientras que un *freelance* deberá subir un currículum.

4.1.1.1.6 Página pública de proyecto

Diseño Imagen / Web

Usuario: Nicolas Cage

 Fecha de publicación: 23/07/2014

 Tipo de proyecto: Diseño gráfico

 Localización: Zaragoza

 Presupuesto: 1500 €

 Porcentaje completado: 0 %

Resumen

Queremos mejorar nuestra visibilidad y mejorar nuestra imagen al exterior.

Descripción detallada

Para ello, hemos pensado crear una imagen corporativa con que que nos sintamos cómodos, que sea limpia, que no robe protagonismo a nuestros proyectos (pese a ser un trabajo estupendo), elegante, técnica, minimalista, coherente, versatil,... Existe la posibilidad de diseñar también una página web. Básica, a modo de portfolio y alguna sección adicional como about, contact, references, etc...

Tareas

- ▾ Diseño de logo
 - Descripción
 - Diseño de logo
 -  Número de candidatos apuntados: 1
- Diseño de hoja de presupuesto y factura
- Diseño de tarjeta
- Diseño de carta

Ilustración 4-8- Interfaz gráfica: Página pública de proyecto

La página pública de un proyecto incluye todos los datos que se han completado del proyecto al registrarlo, incluidas las tareas.

Si un usuario profesional ha iniciado sesión se le permite inscribirse a las tareas a las que no esté inscrito ya.

4.1.1.1.7 Página pública de profesional

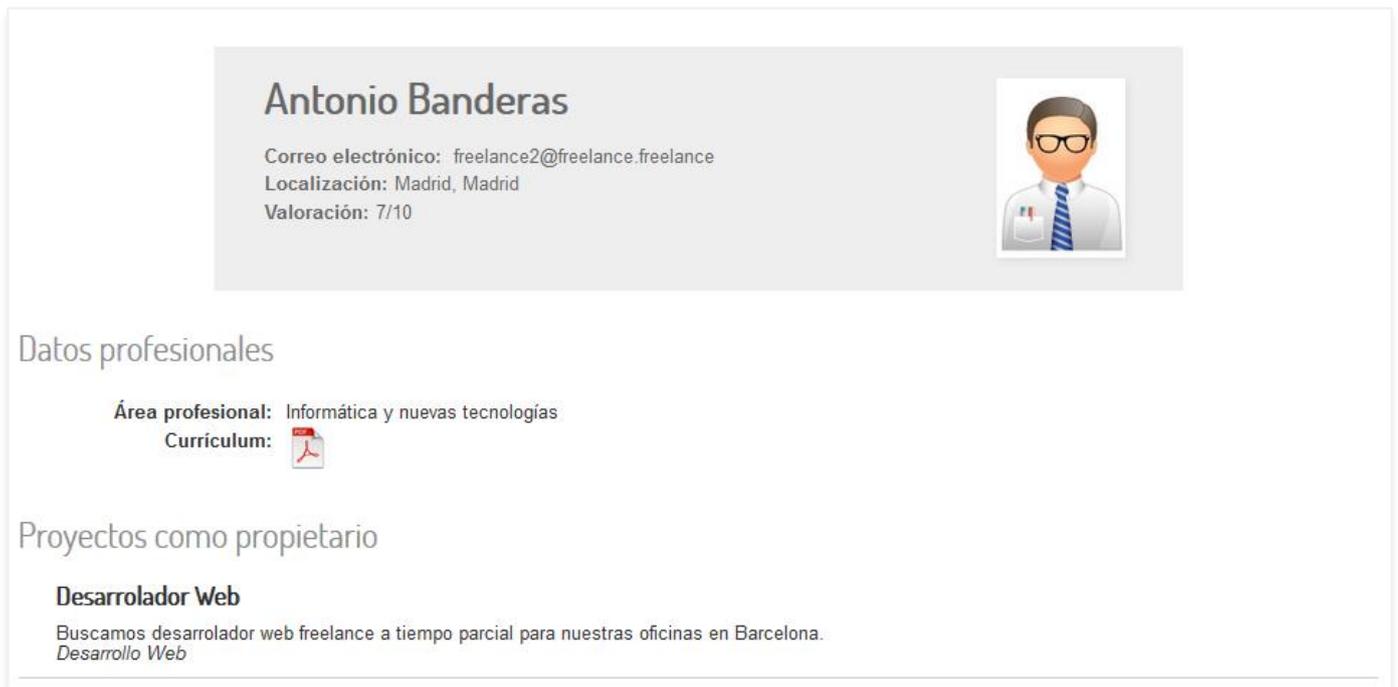


Ilustración 4-9- Interfaz gráfica: Página pública de profesional

En el perfil de usuario se incluyen los datos públicos del usuario (correo electrónico, localización, web y valoración), así como sus datos profesionales.

Si el usuario es colaborador de algún proyecto, este se verá reflejado en una lista en la parte inferior.

4.1.1.1.8 Zona de usuario



Ilustración 4-10- Interfaz gráfica: Zona de usuario

La página principal de la zona de usuario incluye los atajos y enlaces necesarios para el usuario, así como un resumen de su perfil y su valoración.

4.1.1.1.9 Perfil de usuario

Nombre

Nombre de usuario
El nombre de usuario no puede ser modificado.

Nombre

Apellidos

Alias

Información de contacto

Correo electrónico

Web

Contraseña

Nueva contraseña
Si deseas cambiar la contraseña del usuario, escribe aquí dos veces la nueva. En caso contrario, deja las casillas en blanco.

Tecllea tu nueva contraseña otra vez.

Seguridad de la contraseña

Un consejo: La contraseña debe tener, al menos, siete caracteres de longitud. Para hacerla más fuerte, utiliza mayúsculas y minúsculas, números y símbolos como ! * ? \$ % ^ &).

Tus Datos

Datos Personales

NIF/CIF

Dirección

Código Postal

Población

Provincia

Teléfono Fijo

Teléfono Móvil

Ilustración 4-11- Interfaz gráfica: Perfil de usuario

Datos Profesionales

Área profesional

Logotipo



Borrar la imagen

Actualizar la imagen

No se ha seleccionado ningún archivo.

Descripción de la empresa

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi sed consequat quam. Morbi tincidunt venenatis libero, non malesuada felis laoreet id. Proin consequat nec sem non aliquet. Mauris semper condimentum feugiat. In et est bibendum risus aliquam iaculis vel at nulla. Suspendisse felis mi, imperdiet vel adipiscing eget, condimentum sit amet eros. Curabitur eget arcu consequat, mollis elit ut, venenatis tellus.

Ilustración 4-12- Interfaz gráfica: Perfil de usuario

La página de perfil permite editar los datos de cada usuario de la web, con los campos personalizados según del tipo que sea.

4.1.1.1.10 Proyectos de usuario

Proyectos como propietario

Docente Práctico CMS (Drupal, WordPress, Documentum)

selecciona docente práctico de CMS en Drupal, Documentum y Wordpress.
Jefe de proyecto

[Acceder a la intranet del proyecto](#)

[+ Crear nuevo proyecto](#)

Proyectos como candidato

Oracle DBA

Para un proyecto en Madrid de 7 meses, buscamos un DBA en ORACLE, con al menos 2 ó 3 años de experiencia.

Oracle DBA

Tarea: Installation, configuration and upgrading of database server software and related products

Estado: *Presentada*

.NET Complemento IE.

Se desea desarrollar un Browser Helper Object (BHO)

Programación

Tarea: Pruebas de validación

Estado: *Presentada*

EXPERTO EN GMAIL PARA REDACCIÓN DE CONTENIDOS

Estamos seleccionando, un experto en GMAIL que sea capaz de desarrollar el contenido relativo a GMAIL, sus funcionalidades y herramientas, que sirva de base para producir un curso on-line.

Profesor

Tarea: Elaboración del material sobre GMAIL

Estado: *Presentada*

Creación backend PHP

Creación de un backend PHP para la publicación y suscripción a ofertas de trabajo.

Programación

Tarea: Aplicación PHP

Estado: *Presentada*

ANALISTA SAP XI PI

Buscamos un Analista SAP XI PI para proyecto de 6 meses prorrogables con importante empresa del sector.

Analista

Tarea: Pruebas

Estado: *Descartada*

Ilustración 4-13- Interfaz gráfica: Proyectos de usuario

Cada usuario dispone de una lista de los proyectos que ha creado, aquellos a los que se ha inscrito como candidato y/o los que participa como colaborador asignado.

4.1.1.1.11 Valoraciones

Valoraciones

Valoraciones pendientes

Colaborador	Proyecto	Tarea valorada	Puntuación (0-10)
Carlos Irwin Estévez	ANALISTA SAP XI PI	Pruebas	5 <input type="button" value="Valorar"/>

Tus valoraciones

Valoración emitida por	Proyecto	Tarea valorada	Fecha	Puntuación (0-10)
Gabriela Mistral	Docente Práctico CMS (Drupal, Wordpress, Documentum)	Enseñar	23-07-2014	7
Industrias Stark	Docente Práctico CMS (Drupal, Wordpress, Documentum)	Enseñar	23-07-2014	4

Ilustración 4-14- Interfaz gráfica: Valoraciones

La zona de valoraciones permite votar a aquellos usuarios que han sido colaboradores de nuestras tareas o para los que hemos colaborado. También incluye una lista de las valoraciones que se han recibido en proyectos anteriores.

4.1.1.1.12 Nuevo proyecto

Nuevo proyecto

Los campos con * son obligatorios.

Nombre*

Resumen*

Descripción detallada

Tipo de proyecto*

Fecha de inicio prevista

Fecha de fin prevista

Presupuesto previsto €

Tareas

Los campos con * son obligatorios.

Arrástralas para ordenarlas.

▼ Tarea 1

Nombre*

Descripción detallada*

Clase de tarea*

Tiempo previsto horas

Sueldo asignado €

Estado tarea Poner en estado "Buscando colaboradores" Eliminar tarea

Añadir nueva tarea

Ilustración 4-15 - Interfaz gráfica: Nuevo proyecto

El formulario para crear un nuevo proyecto está disponible para cualquier usuario registrado y contiene todos los campos necesarios para publicarlo.

Permite crear tantas tareas como sean necesarias, siempre que haya una como mínimo.

4.1.1.1.13 Intranet proyecto

Docente Práctico CMS (Drupal, Wordpress, Documentum)

Propietario: Umbrella Corporation
Resumen: selecciona docente práctico de CMS en Drupal, Documentum y Wordpress.

[Editar proyecto](#)

Progreso

100%

Tareas

Enseñar
Descripción: Enseñar las herramientas citadas llevando a cabo un proyecto por equipos con las mismas, aprendiendo a la vez a trabajar en equipo y realizar un proyecto, donde el candidato es el jefe del proyecto.
Estado: Finalizada
Colaborador actual: Industrias Stark
[Gestionar tarea](#)

Mensajes

Escribe tu mensaje:

Industrias Stark
23/07/2014 16:13

Fin =)

 **Umbrella Corporation**
22/07/2014 13:43

Comenzamos el **proyecto!**

Ilustración 4-16 - Interfaz gráfica: Intranet de proyecto

En la intranet de un proyecto se puede ver el progreso general del mismo, acceder a las tareas que se tienen asignadas, o a todas si se es el propietario. Incluye una zona de mensajes donde cualquier usuario identificado y con permiso en el proyecto puede escribir.

4.1.1.1.14 Intranet tarea

Creación backend PHP: Aplicación PHP

Propietario: Francis Ford Coppola

Descripción: Realización de una aplicación en PHP consistente en la publicación de ofertas de trabajo para una web hecha en Wordpress (podría hacerse todo externamente sin contar con Wordpress). Los principales puntos de la aplicación son: *Perfiles de empleado (busca trabajo) y administrador (crea las ofertas de trabajo) *El empleado podrá crear su CV mediante un formulario y subirlo en un documento, ambas opciones *El administrador podrá filtrar a los empleados por campos en el backend En un principio la idea es hacer esta aplicación cómo elemento independiente de un sitio web pero sí es posible, cargando el menú de una página hecha en Wordpress a la que pertenece, sí bien esto no es totalmente necesario.

 Editar tarea

Estado: Buscando colaboradores

-  Seleccionar candidato de la lista (Detiene la búsqueda).
-  Pausar tarea
-  Anandonar tarea y eliminar colaborador y candidatos
-  Finalizar tarea y eliminar lista de candidatos

Candidatos	Archivos	
Candidato	Estado candidatura	Acciones
Umbrella Corporation	Presentada	Asignar Hacer finalista Descartar

Mensajes

Escribe tu mensaje:

Enviar

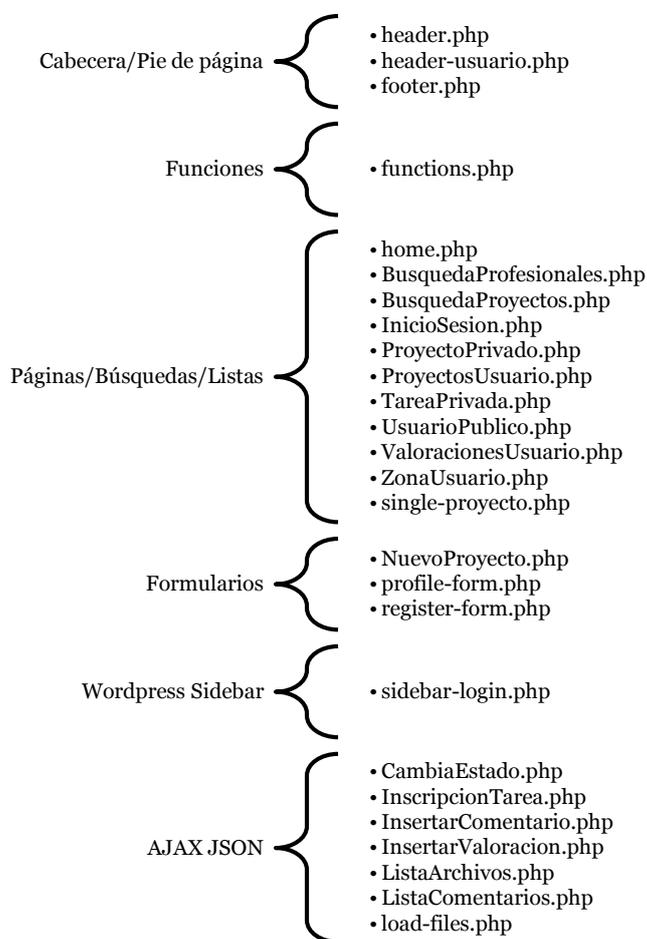
Ilustración 4-17 - Interfaz gráfica: Intranet de tarea

En la intranet de cada tarea el propietario podrá gestionar el estado de las candidaturas de los usuarios inscritos en ella. Tanto él como el candidato asignado podrán subir archivos a la tarea y escribir mensajes.

4.1.2 Nivel de aplicación

Con esta capa definiremos el comportamiento de los objetos que interactuarán en la web. Estos objetos contienen diversos tipos de funciones que se diferencian según su propósito: mostrar información, acceder a la base de datos o procesar datos.

A continuación, se muestran las diferentes acciones que se pueden desarrollar en el sistema, asociando a cada una, los ficheros que contienen las funciones que deben utilizar.



Desde la página principal accederemos a las diversas funciones del portal.



4.1.3 Nivel de persistencia

Es en esta capa donde los datos de nuestra aplicación están almacenados.

4.1.3.1 Base de datos

Para nuestro proyecto hemos utilizado la base de datos proporcionada por Wordpress y hemos añadido las tablas necesarias para completar todas las funciones.

Se muestra a continuación el diagrama entidad-relación que describe las relaciones entre los diferentes usuarios y objetos del sistema y las restricciones entre ellos.

4.1.3.1.1 Diagrama Entidad-Relación

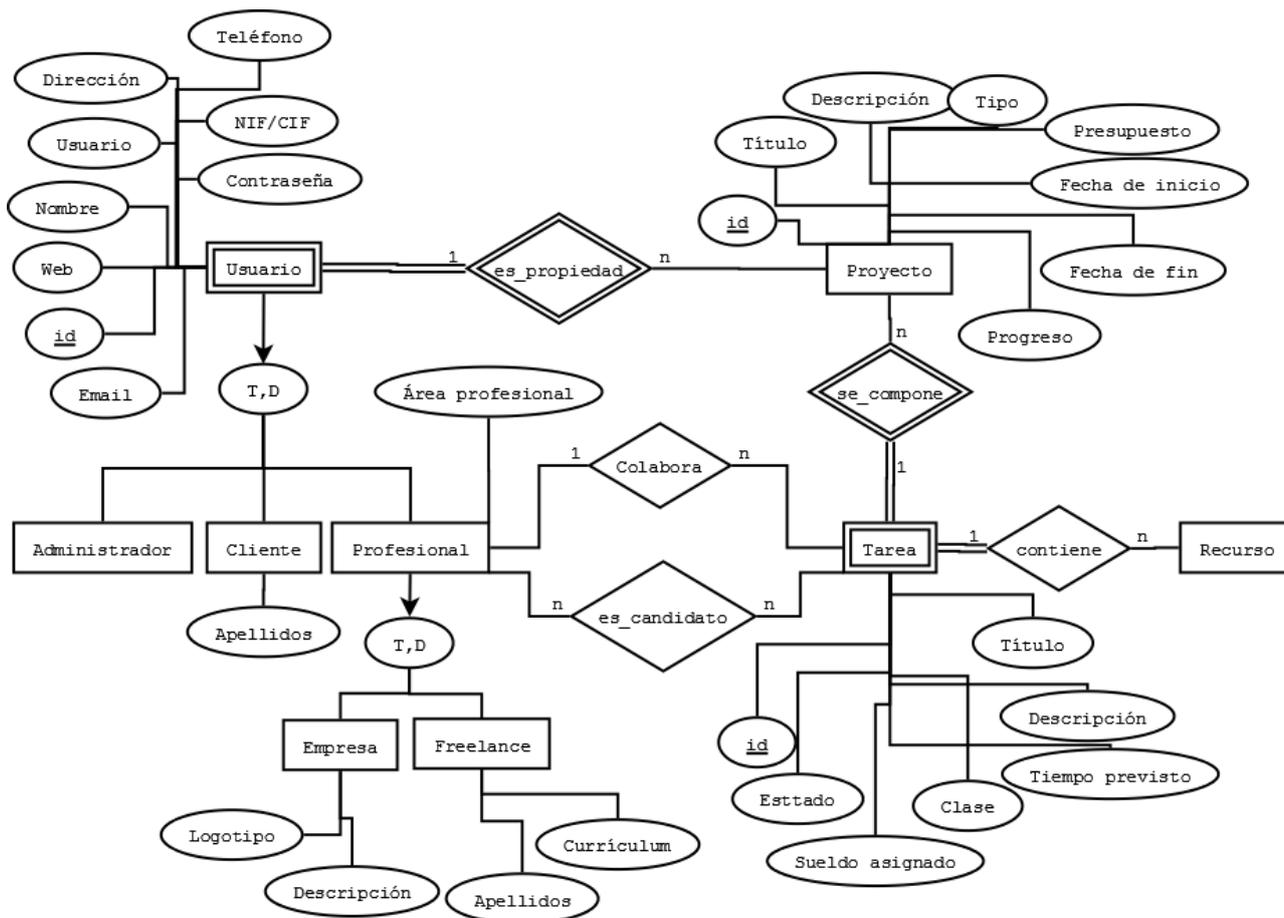


Ilustración 4-18 - Base de datos: diagrama entidad-relación

El diagrama nos detalla los usuarios que interactuarán con el sistema (Empresa, Freelance, Cliente y Administrador) y sus atributos, así como las restricciones a la hora de insertar nuevos datos o modificar los existentes.

4.2 Tecnologías utilizadas

Para realizar este proyecto se han utilizado varias tecnologías y lenguajes, ya sea para el aspecto visual de la web, como del comportamiento y la dinámica.

El portal ha sido creado en Wordpress. De este modo se obtiene una implementación de base para la creación e inicio de sesión de los usuarios y la creación y modificación de diferentes elementos de la web.

Para el aspecto visual de la página se ha utilizado en conjunto HTML y CSS. El primero para la estructura y el segundo para el aspecto. Se ha optado por instalar SimpleCorp, un tema gratuito de Wordpress, que se ha modificado parcialmente para los intereses concretos del portal.

Para el comportamiento de la web se ha utilizado PHP en el lado del servidor, con el que trabaja Wordpress y jQuery (JavaScript), jQuery UI y CSS3 para el lado del cliente. Con PHP nos ocupamos de las acciones de acceso a la base de datos y de envío y recepción de datos desde la web a la base de datos y viceversa, así como de la comprobación de la sesión iniciada y autenticación. Con jQuery, jQuery UI y CSS3 hemos conseguido que la interfaz sea dinámica, cargue datos sin recargar la página y tenga ciertos efectos visuales.

Para la base de datos se ha utilizado MySQL.

A Wordpress se le han instalado algunos plugins que ampliarán su funcionalidad.

4.2.1 HTML

El HTML, *Hyper Text Markup Language* (Lenguaje de marcación de Hipertexto) es el lenguaje de marcas de texto utilizado normalmente en la *www* (*World Wide Web*). Fue creado en 1986 por el físico nuclear Tim Berners-Lee que tomó dos herramientas preexistentes: El concepto de Hipertexto (Conocido también como link o ancla) que permite conectar dos elementos entre si y el SGML (Lenguaje Estándar de Marcación General) que sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML es un sistema de etiquetas que no requiere compilador. (Ravioli, 2007)

4.2.2 CSS

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

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

CSS funciona a base de reglas. Las hojas de estilo están compuestas por una o más de esas reglas aplicadas a un documento HTML o XML. La regla tiene dos partes: un selector y la declaración. A su vez la declaración está compuesta por una propiedad y el valor que se le asigne. ²

4.2.2.1 CSS3

Esta versión de CSS ofrece una gran variedad de opciones muy importantes para las necesidades del diseño web, desde opciones de sombreado y redondeado, hasta funciones avanzadas de movimiento y transformación. (Luca, 2010)

² Hojas de estilo (CSS). Wc3: <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>



4.2.3 PHP

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas.

PHP es un acrónimo recursivo que significa *PHP Hypertext Pre-processor*. Fue creado originalmente por Rasmus Lerdorf en 1994.

Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. (Torres, 2009)

4.2.4 JavaScript

JavaScript es un lenguaje de programación interpretado.

Se utiliza principalmente en su forma del lado del cliente (*client-side*), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del *Document Object Model* (DOM).³

4.2.4.1 jQuery

jQuery es una biblioteca o *framework* de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.⁴

4.2.4.2 jQuery UI

jQuery UI es una biblioteca de componentes para el *framework* jQuery que le añaden un conjunto de *plug-ins*, *widgets* y efectos visuales para la creación de aplicaciones web.⁵

4.2.5 Mysql

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario. (German, 2013)

4.2.6 Wordpress

WordPress es un sistema de gestión de contenido o CMS enfocado a la creación de *blogs* (sitios web periódicamente actualizados). Ha sido desarrollado en PHP para entornos que ejecuten MySQL y Apache, bajo licencia GPL y código modificable, y su fundador es Matt Mullenweg.⁶

4.2.6.1 Tema

Se ha utilizado el tema (o plantilla) *responsive* y gratuito SimpleCorp para Wordpress.⁷

³ JavaScript. Wikipedia: <http://es.wikipedia.org/wiki/JavaScript>

⁴ jQuery. Wikipedia: <http://es.wikipedia.org/wiki/JQuery>

⁵ jQuery UI. Wikipedia: http://es.wikipedia.org/wiki/JQuery_UI

⁶ Wordpress. Wikipedia: <http://es.wikipedia.org/wiki/WordPress>

⁷ Se puede descargar en: <http://www.s5themes.com/theme/simplecorp/>

4.2.6.2 Plugins

Los *plugins* son componentes de código, es decir, programas que se instalan en WordPress para ampliar su funcionalidad estándar.

Están hechos con las mismas tecnologías que WordPress, es decir, están programados en PHP, y usan HTML, hojas de estilo CSS y JavaScript para la parte de interfaz de usuario.

En este trabajo se han instalado los siguientes plugins:

- *Advanced Custom Fields*⁸: Añade nuevos campos a las entradas de Wordpress.
- *Cim� User Extra Fields*⁹: Añade campos al registro y a los perfiles de usuario.
- *Sidebar Login*¹⁰: Permite agregar un widget de inicio de sesión con AJAX.
- *Theme My Login*¹¹: Agrega formularios de registro, olvido de contraseña, etc. acordes con el tema instalado.
- *WPFront User Role Editor*¹²: Permite manejar los roles de usuario.
- *WP Toolbar Removal*¹³: Elimina el aspecto visual y el código de la barra de herramientas superior de Wordpress.

4.2.7 AJAX

AJAX, acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores, dado que está basado en estándares abiertos como JavaScript y *Document Object Model* (DOM).¹⁴

4.2.8 JSON

JSON, acrónimo de *JavaScript Object Notation*, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.¹⁵

⁸ Se puede descargar en: <http://www.advancedcustomfields.com/>

⁹ Se puede descargar en: <http://www.marcocimmino.net/cim�-wordpress-plugins/cim�-user-extra-fields/>

¹⁰ Se puede descargar en: <http://wordpress.org/plugins/sidebar-login/>

¹¹ Se puede descargar en: <http://www.jfarthing.com/development/theme-my-login/>

¹² Se puede descargar en: <http://wpfront.com/wordpress-plugins/user-role-editor-plugin/>

¹³ Se puede descargar en: <http://slangji.wordpress.com/plugins/wp-toolbar-removal/>

¹⁴ AJAX. Wikipedia: <http://es.wikipedia.org/wiki/AJAX>

¹⁵ JSON. Wikipedia: <http://es.wikipedia.org/wiki/JSON>



4.3 Herramientas utilizadas

Para la implementación de la web y la base de datos se han utilizado diferentes aplicaciones o herramientas, que son las siguientes:

- Para la creación de los Scripts de PHP, que incluyen JavaScript y HTML y para las hojas de estilos (CSS) se ha utilizado Adobe *Dreamweaver CS6*.
- La base de datos MySQL, ha sido mantenida y modificada tanto local como remotamente mediante *phpMyAdmin*, una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando Internet.
- Para las pruebas en local se ha utilizado XAMPP.
- Para las pruebas y comprobaciones de la web se han utilizado distintos navegadores (Firefox, Internet Explorer, Chrome, Opera y Safari) sobre los sistemas operativos Windows y Android, así como un complemento para los navegadores llamado *Firebug* para la comprobación de errores en el envío de datos entre los scripts PHP y el desarrollo del CSS de la web.

4.4 Implementación detallada

A continuación se mostrará la descripción detallada de la implementación del proyecto, con ejemplos de código.

4.4.1 Capa de presentación

Para la capa de presentación se ha optado por utilizar HTML para la estructura, CSS para cambiar el aspecto de la página y código jQuery para darle dinamismo.

4.4.1.1 Ocultar Wordpress

Para que los usuarios que accedan a la web puedan trabajar en ella sin ser conscientes de que está implementada sobre Wordpress y no tengan que utilizar su escritorio, se ha optado por ocultarlo. Para esto se han utilizado varios *plugins* disponibles.

Para esconder la barra superior de Wordpress se ha utilizado el plugin ***WP Toolbar Removal***.

Para que el resto de operaciones pudieran hacerse en el *frontpage* de la web se han utilizado dos *plugins*, uno de inicio de sesión (***Sidebar Login***) y otro de registro de usuarios (***Theme My Login***).

4.4.1.2 Cabeceras y pie de página

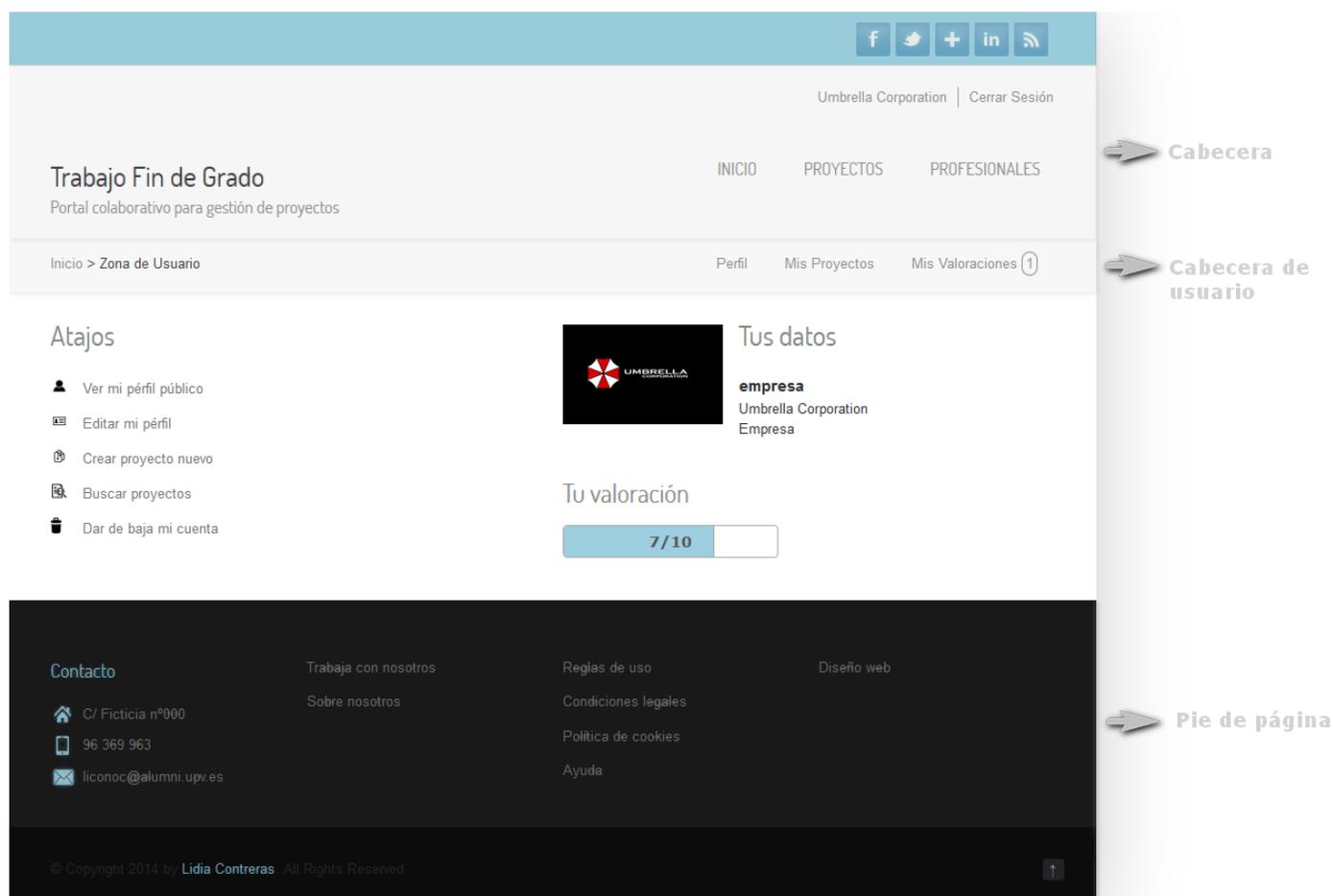


Ilustración 4-19 - Capa de presentación: Cabeceras y pie de página

La estructura de archivos y funciones de Wordpress permite crear una cabecera y un pie de página común para todas las páginas de la web de modo que solo hay que invocar la función `get_header()`¹⁶ o `get_footer()`¹⁷ en el lugar donde deben ir la cabecera o el pie de página, respectivamente y añadir los *widgets*¹⁸, *plugins* o el código que se considere necesario.

4.4.1.3 Tema adaptativo

Dado que la web debe ser accesible desde cualquier dispositivo, adaptándose al tamaño de la pantalla se ha optado por utilizar SimpleCorp, un tema adaptativo ya creado para Wordpress.

Este tema utiliza las reglas `@media` de CSS3 que consiguen mejorar la experiencia del usuario que entra en una web, independientemente del dispositivo que se use, adaptando la web a las resolución y medidas del dispositivo que se esté usando definiendo conjuntos de estilos dependiendo de propiedades comunes de los dispositivos y ajustando propiedades específicas para el alto, ancho, la relación de aspecto, el número de colores disponibles, etc. (Ventura, 2012)

¹⁶ http://codex.wordpress.org/Function_Reference/get_header

¹⁷ http://codex.wordpress.org/Function_Reference/get_footer

¹⁸ En informática un widget es un pequeño programa que da acceso a funciones que usas frecuentemente. (Karen, 2008)

La implementación de estas características se basa principalmente en crear unas reglas CSS diferentes para cada resolución que se activarán dinámicamente según el tamaño de la pantalla o la ventana del navegador.

```
@import url("library/css/main-css.css");
@import url("library/css/768.css");
@import url("library/css/320.css");
@import url("library/css/480.css");
```

Se crea un archivo CSS para cada resolución y en él se generan las reglas que se aplicarán. Por ejemplo para una resolución de hasta 320px y para un dispositivo posicionado de manera vertical (*portrait*) utilizamos:

```
/*-----*/
/* #MOBILE 320px (Portrait)*/
/*-----*/
@media only screen and (max-width: 480px) , screen and (max-device-width: 480px) {
```

De este modo tendremos una vista de la web adaptada a cada pantalla. Por ejemplo para las resoluciones de 320px, 768px y 1024px tendremos las siguientes versiones de la web, donde se puede apreciar la diferencia en los menús y las imágenes que cambian de tamaño y se acoplan al tamaño de la pantalla.



Ilustración 4-21 - Capa de presentación: Resolución 320px



Ilustración 4-20 - Capa de presentación: Resolución 768px

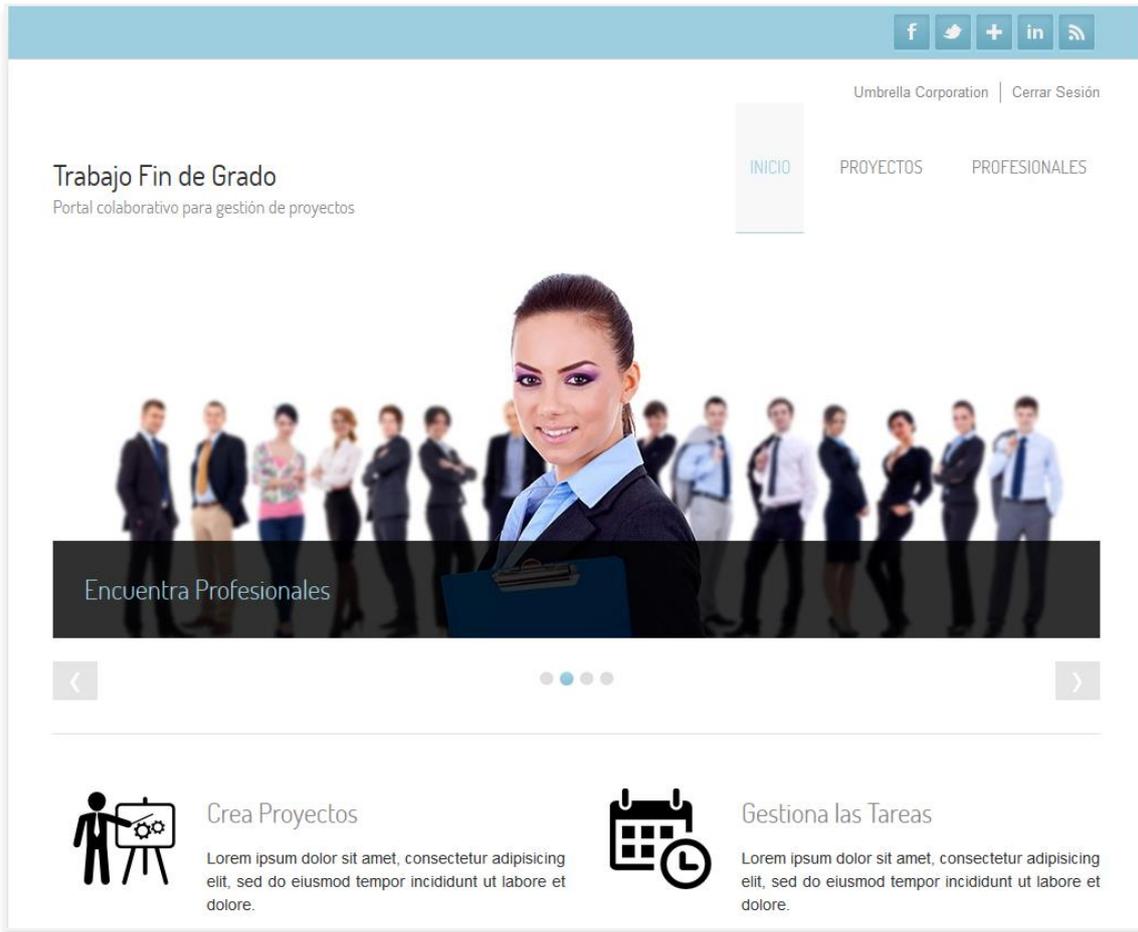


Ilustración 4-22 - Capa de presentación: Resolución 1024px

4.4.1.4 Formularios

Los formularios de la web se han creado con HTML5 de modo que la validación de los campos obligatorios o con valores especiales como números, sea automática al pulsar el botón de envío.

Ilustración 4-23 - Capa de presentación: Formularios

Ilustración 4-24 - Capa de presentación: Formularios

Los campos obligatorios quedan sombreados de manera automática mediante CSS:

```
input:invalid, textarea:invalid {  
    background-color: #EEEEEE !important;  
}
```

4.4.1.5 Valoración y progreso

La barra de valoración de un usuario o de progreso de un proyecto se genera mediante el *widget Progressbar*¹⁹ de jQuery UI, obteniendo la valoración mediante PHP y generando la barra de progreso mediante jQuery.

```
$valoraciones = $wpdb->get_results(  
    "  
        SELECT media  
        FROM valoraciones_media  
        WHERE 1 AND usuario=".$get_user_id."  
    "  
    );  
  
if(count($valoraciones)==0)  
{  
    $valoracion=0;  
}  
else  
{  
    $valoracion=$valoraciones[0]->media;  
}
```

```
<script type="text/javascript">  
jQuery(document).ready( function(){  
    $j=jQuery.noConflict();  
  
    $j("#valoracion").progressbar({  
        value: <?php echo $valoracion; ?>,  
        max: 10  
    });  
  
});  
</script>
```

¹⁹ <http://jqueryui.com/progressbar/>

4.4.2 Capa de aplicación

Para la capa de aplicación se ha utilizado PHP y jQuery, así como los *plugins* de Wordpress que se han considerado útiles.

Se ha optado por utilizar como base para las zonas de la web, las páginas²⁰ de Wordpress, de modo que para cada una de las diferentes secciones de la web (inicio de sesión, zona de usuario, perfil de usuario, etc.), se ha creado una página en Wordpress y se le ha implementado una plantilla²¹ en PHP.

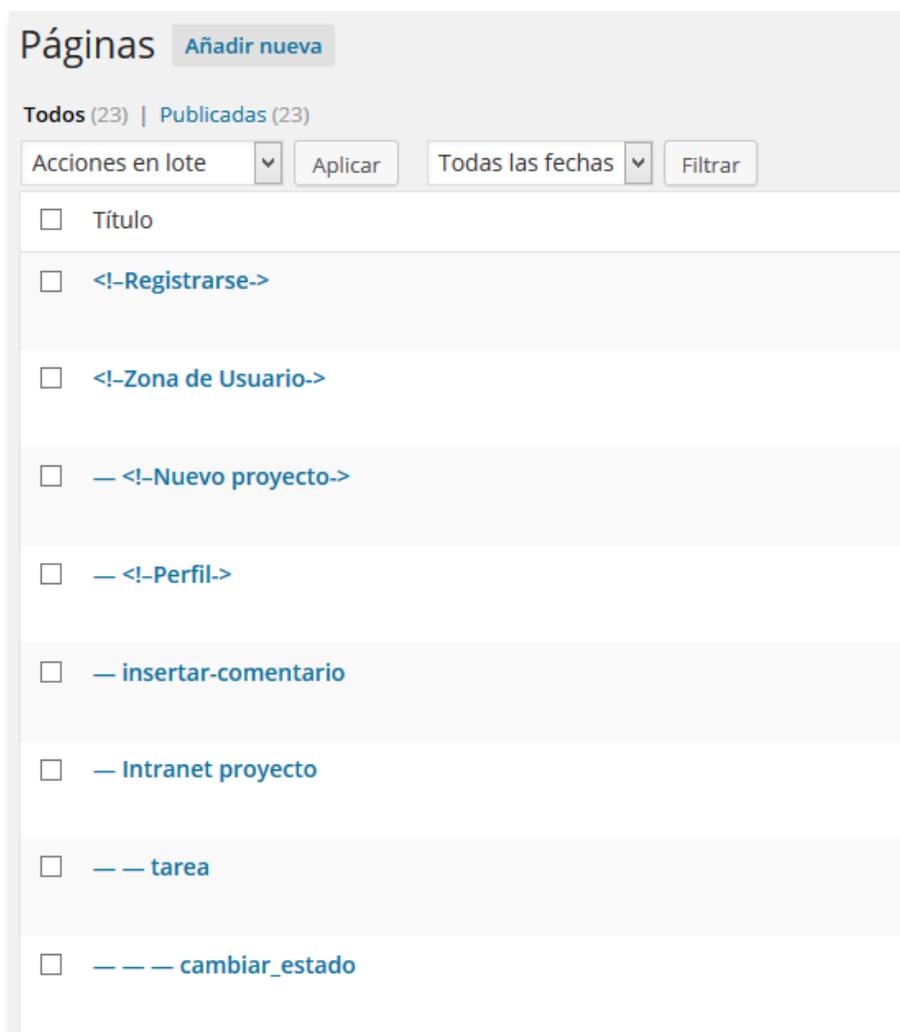


Ilustración 4-25 - Capa de aplicación: Páginas de Wordpress

²⁰ <http://codex.wordpress.org/Pages>

²¹ <http://codex.wordpress.org/Templates>

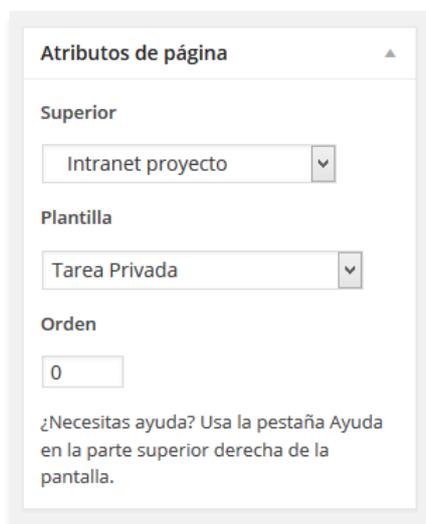


Ilustración 4-26 - Capa de aplicación: Plantillas de Wordpress

De igual modo se ha utilizado como base las entradas²² de Wordpress para crear tanto los proyectos como las tareas.

La página principal, la cabecera de la web y el pie de página son los que vienen por defecto en el tema seleccionado, con ciertas modificaciones, para adaptarla al proyecto.

4.4.2.1 Roles

Para definir los diferentes perfiles de usuario de la web se han creado tres roles en Wordpress para clasificar cada usuario que se dé de alta.

El administrador ya está definido, de modo que no es necesario implementarlo. Para el resto de los tipos de usuario (cliente, *freelance* y empresa) se ha utilizado el *plugin WPFront User Role Editor*.

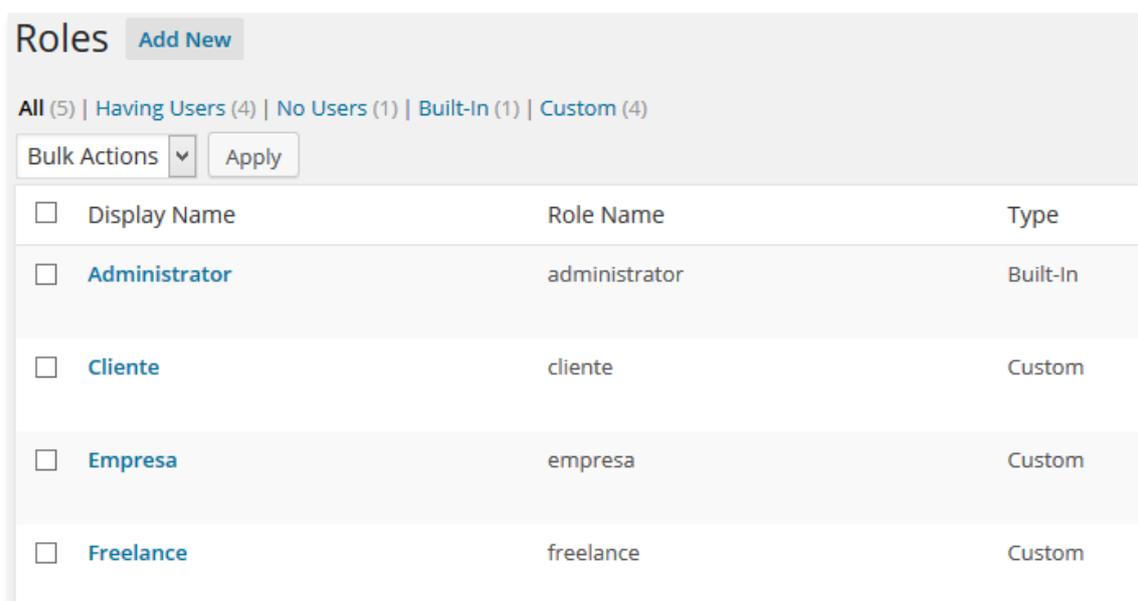


Ilustración 4-27 - Capa de aplicación: Roles

²² http://codex.wordpress.org/Administration_Panels#Posts

4.4.2.2 Proyectos y tareas

Tanto los proyectos como las tareas se han implementado utilizando como base las entradas de Wordpress. De este modo se almacenan en la base de datos con todos los datos que Wordpress necesita para su manejo.

Para crear estos nuevos tipos de entradas se han implementado sus funciones en el archivo `functions.php`²³ de Wordpress.

```
## Nuevo tipo de post: Proyectos ##
// Defino las etiquetas de los menu y formularios
//-----
$proyecto_labels = array(
    'name' => _x('Proyecto', 'post type general name'),
    'singular_name' => _x('Proyectos', 'post type singular name'),
    'add_new' => _x('Nuevo Proyecto', 'proyecto'),
    'add_new_item' => __("Nuevo Proyecto"),
    'edit_item' => __("Editar proyecto"),
    'new_item' => __("Nuevo Proyecto"),
    'view_item' => __("Ver Proyecto"),
    'search_items' => __("Buscar Proyecto"),
    'not_found' => __("No se encontraron Proyectos"),
    'not_found_in_trash' => __("No hay Proyectos"),
    'parent_item_colon' => ''
);

//Creo los argumentos para la base de datos
$proyecto_args = array(
    'labels' => $proyecto_labels,
    'public' => true,
    'publicly_queryable' => true,
    'show_ui' => true,
    'query_var' => true,
    'rewrite' => true,
    'hierarchical' => false,
    'menu_position' => null,
    'capability_type' => 'post',
    'supports' => array('title', 'excerpt', 'editor', 'thumbnail'),
    'menu_icon' => get_bloginfo('template_directory') . '/imagenes/iconos/proyecto.png' //16x16
);

//Registro el post
register_post_type('proyecto', $proyecto_args);

// Crear Tipo de proyectos
add_action( 'init', 'tx_tipo_proyecto', 0);
```

Así, en el escritorio²⁴ de Wordpress aparecen los dos nuevos tipos de entrada para poder manipularlos.

²³ http://codex.wordpress.org/Functions_File_Explained

²⁴ http://codex.wordpress.org/Dashboard_Screen



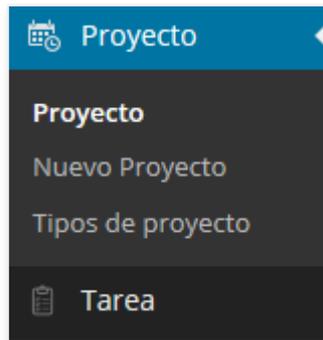


Ilustración 4-28 - Capa de aplicación: Nuevas entradas

Para añadirle más campos a Proyecto y Tarea se ha utilizado el *plugin Advanced Custom Fields*. De este modo se han añadido a los proyectos el presupuesto y el progreso, por ejemplo.

Label del campo	Nombre del campo	Tipo de campo
Propietario	propietario	User
Fecha de Inicio	fecha_de_inicio	Selector de Fecha
Fecha de Fin	fecha_de_fin	Selector de Fecha
Presupuesto	presupuesto	Number
Progreso	progreso	Number
Tareas	tareas	Post Object

[+ Añadir Campo](#)

Ilustración 4-29 - Capa de aplicación: Advanced Custom Fields

4.4.2.3 Cabecera

En la cabecera de la web, se muestran dos enlaces:

- Iniciar Sesión / Registrarse; si el usuario es anónimo.
- Nombre + apellido del usuario/Cerrar Sesión; si el usuario ha iniciado sesión.

Para conseguir esto se utiliza la función de Wordpress `is_user_logged_in()`²⁵ que devuelve si hay una sesión iniciada en el sistema. De este modo obtenemos los datos del usuario para poder generar los enlaces correspondientes.

```
<?php
global $current_user;
get_currentuserinfo();

if(!is_user_logged_in())
{
    ?>
    <ul>
        <li><a href="<?php echo get_bloginfo( 'url' ) ?>/login">Iniciar Sesión</a></li>
        <li><a href="<?php echo get_bloginfo( 'url' ) ?>/login">Registrarse</a></li>
    </ul>
    <?php
}
else
{
    ?>
    <ul>
        <li><a href="<?php echo get_bloginfo( 'url' ) ?>/zona-usuario"><?php echo $current_user->
user_firstname . " " . $current_user->user_lastname?></a></li>
        <li><a href="<?php echo get_bloginfo( 'url' ) ?>/logout">Cerrar Sesión</a></li>
    </ul>
    <?php
}
?>
```

²⁵ http://codex.wordpress.org/Function_Reference/is_user_logged_in



4.4.2.4 Cabecera de usuario

La cabecera de usuario contiene el menú de migas que permite al usuario saber en qué zona de la web se encuentra en cada momento. Esto se consigue mediante la función de Wordpress `is_page("x")`²⁶, que devuelve `true` en caso de que nos encontremos en la página x.

```
<?php if(is_page("zona-usuario")){?>
<span><a href="<?php echo home_url( '/' ); ?>">Inicio</a> > Zona de Usuario</span>
<?php }
elseif(is_page("perfil")){?>
<span><a href="<?php echo home_url( '/' ); ?>">Inicio</a> > <a href="<?php echo home_url( '/' );
?>zona-usuario">Zona de Usuario</a> > Perfil</span>
<?php }
elseif(is_page("proyectos-usuario")){?>
<span><a href="<?php echo home_url( '/' ); ?>">Inicio</a> > <a href="<?php echo home_url( '/' );
?>zona-usuario">Zona de Usuario</a> > Mis Proyectos</span>
<?php }
elseif(is_page("nuevo-proyecto")){?>
<span><a href="<?php echo home_url( '/' ); ?>">Inicio</a> > <a href="<?php echo home_url( '/' );
?>zona-usuario">Zona de Usuario</a> > Nuevo Proyecto</span>
<?php }
elseif(is_page("intranet-proyecto")){?>
<span><a href="<?php echo home_url( '/' ); ?>">Inicio</a> > <a href="<?php echo home_url( '/' );
?>zona-usuario">Zona de Usuario</a> > Intranet Proyecto</span>
<?php } elseif(is_page("tarea")){?>
<span><a href="<?php echo home_url( '/' ); ?>">Inicio</a> > <a href="<?php echo home_url( '/' );
?>zona-usuario">Zona de Usuario</a> > <a href="<?php echo home_url( '/' ); ?>
zona-usuario/intranet-proyecto?p=<?php echo S_GET['p']; ?>">Intranet Proyecto</a> > Tarea </span>
<?php } elseif(is_page("valoraciones")){?>
<span><a href="<?php echo home_url( '/' ); ?>">Inicio</a> > <a href="<?php echo home_url( '/' );
?>zona-usuario">Zona de Usuario</a> > Valoraciones </span>
<?php } ?>
```

Dentro de la zona de usuario también se incluye un menú para el usuario que ha iniciado sesión, que le permite acceder a modificar su perfil, ver sus proyectos o acceder a las valoraciones. Si el usuario tiene valoraciones pendientes de emitir se avisará con el número de valoraciones en rojo.

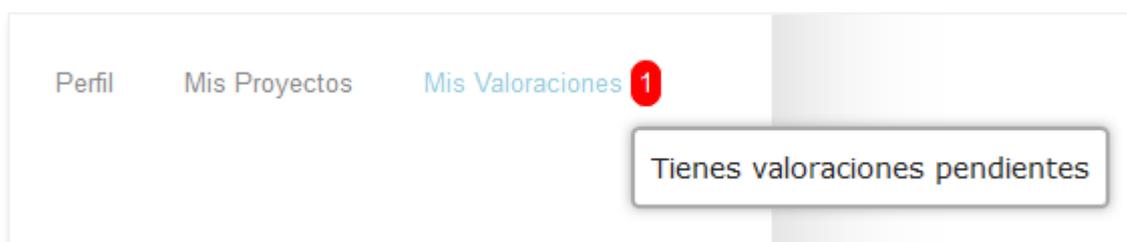


Ilustración 4-30 - Capa de aplicación: Valoraciones pendientes

²⁶ http://codex.wordpress.org/Function_Reference/is_page

4.4.2.5 Página principal

La página principal de la web tiene diferentes secciones. Las tres primeras (menú principal y logotipo, *slider*²⁷ de fotos y resúmenes) vienen por defecto en el tema de Wordpress. De este modo solo es necesario personalizarlas dentro del escritorio de Wordpress como administrador.



The screenshot shows a WordPress theme options panel titled "Display Content Boxes on Homepage". It contains the following elements:

- A checked checkbox.
- A section titled "Content Box 1 Title" with a text input field containing "Crea Proyectos".
- A section titled "Content Box 1 Text" with a text area containing the placeholder text "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore.".
- A section titled "Content Box 1 Image" with a text input field containing the URL "http://gothi-k.com/TFG/wp-content/uploads/2014/06/crea.png" and an "Upload" button.
- A placeholder image showing a person pointing at a whiteboard with gears.
- A red "x" icon and a message: "Click to 'Upload Image' button and upload Content Box 1 image."

Ilustración 4-31 - Capa de aplicación: Opciones del tema

²⁷ Un slider es un elemento de las interfaces gráficas que permiten seleccionar un valor moviendo un indicador o, en algunos casos, el usuario puede hacer clic sobre algún punto del slider para cambiar hacia ese valor. (TECNOLOGÍA)

4.4.2.5.1 Últimos proyectos

La sección de últimos proyectos está parcialmente creada en el tema, pero se ha modificado para eliminar las imágenes que incluye, ya que los proyectos de la web no contienen una imagen predefinida.

```
<?php
$sids_filtradas = $wpdb->get_results(
"
    SELECT DISTINCT(wp2.meta_value)
    FROM wp_postmeta AS wp1, wp_postmeta AS wp2
    WHERE 1
    AND wp1.post_id = wp2.post_id
    AND wp1.meta_value LIKE 'Buscando colaboradores'
    AND wp2.meta_key LIKE 'proyecto'
    ORDER BY wp2.meta_value DESC
    LIMIT 8
"
);
```

Se filtran los proyectos con una consulta MySQL para obtener los últimos ocho proyectos creados con búsqueda de colaboradores activa.

4.4.2.5.2 Profesionales mejor valorados

La sección de profesionales mejor valorados muestra los tres *freelance* y tres empresas con mejor puntuación en las valoraciones.

```
$freelance_valorados = $wpdb->get_results(
"
    SELECT DISTINCT(media.usuario) AS usuario, media.media AS media
    FROM valoraciones_media AS media, wp_usermeta AS user
    WHERE 1 AND media.usuario=user.user_id AND user.meta_value LIKE '%freelance%'
    ORDER BY media.media
    LIMIT 3
");
```

Para obtener esto, se generan varias consultas que filtran los resultados y después se consultan los datos personales del usuario, que se muestran por pantalla.

4.4.2.6 Búsquedas

Tanto la búsqueda de proyectos como la de profesionales se han generado mediante un formulario que se devuelve a la misma página mediante POST²⁸ para filtrar los resultados.

```
<form id="buscador" action="<?php echo home_url( '/' ); ?>proyectos" method="post">
<select name="buscador_campos" id="buscador_campos">
  <option value="nombre" <?php if($buscador_campos==" " || $buscador_campos=="nombre") echo "
selected"; ?>>Nombre</option>
  <option value="tipo" <?php if($buscador_campos=="tipo") echo " selected"; ?>>Tipo de Proyecto
  </option>
  <option value="localizacion" <?php if($buscador_campos=="localizacion") echo " selected"; ?>>
Localización
  </option>
</select>
<input name="buscador_texto" <?php if($buscador_texto!="") echo "value=\"".$buscador_texto."\"";
?> type="text" id="buscador_texto" />
<button id="buscador_boton">Buscar</button>
</form>
```

Según los datos seleccionados para la búsqueda se realizan las consultas pertinentes y se muestran los resultados cargando diez más cada vez que se pulsa el botón del final de la página.

4.4.2.7 Perfil de usuario

Al perfil de un usuario se accede mediante GET con el alias del usuario, que se utiliza posteriormente en la función `get_user_by()`²⁹ de Wordpress para obtener sus datos.

```
$usuario = get_user_by( 'slug' , $alias_usuario );

$user_roles = $usuario->roles;
$user_role = array_shift($user_roles);

if($user_role=="administrator")
{
  //header('Location: '.home_url( '/' ));
  ?><script type="text/javascript">
  <!--
  window.location = "<?php echo home_url( '/' ); ?>"
  //-->
  </script><?php
}

$get_user_id = $usuario->ID;
$nombre=$usuario->user_firstname;
$apellidos=$usuario->user_lastname;
$email=$usuario->user_email;
$web=$usuario->user_url;
```

Se mostrarán los perfiles de todos los usuarios registrados en la web. Sin embargo si se intenta acceder al de un administrador el portal nos redirige a la página principal.

²⁸ POST es un método de peticiones HTTP.

²⁹ http://codex.wordpress.org/Function_Reference/get_user_by



4.4.2.8 Página de proyecto pública

Para mostrar los datos públicos de un proyecto se ha optado por seguir el modo de mostrar entradas individuales en Wordpress (`single.php`) y se ha creado el archivo `single-proyecto.php`, que permite acceder a los datos del proyecto al que se acceda, mediante el *loop*³⁰ de Wordpress.

```
<?php if (have_posts()) : while (have_posts()) : the_post(); ?>

<?php
$хid_propietario=$post->post_author;
$хid_proyecto=get_the_ID();
$хpropietario=get_user_by('id', $хid_propietario);
$хrol=$хpropietario->roles[0];
$хdatos_propietario=$хpropietario->data;
$хalias=$хdatos_propietario->user_nicename;
$хdescripcion=get_the_content();
$хtipo_proyecto=get_the_terms( $хid_proyecto, 'tipo'); //array

$хcustom_proyecto = $wpdb->get_results(
    "
        SELECT meta_key, meta_value
        FROM wp_postmeta
        WHERE 1 AND post_id=".$хid_proyecto."
        AND (
            (meta_key LIKE 'fecha_de_inicio') OR
            (meta_key LIKE 'fecha_de_fin') OR
            (meta_key LIKE 'presupuesto') OR
            (meta_key LIKE 'progreso') OR
            (meta_key LIKE 'tareass')
        )
    "
);
```

Utilizando AJAX con jQuery se realiza la inscripción a una tarea de forma dinámica si un usuario ha iniciado sesión. Los resultados de la inscripción se devolverán en JSON.

³⁰ http://codex.wordpress.org/The_Loop

```

$j(".inscripcion").click( function(){
    var boton=$j(this);
    var tarea=$j(this).prev(".inscripcion_tarea").val();
    $j(boton).hide();
    $j(".loader").show();

    $j.post("<?php echo home_url( '/' ); ?>proyectos/inscripcion",
        {
            nombre:tarea
        },
        function(datos){
            if (datos.status=="OK")
            {
                alert("Te has inscrito correctamente en la tarea.");
                $j(".loader").hide();
                $j("#candidatos_"+tarea).html('<span>Número de candidatos apuntados:
</span>'+datos.candidatos);
            }
            else
            {
                alert("Se ha producido un error en el servidor ("<+datos.texto+").");
            }
        }, "json");
});

```

4.4.2.9 Alta de usuario

Para las altas de los nuevos usuarios se usa el *plugin Theme My Login*, que aporta un formulario que sustituye al de Wordpress para utilizarlo en el *frontpage* de la web.

Un usuario de Wordpress tiene unos campos predefinidos como el nombre, apellidos, dirección de correo electrónico, etc. La mayoría de estos campos se utilizan en la web, pero algunos como el currículum o el NIF no existen. Para crearlos se ha utilizado el *plugin Cimy User Extra Fields*.

Campos Adicionales

Invertir Selección Cambiar orden Borrar campos seleccionados

Orden	Nombre - Valor - Tipo	Etiqueta - Descripción	Reglas
<input type="checkbox"/> Orden 1	Nombre NIF Valor Tipo text	Etiqueta NIF/CIF Descripción	<input type="checkbox"/> Longitud mínima [1-50000]: <input type="checkbox"/> Longitud exacta [1-50000]: <input checked="" type="checkbox"/> Longitud máxima [1-50000]: 250 <input checked="" type="checkbox"/> Puede estar vacío <input type="checkbox"/> Checar formato de e-mail Puede ser modificado <input type="checkbox"/> Debe ser igual a: <input type="checkbox"/> Distinción mayúsculas y minúsculas <input type="checkbox"/> Expresión regular <input checked="" type="checkbox"/> Mostrar el campo en el registro <input checked="" type="checkbox"/> Mostrar el campo en el perfil del usuario <input checked="" type="checkbox"/> Show the field in Users Extended section <input type="checkbox"/> Show the field in the search engine <input type="checkbox"/> Show the field in the blog Show the field if the role is at least: Anónimo <input type="checkbox"/> Send an email to the admin if the user changes its value Advanced options:

Ilustración 4-32 - Capa de aplicación: Cimy User Extra Fields

El formulario de registro de usuario cambiará dependiendo del tipo de usuario que se haya elegido para dar de alta, de modo que un *freelance* tenga currículum y una empresa descripción de empresa.

```
## Definir vista de perfil según rol ##
# $unique_id contiene la id del objeto del formulario #
# $rol es administrator, cliente, empresa o freelance
switch($unique_id)
{
    //Nombre
    case "cimy_uef_wp_1":
        echo $obj_label;
        break;

    //Apellidos
    case "cimy_uef_wp_2":
        if($TipoUsuario=="cliente" || $TipoUsuario=="freelance")
            echo $obj_label;
        break;
}
```

4.4.2.10 Inicio de sesión

El inicio de sesión va determinado por un *plugin* llamado **Sidebar Login**.

Para este *plugin* se ha creado un *sidebar* de Wordpress en el archivo `functions.php`.

```
## Sidebar Login ##
function login_register_sidebars(){
    register_sidebar(array(
        "name" => "login",
        "id" => "login",
        "descripcion" => "Sidebar de Inicio de Sesión",
        "class" => "login",

    ));
}
add_action('widgets_init','login_register_sidebars');
```

Este *sidebar* se ha añadido a la plantilla de inicio de sesión de modo que aparezca en el lado izquierdo de la página.

```
<div id="login" class="one-half">
    <?php get_sidebar('login'); ?>
</div>
```

4.4.2.11 Zona de usuario

La zona de usuario se personaliza según el usuario que ha iniciado sesión. Para ello se utiliza la función `is_user_logged_in()` para comprobar que haya una sesión iniciada, y en caso afirmativo la función `get_currentuserinfo()`³¹ que dejará en la variable global `$current_user` toda la información del usuario actual.

```
/*
 * Template Name: Zona de Usuario
 */
if(!is_user_logged_in())
{
    //header('Location: '.home_url( '/' ).'login');
    ?><script type="text/javascript">
    <!--
    window.location = "<?php echo home_url( '/' ); ?>login"
    //-->
    </script><?php
}

get_header();
get_header('usuario');

global $current_user;
get_currentuserinfo();

$user_info = get_userdata($current_user->ID);
$user_roles = $user_info->roles;
$user_role = array_shift($user_roles);
$get_user_id = $current_user->ID;
```

4.4.2.12 Nuevo proyecto

```
<form id="nuevo_proyecto" name="nuevo_proyecto" method="post" action="<?php echo home_url( '/' ); ?>zona-usuario/nuevo-proyecto">
    <h1>Nuevo proyecto
        <span>Los campos con * son obligatorios.</span>
    </h1>
    <label for="nombre_proyecto"> <span> Nombre* </span> </label>
    <input type="text" id="nombre_proyecto" name="nombre_proyecto" required />
    <label for="resumen_proyecto"> <span> Resumen* </span> </label>
    <textarea id="resumen_proyecto" name="resumen_proyecto" rows="4" cols="50" maxlength="200"
required></textarea>
    <label for="descripcion_proyecto"> <span> Descripción detallada </span></label>
    <textarea id="descripcion_proyecto" name="descripcion_proyecto" rows="10" cols="50" maxlength=
"5000"></textarea>
    <label for="tipo_proyecto"> <span> Tipo de proyecto* </span></label>
    <input type="text" id="tipo_proyecto" name="tipo_proyecto" required />
    <label for="finicio_proyecto"> <span> Fecha de inicio prevista</span> </label>
    <input type="text" id="finicio_proyecto" name="finicio_proyecto" />
    <label for="ffin_proyecto"> <span> Fecha de fin prevista</span> </label>
    <input type="text" id="ffin_proyecto" name="ffin_proyecto" />
    <label for="presupuesto_proyecto"> <span> Presupuesto previsto </span></label>
    <input type="number" id="presupuesto_proyecto" name="presupuesto_proyecto" step="any" min="0"
/><span>€</span>
```

³¹ [http://codex.wordpress.org/Function Reference/get_currentuserinfo](http://codex.wordpress.org/Function_Reference/get_currentuserinfo)



```

<!-- TAREAS -->
<h2> Tareas
<span style="color:red;">Los campos con * son obligatorios.</span>
<span>Arrástralas para ordenarlas.</span>
</h2>
<div id="tareas">

<div class="tarea">
<h3>Tarea 1</h3>
<div>
  <label for="nombre_tarea_1"> <span> Nombre* </span> </label>
  <input type="text" id="nombre_tarea_1" name="nombre_tarea_1" class="input_nombre_tarea"
required />
  <label for="descripcion_tarea_1"> <span> Descripción detallada* </span></label>
  <textarea id="descripcion_tarea_1" name="descripcion_tarea_1" rows="10" cols="50" maxlength=
"5000" class="input_descripcion_tarea" required></textarea>
  <label for="clase_tarea_1"> <span> Clase de tarea* </span></label>
  <input type="text" id="clase_tarea_1" class="clase_tarea" name="clase_tarea_1" required />
  <label for="tiempo_previsto_1"> <span> Tiempo previsto </span></label>
  <input type="number" id="tiempo_previsto_1" name="tiempo_previsto_1" step="any" min="0" />
<span>horas</span>
  <label for="sueldo_asignado_1"> <span> Sueldo asignado </span></label>
  <input type="number" id="sueldo_asignado_1" name="sueldo_asignado_1" step="any" min="0" />
<span>€</span>
  <label for="estado_tarea_1"> <span> Estado tarea </span></label>
  <input type="checkbox" name="estado_tarea_1" id="estado_tarea_1" value="activo">Poner en
estado "Buscando colaboradores"
  <div class="eliminar_tarea"><span> Eliminar tarea</span></div>
</div>
</div>

```

Todos los usuarios registrados pueden crear un nuevo proyecto, de modo que una vez confirmado que hay una sesión iniciada se muestra al usuario un formulario para crear el proyecto.

Un proyecto tiene como mínimo una tarea. Para crear nuevas tareas se ha utilizado *Accordion*³² de jQuery UI.

```

/*Tareas*/
$( "#tareas" ).accordion({ header: "> div > h3"})
.sortable({
  axis: "y",
  handle: "h3",
  stop: function( event, ui ) {
    // IE doesn't register the blur when sorting
    // so trigger focusout handlers to remove .ui-state-focus
    ui.item.children( "h3" ).triggerHandler( "focusout" );
    // Refresh accordion to handle new order
    $( this ).accordion( "refresh" );
  }
});

```

Para que se puedan generar nuevas tareas y se añadan o eliminen dinámicamente del acordeón, se ha utilizado la función `.on()`³³ de jQuery.

³² <http://jqueryui.com/accordion>

³³ <http://api.jquery.com/on/>

```

var num_tareas = 1;
var num_tareas_creadas=1;

$( "#nuevo_proyecto" ).on( "click", "#nueva_tarea", function(){
    num_tareas++;
    num_tareas_creadas++;
    var tarea_contenedor = $( ".tarea" );
    var tarea = "<div class='\"tarea\"'><h3>Tarea "+num_tareas_creadas+"</h3><div><label
for='\"nombre_tarea_\"+num_tareas_creadas+'\"> <span> Nombre* </span> </label>";
    tarea += "<input type='\"text\"' id='\"nombre_tarea_\"+num_tareas_creadas+'\"
name='\"nombre_tarea_\"+num_tareas_creadas+'\" class='\"input_nombre_tarea\" required/'>";

    tarea += "<label for='\"descripcion_tarea_\"+num_tareas_creadas+'\"> <span> Descripción
detallada* </span></label>";
    tarea += "<textarea id='\"descripcion_tarea_\"+num_tareas_creadas+'\"
name='\"descripcion_tarea_\"+num_tareas_creadas+'\" rows='\"10\"' cols='\"50\"' maxlength='\"5000\"'
class='\"input_descripcion_tarea\" required></textarea><label for='\"clase_tarea_\"+
num_tareas_creadas+'\"> <span> Clase de tarea* </span></label>";
    tarea += "<input type='\"text\"' id='\"clase_tarea_\"+num_tareas_creadas+'\"
class='\"clase_tarea\"' name='\"clase_tarea_\"+num_tareas_creadas+'\" required />";
    tarea += "<label for='\"tiempo_previsto_\"+num_tareas_creadas+'\"> <span> Tiempo previsto
</span></label>";
    tarea += "<input type='\"number\"' id='\"tiempo_previsto_\"+num_tareas_creadas+'\"
name='\"tiempo_previsto_\"+num_tareas_creadas+'\" step='\"any\"' min='\"0\"' /><span>horas</span>";
    tarea += "<label for='\"sueldo_asignado_\"+num_tareas_creadas+'\"> <span> Sueldo asignado
</span></label>";
    tarea += "<input type='\"number\"' id='\"sueldo_asignado_\"+num_tareas_creadas+'\"
name='\"sueldo_asignado_\"+num_tareas_creadas+'\" step='\"any\"' min='\"0\"' /><span>€</span>";
    tarea += "<label for='\"estado_tarea_\"+num_tareas_creadas+'\"> <span> Estado tarea </span>
</label>";
    tarea += "<input type='\"checkbox\"' name='\"estado_tarea_\"+num_tareas_creadas+'\"
id='\"estado_tarea_\"+num_tareas_creadas+'\" value='\"activo\"'>Poner en estado \"Buscando
colaboradores\"<div class='\"eliminar_tarea\"'><img src='\"<?php bloginfo('template_directory')?>
/imagenes/iconos/menos.png\" /><span> Eliminar tarea</span></div></div></div>";
    $( "#tareas" ).append( $( tarea ) );
    $( "#tareas" ).accordion( "refresh" );
    $( "#num_tareas" ).val( num_tareas_creadas );
});

$( "#tareas" ).on( "click", ".eliminar_tarea", function(){
    if( num_tareas==1 )
    {
        alert( "Tiene que haber al menos una tarea" );
    }
    else
    {
        $( this ).parent().parent().remove();
        $( "#tareas" ).accordion( "refresh" );
        num_tareas--;
    }
});

```

Los datos del formulario se envían por POST al mismo archivo que los analiza, inserta el proyecto y sus tareas y muestra el resultado de la operación.



```

$nombre=$_POST['nombre_proyecto'];
$resumen=$_POST['resumen_proyecto'];
$descripcion=$_POST['descripcion_proyecto'];
$tipo=$_POST['tipo_proyecto'];
$inicio=$_POST['inicio_proyecto'];
$ffin=$_POST['ffin_proyecto'];
$presupuesto=$_POST['presupuesto_proyecto'];

## Si $nombre tiene valor, generamos un nuevo proyecto con los datos ##
if(isset($nombre))
{
    $proyecto = array(
        'post_title' => $nombre,
        'post_author' => $get_user_id,
        'post_content' => $descripcion,
        'post_excerpt' => $resumen,
        'post_status' => 'publish',
        'post_type' => 'proyecto',
        'post_date' => date("Y-m-d H:i:s")
    );

    $proyecto_id = wp_insert_post($proyecto);
}

```

4.4.2.13 Proyectos del usuario

En esta página se muestran al usuario los proyectos de los que es propietario, así como aquellos en los que se ha inscrito como candidato y los que participa como colaborador asignado.

Para obtener los proyectos de los cuales es propietario se utiliza el *loop* de Wordpress, personalizando la *Query*³⁴ poniendo como autor del post el usuario actual.

```

##proyectos en los que es propietario
// The Query
$args=array(
    'post_type' => 'proyecto',
    'author' => $get_user_id
);

$query = new WP_Query($args);

```

Para los proyectos como colaborador o candidato se generan consultas MySQL mediante la función de Wordpress `$wpdb->get_results()`³⁵.

```

##proyectos en los que es colaborador
$custom_tarea = $wpdb->get_results(
    "
        SELECT post_id
        FROM wp_postmeta
        WHERE 1 AND meta_key LIKE 'colaborador_asignado'
        AND meta_value = ".$get_user_id."
    "
);

```

³⁴ http://codex.wordpress.org/Function_Reference/query_posts

³⁵ http://codex.wordpress.org/Class_Reference/wpdb

4.4.2.14 Intranet de proyecto

La plantilla de intranet de un proyecto recibe mediante GET el *slug*³⁶ del proyecto, con el que genera una consulta con la función `get_post()`³⁷.

```
$args=array(
    'name' => $slug_proyecto,
    'post_type' => 'proyecto',
    'post_status' => 'publish',
    'numberposts' => 1
);
$my_posts = get_posts($args);
if( $my_posts ) {
    $id_proyecto=$my_posts[0]->ID;
    $id_propietario=$my_posts[0]->post_author;
    $titulo_proyecto=$my_posts[0]->post_title;
    $fecha_proyecto=$my_posts[0]->post_date;
    $resumen_proyecto=$my_posts[0]->post_excerpt;
}
```

Con la *id* del proyecto, obtiene las tareas y comprueba que el usuario que está intentando acceder es el candidato asignado de alguna de ellas o el propietario del proyecto. Si no es así le devolverá a la zona de usuario.

```
/*Comprobamos que el usuario pertenece al proeycto */
if ($id_propietario!=$get_user_id)
{
    $pertenece=false;
    foreach ($datos_tareas as $tarea)
    {
        if($tarea->meta_value==$get_user_id)
        {
            $pertenece=true;
            $tipo_usuario="asignado";
            break;
        }
    }
    if(!$pertenece)
    {
        ?><script type="text/javascript">
        <!--
        window.location = "<?php echo home_url( '/' ); ?>zona-usuario"
        //-->
        </script><?php
    }
}

else $tipo_usuario="propietario";
```

³⁶ El concepto de Slug es un término adaptado del periodismo anglosajón y hace referencia al título de una noticia o artículo en el que se han sustituido los espacios en blanco por guiones y se han eliminado todos los caracteres que no sean letras o números. (atCreativa, 2010)

³⁷ https://codex.wordpress.org/Function_Reference/get_post



4.4.2.14.1 Mensajes

Tanto en la intranet del proyecto como en la de las tareas, los colaboradores y el propietario pueden enviar mensajes que el resto pueden leer al entrar. Para implementarlos se ha utilizado como base los comentarios de las entradas de Wordpress. De este modo son accesibles desde el escritorio de Wordpress.

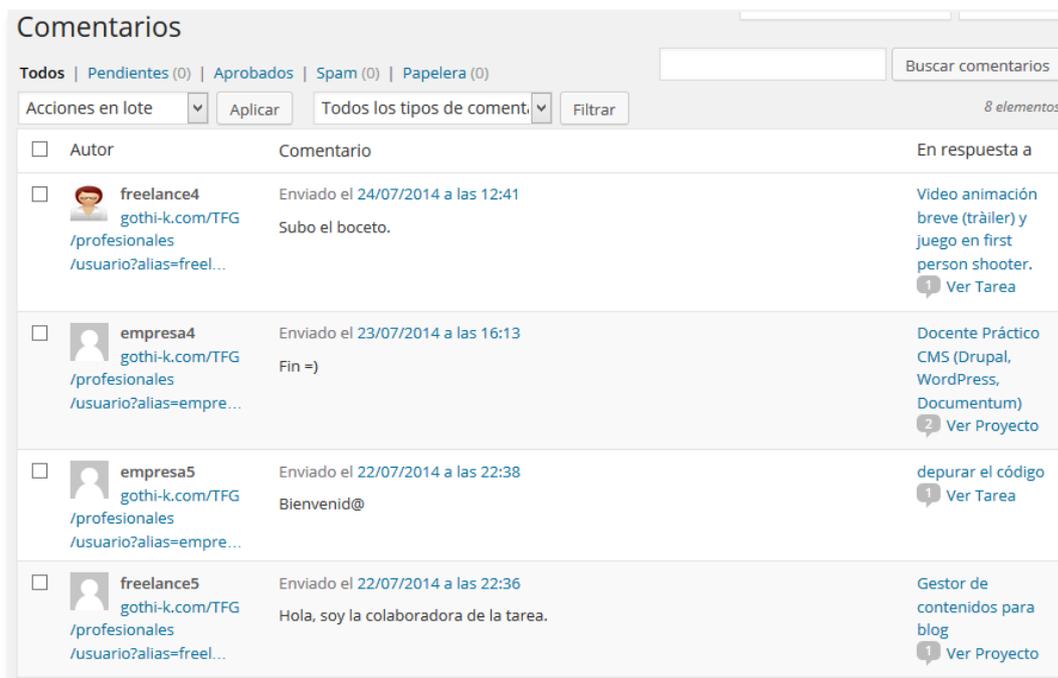


Ilustración 4-33 - Capa de aplicación: Comentarios

Estos comentarios se enviarán y cargarán en la página de manera dinámica con AJAX.

```

$jQuery("#lista-comentarios ul").load( "<?php echo home_url( '/' ); ?>zona-usuario/lista_comentarios",
{id_post: <?php echo $id_proyecto ?> });

$jQuery("#enviar_comentario").click(function(event){
    event.stopPropagation(); //preventDefault() detiene la ejecución de los comprobadores de
    campos también
    var comentario=$jQuery("#texto_comentarios").val();
    var datos_comentario = {comentario:comentario, usuario:<?php echo $get_user_id; ?>, post:<?php
    echo $id_proyecto; ?>};
    if(comentario!="")
    {
        event.preventDefault();
        $jQuery(".loader").show();

        $jQuery.post("<?php echo home_url( '/' ); ?>zona-usuario/insertar-comentario",
        {
            comentario:datos_comentario
        },
        function(datos){
            if (datos.status=="OK")
            {
                $jQuery("#lista-comentarios ul").load( "<?php echo home_url( '/' ); ?>
                zona-usuario/lista_comentarios",
                { id_post: <?php echo $id_proyecto ?> },
                function() {
                    $jQuery(".loader").hide();
                });
            }
            else
            {
                alert("Se ha producido un error en el servidor ("<+datos.texto+>).");
            }
        }, "json");
    }
});

```

4.4.2.15 Intranet de tarea

La intranet de la tarea recibe por GET el *slug* de la tarea y el del proyecto al que pertenece, con los que procede a recuperar los datos de la tarea.

```
$datos_tarea = $wpdb->get_results(
    "
        SELECT meta_key, meta_value
        FROM wp_postmeta
        WHERE 1 AND post_id=".$id_tarea."
    "
);

foreach($datos_tarea as $array_tarea)
{
    switch($array_tarea->meta_key)
    {
        case "estado":
            $estado_tarea=$array_tarea->meta_value;
            break;

        case "proyecto":
            $proyecto_tarea=$array_tarea->meta_value;
            break;

        case "recursos":
            $recursos_tarea=unserialize($array_tarea->meta_value);
            break;

        case "colaborador_asignado":
            $colaborador_tarea=$array_tarea->meta_value;
            break;

        case "colaboradores_candidatos":
            $colaboradores_tarea=unserialize($array_tarea->meta_value);
            break;

        default:
            break;
    }
}
```

El colaborador de la tarea, si hubiera alguno asignado, solo podrá acceder a la misma si está en estado “En proceso”.

4.4.2.15.1 Cambio de estado de la tarea

Dependiendo del estado de la tarea, se le mostrarán al propietario de la tarea diferentes enlaces para realizar las operaciones pertinentes, por ejemplo “Seleccionar candidato de la lista” o “Pausar tarea”.

```

if($tipo_usuario=="propietario")
{
?>
<div id="opciones_tarea">
<?php
if($estado_tarea=="Buscando colaboradores")
{
?>
<a href="#" id="enlace_seleccionar">Seleccionar candidato de la lista (Detiene la
búsqueda).</a>
<?php
}

if($estado_tarea=="En proceso" || $estado_tarea=="Pausada")
{
?>
<a href="#" id="enlace_seleccionar">Seleccionar nuevo candidato de la lista
(Descarta el actual colaborador)</a>

```

4.4.2.15.2 Lista de candidatos

El propietario verá una lista de los candidatos inscritos en la tarea y tendrá tres acciones posibles para cambiarlos de estado “Descartar”, “Asignar” y “Hacer finalista”. Los enlaces cambiarán el estado de las candidaturas dinámicamente mediante AJAX.

```

$j("#hacer_finalista").click(function(){
    $j("#cargador").show();
    var span=$j(this);
    var candidato = $j(this).parent().parent("tr").attr("id").substr(10);
    var datos_estado = {estado:"hacer_finalista", post:<?php echo $id_tarea; ?>, candidato:
candidato};
    $j.post("<?php echo home_url( '/' ); ?>zona-usuario/intranet-proyecto/tarea/cambiar_estado"
    {
        datos_estado:datos_estado
    },
    function(datos){
        if (datos.status=="OK")
        {
            $j(span).parent().prev("td").empty().append("Finalista").css("color", "green");
            $j(span).remove();
            $j("#cargador").hide();
        }
        else
        {
            $j("#cargador").hide();
            alert("Se ha producido un error en el servidor (" +datos.texto+").");
        }
    }, "json");
});

```

4.4.2.15.3 Lista de archivos

El propietario y el colaborador compartirán una lista de archivos a la que podrán agregar los recursos que sean necesarios.

Los archivos se cargarán mediante AJAX al servidor y se listarán de este modo también.

```
$j("#subir_archivo_submit").click(function(event){
    event.preventDefault();
    $j("#cargador").show();
    var formData = new FormData($j("#subir_archivo form")[0]);
    formData.append("t", "<?php echo $id_tarea ?>");

    $j.ajax({
        url: "<?php echo home_url( '/' ); ?>load-files/",
        type: 'POST',
        data: formData,
        mimeType:"multipart/form-data",
        contentType: false,
        cache: false,
        processData:false,
        dataType:"json",
        success: function(datos){
            if (datos.status=="OK")
            {
                $j("#lista_archivos ol").load( "<?php echo home_url( '/' ); ?>
zona-usuario/lista_archivos/",
                { id_post: <?php echo $id_tarea ?> },
                function() {
                    $j("#cargador").hide();
                });
            }
            else
            {
                $j("#cargador").hide();
                alert("Se ha producido un error en el servidor (" +datos.texto+").");
            }
        }
    });
});
```

4.4.2.16 Valoraciones

Cada vez que el propietario de una tarea la finaliza o abandona, o descarta el actual colaborador, se generarán dos valoraciones vacías en la base de datos; una del propietario al colaborador y otra viceversa.

```
$custom_tarea = $wpdb->get_results(
    "INSERT INTO valoraciones
    (emisor, receptor, tarea, nota)
    VALUES
    (".$propietario.", ".$colaborador_asignado.", ".$post.", NULL),
    (".$colaborador_asignado.", ".$propietario.", ".$post.", NULL)"
);
```

Cuando un usuario inicie sesión, la cabecera de usuario comprobará si tiene valoraciones pendientes de emitir y las mostrará en el menú de usuario.

Al entrar en la página de valoraciones, el usuario podrá ver su lista de valoraciones recibidas de los proyectos en los que haya participado, así como emitir las valoraciones que tenga pendientes.

```
var pendientes=<?php echo $pendientes; ?>;

Sj(".valorar").click(function(){
    $j("#cargador").show();
    var fila=$j(this).parent().parent();
    var nota=$j(this).prevAll(".nota").val();
    var receptor=$j(this).prevAll(".id_receptor").val();
    var tarea=$j(this).prevAll(".id_tarea").val();
    var valoracion = { nota:nota, tarea:tarea, emisor:<?php echo $get_user_id; ?>, receptor:
receptor};
    $j.post("<?php echo home_url( '/' ); ?>zona-usuario/valoraciones/valorar",
    {
        valoracion:valoracion
    },
    function(datos){
        if (datos.status=="OK")
        {
            alert("valoración emitida con éxito.");
            if(pendientes>1)
            {
                $j(fila).remove();
                pendientes--;
            }
            else
            {
                $j("#pendientes").empty().html('No tienes valoraciones pendientes de emitir.');
```

4.4.2.17 Seguridad

Las zonas privadas de la web solo son accesibles para el administrador y los usuarios registrados. Las intranets de los proyectos y tareas solo lo son para aquellos usuarios inscritos o propietarios de las mismas.

Cada una de las plantillas de una zona privada comprobará si hay un usuario activo en el momento en que se abra la página y si no es así o el usuario no tuviera los permisos adecuados, se redirigirá automáticamente a una zona accesible para él. De este modo un usuario no autorizado no puede ver el contenido de la página sin permiso.

4.4.3 Capa de persistencia

Para la manipulación y almacenamiento de los datos de nuestra web, se utiliza una base de datos creada en MySQL con la siguiente estructura:

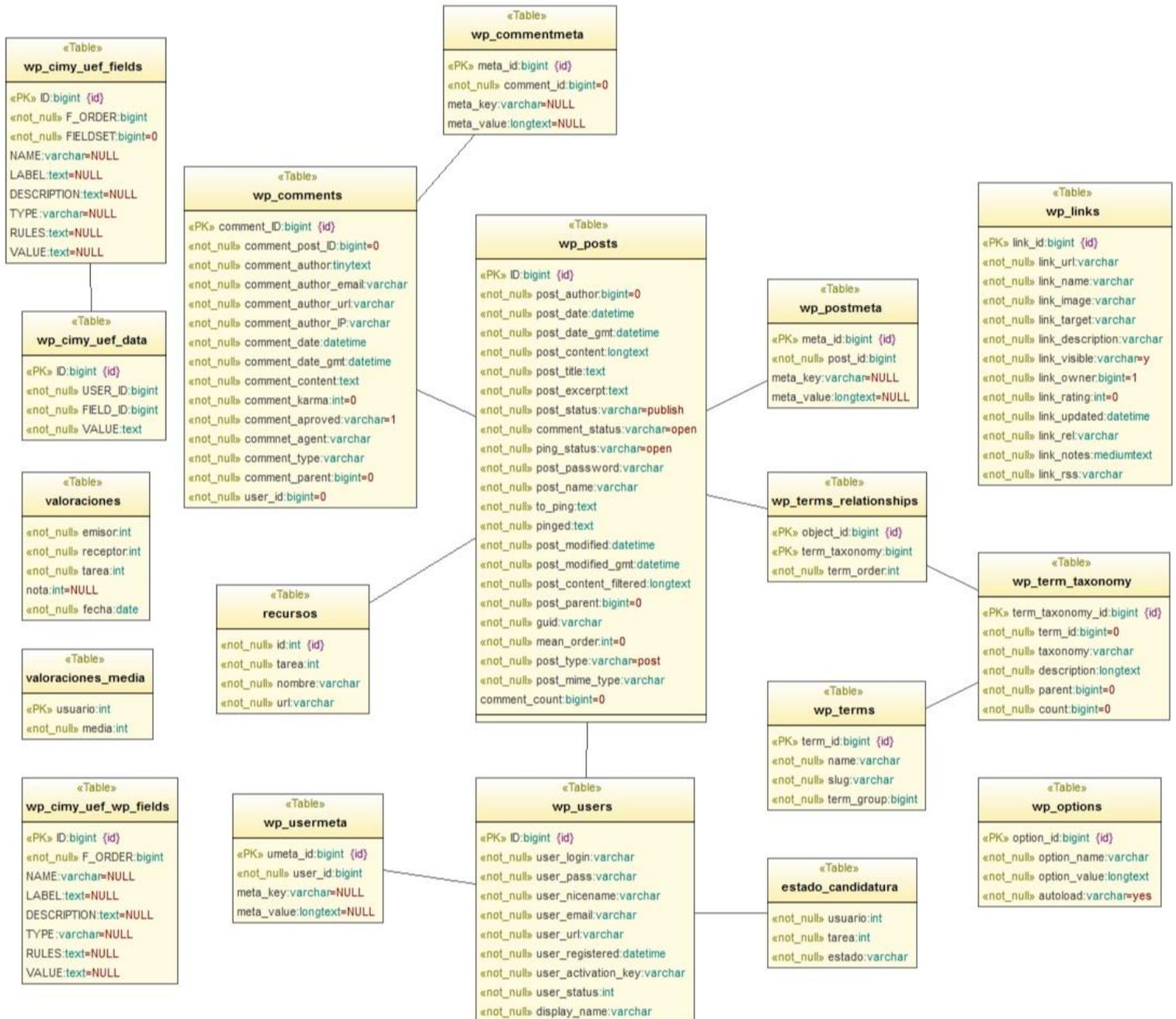


Ilustración 4-34 - Capa de persistencia: Base de datos

La base de datos contiene tablas predefinidas en Wordpress, tablas pertenecientes a plugins y otras creadas para diferentes fines como “valoraciones”.

5 Conclusiones

En este trabajo se ha tratado de abordar el problema de colaborar en proyectos puntuales sin tener que realizar contratos o crear empresas.

Del mismo modo se requería que el portal creado estuviese disponible desde cualquier dispositivo con navegador.

De este modo se ha creado una web adaptativa, accesible tanto desde un ordenador de sobremesa como de un dispositivo móvil, disponible siempre desde internet y que permite a cualquier persona registrarse y compartir o colaborar en proyectos.

Se ha utilizado Wordpress como plataforma base para la web, ya que sus características funcionales básicas hacían que el portal fuera mucho más seguro y con gran variedad de funciones, que se han ampliado integrándolo con diversas tecnologías como AJAX, jQueryUI, HTML5 y CSS3 que hacen que la web sea dinámica y mejore visualmente de forma notable.

Podríamos concluir que los objetivos marcados en la introducción de este trabajo han sido cumplidos en su totalidad.

5.1 Posibles ampliaciones

En una futura mejora de la web, poniendo como meta la explotación del portal, se podrían dar las siguientes ampliaciones:

- Pasarela de pago para realizar ingresos de las facturas de los colaboradores que trabajen en los proyectos.
- Servicios de valor añadido para los perfiles de usuario (comprobación de nivel de idiomas, recomendaciones, etc.) o para las inscripciones (salir el primero de la lista, etc).
- Aplicación para Android u otras plataformas móviles o *WebApp* (realizada con jQuery Mobile, por ejemplo), para crear un portal móvil para acceder a la intranet del portal.



6 Bibliografía

4odeFiebre. 4odeFiebre. *¿Qué es el Diseño responsive?* [En línea] <http://www.4odefiebre.com/que-es/diseno-responsive/>.

atCreativa. 2010. atCreativa. *¿Qué es slug de Wordpress?* [En línea] 28 de 07 de 2010. <http://www.atcreativa.com/blog/que-es-slug-de-wordpress/>.

German. 2013. Días de Linux. [En línea] 5 de 3 de 2013. <http://www.diasdelinux.es/mysql-base-de-datos/>.

Karen. 2008. Creando tu propio blog, desde cero. *¿Qué es un widget y cómo funciona?* [En línea] 13 de 04 de 2008. <http://karenblixen.wordpress.com/2008/04/13/que-es-un-widget/>.

Luca, Damián De. 2010. CSS3 y HTML5. *¿Qué es CSS3?* [En línea] 23 de 11 de 2010. <http://html5.dwebapps.com/que-es-css3/>.

Ravioli, Pablo. 2007. monografias.com. [En línea] 2007. <http://www.monografias.com/trabajos7/html/html.shtml>.

TECNOLOGÍA, DICCIONARIO DE INFORMÁTICA Y. DICCIONARIO DE INFORMÁTICA Y TECNOLOGÍA. *Definición de Slider.* [En línea] <http://www.alegsa.com.ar/Dic/slider.php>.

Torres, Maykel Arias. 2009. ProgramacionWeb.net. [En línea] 06 de 10 de 2009. <http://www.programacionweb.net/articulos/articulo/que-se-puede-decir-php/>.

Ventura, Pedro. 2012. Uso básico CSS3 Media Queries para crear versiones web para móvil. *Pedro Ventura.* [En línea] 12 de 6 de 2012. <http://www.pedroventura.com/desarrollo-web/uso-basico-css3-media-queries-para-crear-versiones-web-para-movil/>.

7 Anexos

7.1 Anexo I: Casos de uso

Caso de uso	Registrarse	
Actores	Anónimo	
Propósito	Obtener un nombre de usuario para acceder al sistema	
Resumen	Cualquier persona puede registrarse en la aplicación	
Precondiciones	Elegir tipo de usuario (Cliente, Empresa, <i>Freelance</i>) y disponer de correo electrónico	
Pos condiciones		
Flujo de eventos	Interacción con el usuario	Obligaciones del sistema
	1. El usuario pulsa el botón Registrarse	2. El sistema redirige al usuario a la página de inicio de sesión/registro
	3. El usuario elige el tipo de usuario y hace clic en el enlace correspondiente	4. El sistema redirige al usuario al formulario de registro
	5. El usuario rellena el formulario y le da al botón de enviar	6. El sistema inserta el nuevo usuario y le envía un email
Extensiones síncronas	Hasta el paso 5 se puede cancelar Si no se rellena algún dato obligatorio, el sistema lo comprobará y mostrará el error	

Caso de uso	Navegar	
Actores	Cualquier usuario	
Propósito	Visitar las diferentes páginas de la web	
Resumen	Cualquier persona que visite la web puede navegar por sus secciones	
Precondiciones	Disponer de navegador web e internet	
Pos condiciones		
Flujo de eventos	Interacción con el usuario	Obligaciones del sistema
	1. El usuario hace clic en cualquiera de los enlaces de la web	2. El sistema redirige al usuario a la zona elegida
Extensiones síncronas		

Caso de uso	Buscar profesionales	
Actores	Cualquier usuario	
Propósito	Listar los profesionales registrados en la web	
Resumen	Cualquier usuario puede ver una lista de los profesionales registrados en la web	
Precondiciones	Disponer de navegador web e internet	
Pos condiciones		
Flujo de eventos	Interacción con el usuario <ol style="list-style-type: none"> 1. El usuario hace clic en el enlace Profesionales 3. El usuario realiza una búsqueda empleando el formulario 	Obligaciones del sistema <ol style="list-style-type: none"> 2. El sistema redirige al usuario a la página de búsqueda de profesionales 4. El sistema muestra los resultados de la búsqueda al usuario
Extensiones síncronas	Hasta el paso 3 se puede cancelar	

Caso de uso	Buscar proyectos	
Actores	Cualquier usuario	
Propósito	Listar los proyectos con búsqueda de colaboradores abierta	
Resumen	Cualquier usuario puede ver una lista de los proyectos	
Precondiciones	Disponer de navegador web e internet	
Pos condiciones		
Flujo de eventos	Interacción con el usuario <ol style="list-style-type: none"> 1. El usuario hace clic en el enlace Proyectos 3. El usuario realiza una búsqueda empleando el formulario 	Obligaciones del sistema <ol style="list-style-type: none"> 2. El sistema redirige al usuario a la página de búsqueda de proyectos 4. El sistema muestra los resultados de la búsqueda al usuario
Extensiones síncronas	Hasta el paso 3 se puede cancelar	

Caso de uso	Identificarse / Acceder a la zona de usuario	
Actores	Cualquier usuario con clave	
Propósito	Acceder a la zona restringida de la web	
Resumen	Cualquier usuario que esté registrado puede intentar acceder a su zona privada mediante su usuario y contraseña	
Precondiciones	Ser un usuario registrado	
Pos condiciones	El usuario accederá a la intranet	
Flujo de eventos	Interacción con el usuario <ol style="list-style-type: none"> 1. El usuario pulsa el enlace Iniciar Sesión 3. El usuario rellena el formulario con los datos y pulsa el botón de entrar 	Obligaciones del sistema <ol style="list-style-type: none"> 2. El sistema redirige al usuario al formulario de inicio de sesión 4. El sistema comprueba que los datos sean correctos y redirige al usuario a su zona privada
Extensiones síncronas	Hasta el paso 3 se puede cancelar Si en el paso 3 se comprueba que algún dato no es correcto el sistema informará del error	

Caso de uso	Valorar	
Actores	Usuario registrado	
Propósito	Poner nota a otro usuario	
Resumen	Cualquier usuario que haya participado en una tarea puede poner nota a su propietario o colaborador	
Precondiciones	Ser usuario registrado y tener valoraciones pendientes	
Pos condiciones	El otro usuario queda valorado	
Flujo de eventos	Interacción con el usuario <ol style="list-style-type: none"> 1. El usuario elige una nota en su página de valoraciones 	Obligaciones del sistema <ol style="list-style-type: none"> 2. El sistema pondrá la nota al usuario
Extensiones síncronas		

Caso de uso	Modificar datos	
Actores	Usuario registrado	
Propósito	Modificar los datos personales	
Resumen	Cualquier usuario registrado puede modificar sus datos	
Precondiciones	Ser usuario registrado	
Pos condiciones	Los datos serán modificados	
Flujo de eventos	Interacción con el usuario <ol style="list-style-type: none"> 1. El usuario rellenará el formulario modificando los datos que quiera cambiar y pulsará el botón de enviar 	Obligaciones del sistema <ol style="list-style-type: none"> 2. El sistema comprobará los datos y modificará los que hayan sido cambiados.
Extensiones síncronas	Hasta el paso 1 se puede cancelar Si en el paso 2 el sistema comprueba que algún campo obligatorio no se ha rellenado, informará del error	

Caso de uso	Crear proyecto	
Actores	Usuario registrado	
Propósito	Publicar un nuevo proyecto en la web	
Resumen	Cualquier usuario registrado puede crear un nuevo proyecto	
Precondiciones	Ser usuario registrado	
Pos condiciones	El proyecto será publicado	
Flujo de eventos	Interacción con el usuario <ol style="list-style-type: none"> 1. El usuario rellena los datos del formulario de nuevo proyecto y hace clic en publicar 	Obligaciones del sistema <ol style="list-style-type: none"> 2. El sistema comprueba los datos y publica el nuevo proyecto
Extensiones síncronas	Hasta el paso 1 se puede cancelar Si en el paso 2 el sistema comprueba que algún campo obligatorio no se ha rellenado, informará del error	

Caso de uso	Acceder a la intranet de proyecto	
Actores	Usuario registrado	
Propósito	Entrar en la página privada de un proyecto	
Resumen	Cualquier usuario propietario o colaborador puede acceder a la intranet de un proyecto	
Precondiciones	Ser usuario registrado y ser propietario o colaborador de un proyecto	
Pos condiciones		
Flujo de eventos	Interacción con el usuario <ol style="list-style-type: none"> 1. El usuario hace clic en el enlace acceder a la Intranet del proyecto en su página de proyectos 	Obligaciones del sistema <ol style="list-style-type: none"> 2. El sistema comprueba los datos y redirige al usuario a la intranet
Extensiones síncronas	Si en el paso 2 el sistema comprueba que el usuario no es propietario o colaborador, le redirigirá a su zona de usuario	

Caso de uso	Inscribirse en tarea				
Actores	Profesional				
Propósito	Aparecer como candidato a una tarea				
Resumen	Cualquier profesional puede inscribirse en tareas para participar en proyectos				
Precondiciones	Ser usuario registrado de tipo profesional				
Pos condiciones	El usuario quedará inscrito en la tarea				
Flujo de eventos	<table border="0"> <tr> <td style="vertical-align: top;">Interacción con el usuario</td> <td style="vertical-align: top;">Obligaciones del sistema</td> </tr> <tr> <td> <ol style="list-style-type: none"> 1. El usuario hace clic en el botón inscribirse de la tarea que quiera </td> <td> <ol style="list-style-type: none"> 2. El sistema comprueba los datos e inscribe al usuario </td> </tr> </table>	Interacción con el usuario	Obligaciones del sistema	<ol style="list-style-type: none"> 1. El usuario hace clic en el botón inscribirse de la tarea que quiera 	<ol style="list-style-type: none"> 2. El sistema comprueba los datos e inscribe al usuario
Interacción con el usuario	Obligaciones del sistema				
<ol style="list-style-type: none"> 1. El usuario hace clic en el botón inscribirse de la tarea que quiera 	<ol style="list-style-type: none"> 2. El sistema comprueba los datos e inscribe al usuario 				
Extensiones síncronas	Si en el paso 2 el sistema comprueba que el usuario no es Profesional le mostrará un error				

Caso de uso	Listar datos				
Actores	Administrador				
Propósito	Obtener una lista de los usuarios, proyectos, tareas, etc.				
Resumen	El administrador puede acceder a cualquier dato de la web				
Precondiciones	Ser administrador y estar en el escritorio de Wordpress				
Pos condiciones					
Flujo de eventos	<table border="0"> <tr> <td style="vertical-align: top;">Interacción con el usuario</td> <td style="vertical-align: top;">Obligaciones del sistema</td> </tr> <tr> <td> <ol style="list-style-type: none"> 1. El administrador hace clic en el enlace correspondiente a los datos que quiera listar en el escritorio de Wordpress </td> <td> <ol style="list-style-type: none"> 2. El sistema le mostrará la lista de los datos </td> </tr> </table>	Interacción con el usuario	Obligaciones del sistema	<ol style="list-style-type: none"> 1. El administrador hace clic en el enlace correspondiente a los datos que quiera listar en el escritorio de Wordpress 	<ol style="list-style-type: none"> 2. El sistema le mostrará la lista de los datos
Interacción con el usuario	Obligaciones del sistema				
<ol style="list-style-type: none"> 1. El administrador hace clic en el enlace correspondiente a los datos que quiera listar en el escritorio de Wordpress 	<ol style="list-style-type: none"> 2. El sistema le mostrará la lista de los datos 				
Extensiones síncronas	Si en el paso 2 se comprueba que el usuario no es Administrador, se le denegará el acceso al escritorio de Wordpress				

Caso de uso	Crear datos	
Actores	Administrador	
Propósito	Crear nuevos usuarios, proyectos, tareas, etc.	
Resumen	El administrador puede crear cualquier dato de la web	
Precondiciones	Ser administrador y estar en el escritorio de Wordpress	
Pos condiciones		
Flujo de eventos	<p>Interacción con el usuario</p> <ol style="list-style-type: none"> 1. El administrador hace clic en el enlace correspondiente a los datos que quiera crear en el escritorio de Wordpress 3. El administrador rellena los datos del formulario y le da al botón de crear 	<p>Obligaciones del sistema</p> <ol style="list-style-type: none"> 2. El sistema redirige al usuario al formulario de creación de los nuevos datos 4. El sistema comprueba los datos y crea al nuevo contenido
Extensiones síncronas	Si en el paso 2 se comprueba que el usuario no es Administrador, se le denegará el acceso al escritorio de Wordpress	

Caso de uso	Administrador	
Actores	Administrador	
Propósito	Modificar usuarios, proyectos, tareas, etc.	
Resumen	El administrador puede modificar cualquier dato de la web	
Precondiciones	Ser administrador y estar en el escritorio de Wordpress	
Pos condiciones		
Flujo de eventos	<p>Interacción con el usuario</p> <ol style="list-style-type: none"> 1. El administrador hace clic en el enlace correspondiente a los datos que quiera modificar en el escritorio de Wordpress 3. El administrador rellena los datos del formulario y le da al botón de crear 	<p>Obligaciones del sistema</p> <ol style="list-style-type: none"> 2. El sistema redirige al usuario al formulario de modificación de los datos 4. El sistema comprueba los datos y modifica el contenido
Extensiones síncronas	Si en el paso 2 se comprueba que el usuario no es Administrador, se le denegará el acceso al escritorio de Wordpress	

Caso de uso		
Actores	Administrador	
Propósito	Eliminar usuarios, proyectos, tareas, etc.	
Resumen	El administrador puede eliminar cualquier dato de la web	
Precondiciones	Ser administrador y estar en el escritorio de Wordpress	
Pos condiciones		
Flujo de eventos	Interacción con el usuario <ol style="list-style-type: none"> 1. El administrador hace clic en el enlace correspondiente a los datos que quiera eliminar en el escritorio de Wordpress 	Obligaciones del sistema <ol style="list-style-type: none"> 2. El sistema elimina el contenido
Extensiones síncronas	Si en el paso 2 se comprueba que el usuario no es Administrador, se le denegará el acceso al escritorio de Wordpress	

7.2 Anexo II: Diseño conceptual de la base de datos

ESTADO_CANDIDATURA (usuario:dom_usuario, tarea:dom_tarea, estado:dom_estado)

CP{ }

CAj: {usuario}→WP_USERS

CAj: {tarea}→WP_POSTS

RECURSOS (id:dom_id, tarea:dom_tarea, nombre:dom_nombre, url:dom_url)

CP{ }

CAj: {tarea}→WP_POSTS

VALORACIONES (emisor:dom_usuario, receptor:dom_usuario, tarea:dom_tarea, nota:dom_nota, fecha:dom_fecha)

CP{ }

CAj: {emisor}→WP_USERS

CAj: {receptor}→WP_USERS

CAj: {tarea}→WP_POSTS

VALORACIONES_MEDIA (usuario:dom_usuario, media:dom_media)

CP{usuario}

CAj: {usuario}→WP_USERS

WP_CIMY_UEF_DATA (id:dom_bigid, user_id:dom_bigid, field_id:dom_bigid, value:dom_value)

CP{id}

CAj: {user_id}→WP_USERS

CAj: {field_id}→WP_CIMY_UEF_DATA

WP_CIMY_UEF_FIELDS (id:dom_bigid, f_order:dom_f_order,

fieldset:dom_fieldset, name:dom_name, label:dom_label,

description:dom_description, type:dom_type, rules:dom_rules, value:dom_value)

CP{id}

WP_CIMY_UEF_WP_FIELDS (id:dom_bigid, f_order:dom_f_order,

name:dom_name, label:dom_label, description:dom_description, type:dom_type,

rules:dom_rules, value:dom_value)

CP{id}

WP_COMMENTMETA (meta_id:dom_bigid, comment_id:dom_bigid,

meta_key:dom_meta_key, meta_value:dom_meta_value)

CP{meta_id}

CAj: {comment_id}→WP_COMMENTS

WP_COMMENTS (comment_id:dom_bigid, comment_post_id:dom_bigid,
comment_author:dom_comment_author,
comment_author_email:dom_comment_author_email,
comment_author_url:dom_comment_author_url,
comment_author_ip:dom_comment_author_ip, comment_date:dom_date,
comment_date_gmt:dom_date, comment_content:dom_comment_content,
comment_karma:dom_comment_karma,
comment_approved:dom_comment_approved,
comment_agent:dom_comment_agent, comment_type:dom_type,
comment_parent:dom_bigid, user_id:dom_bigid)
CP{comment_id}
CAj: {comment_post_id}→WP_POSTS
CAj: {comment_parent}→WP_COMMENTS
CAj: {user_id}→WP_USERS

WP_OPTIONS (option_id:dom_bigid, option_name:dom_option_name,
option_value:dom_option_value, autoload:dom_autoload)
CP{option_id}

WP_LINKS (link_id:dom_bigid, link_url:dom_link_url, link_name:dom_link_name,
link_image:dom_link_image, link_target:dom_link_target,
link_description:dom_link_description, link_visible:dom_link_visible,
link_owner:dom_bigid, link_rating:dom_link_rating, link_updated:dom_date,
link_rel:dom_link_rel, link_notes:dom_link_notes, link_rss:dom_link_rss)
CP{link_id}

WP_POSTMETA (meta_id:dom_bigid, post_id:dom_bigid,
meta_key:dom_meta_key, meta_value:dom_meta_value)
CP{meta_id}
CAj: {post_id}→WP_POSTS

WP_POSTS (id:dom_bigid, post_author:dom_bigid, post_date:dom_date,
post_date_gmt:dom_date, post_content:dom_post_content,
post_title:dom_post_title, post_excerpt:dom_post_excerpt,
post_status:dom_post_status, post_password:dom_post_password,
post_name:dom_post_name, to_ping:dom_to_ping, pinged:dom_to_pinged,
post_modified:dom_date, post_content_filtered:dom_post_content,
post_parent:dom_bigid, guid:dom_guid, mean_order:dom_mean_order,
post_type:dom_post_type, post_mime_type:dom_post_mime_type,
comment_count:dom_comment_count)
CP{id}
CAj: {post_author}→WP_USERS
CAj: {post_parent}→WP_POSTS

WP_TERMS (term_id:dom_bigid, name:dom_term_name, slug:dom_slug,
term_group:dom_term_group)
CP{term_id}



WP_TERMS_RELATIONSHIPS (object_id:dom_bigid,
term_taxonomy_id:dom_bigid, term_order:dom_term_order)
CP{object_id, term_taxonomy_id}
CAj: {object_id}→WP_POSTS
CAj: {term_taxonomy_id}→WP_TERM_TAXONOMY

WP_TERM_TAXONOMY (term_taxonomy_id:dom_bigid, term_id:dom_bigid,
taxonomy:dom_taxonomy, description:dom_description, parent:dom_bigid,
count:dom_count)
CP{term_taxonomy_id}
CAj: {term_id}→WP_TERMS
CAj: {parent}→WP_TERMS

WP_USERMETA (umeta_id:dom_bigid, user_id:dom_bigid,
meta_key:dom_meta_key, meta_value:dom_meta_value)
CP{umeta_id}
CAj: {user_id}→WP_USERS

WP_USERS (id:dom_bigid, user_login:dom_user_login, user_pass:dom_user_pass,
user_nicename:dom_user_nicename, user_email:dom_user_email,
user_url:dom_user_url, user_registered:dom_date,
user_activation_key:dom_user_activation_key, user_status:dom_user_status,
display_name:dom_display_name)
CP{id}

7.3 Anexo III: Dominios base de datos

Nombre dominio	Tipo de dato
dom_usuario	Int(11)
dom_tarea	Int(11)
dom_estado	Varchar(20)
dom_id	Int(11)
dom_nombre	Varchar(500)
dom_url	Varchar(1000)
dom_nota	Int(2)
dom_fecha	date
dom_bigid	Bigint(20)
dom_value	text
dom_f_order	Bigint(20)
dom_fieldset	Bigint(20)
dom_name	Varchar(20)
dom_label	text
dom_description	text
dom_type	Varchar(20)
dom_rules	text
dom_meta_key	Varchar(255)
dom_meta_value	longtext
dom_comment_author	tinytext
dom_comment_author_email	Varchar(100)
dom_comment_author_url	Varchar(200)
dom_comment_author_ip	Varchar(100)
dom_date	datetime
dom_comment_content	text
dom_comment_karma	Int(11)
dom_comment_approved	Varchar(20)
dom_comment_agent	Varchar(255)
dom_option_name	Varchar(64)
dom_option_value	longtext
dom_autoload	Varchar(20)
dom_link_url	Varchar(255)
dom_link_name	Varchar(255)
dom_link_image	Varchar(255)
dom_link_target	Varchar(25)
dom_link_description	Varchar(255)
dom_link_visible	Varchar(20)
dom_link_rating	Int(11)
dom_link_rel	Varchar(255)
dom_link_notes	mediumtext
dom_link_rss	Varchar(255)
dom_post_content	longtext
dom_post_title	text
dom_post_excerpt	text
dom_post_status	Varchar(20)

dom_post_password	Varchar(20)
dom_post_name	Varchar(200)
dom_to_ping	text
dom_to_pinged	text
dom_mean_order	Int(11)
dom_guid	Varchar(255)
dom_post_type	Varchar(20)
dom_post_mime_type	Varchar(100)
dom_comment_count	Bigint(20)
dom_term_name	Varchar(200)
dom_slug	Varchar(200)
dom_term_group	Bigint(10)
dom_term_order	Int(11)
dom_taxonomy	Varchar(32)
dom_count	Bigint(20)
dom_user_login	Varchar(60)
dom_user_pass	Varchar(64)
dom_user_nicename	Varchar(50)
dom_user_email	Varchar(100)
dom_user_url	Varchar(100)
dom_user_status	Int(11)
dom_display_name	Varchar(250)

7.4 Anexo IV: Fichero de creación de la base de datos

```
CREATE TABLE estado_candidatura (  
  usuario int(11) NOT NULL,  
  tarea int(11) NOT NULL,  
  estado varchar(20) NOT NULL DEFAULT 'Presentada'  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

```
CREATE TABLE recursos (  
  id int(11) NOT NULL,  
  tarea int(11) NOT NULL,  
  nombre varchar(500) NOT NULL,  
  url varchar(1000) NOT NULL  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

```
CREATE TABLE valoraciones (  
  emisor int(11) NOT NULL,  
  receptor int(11) NOT NULL,  
  tarea int(11) NOT NULL,  
  nota int(2) DEFAULT NULL,  
  fecha date NOT NULL  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

```
CREATE TABLE valoraciones_media (  
  usuario int(11) NOT NULL,  
  media int(6) NOT NULL,  
  PRIMARY KEY (usuario)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

```
CREATE TABLE wp_cimy_uf_data (  
  ID bigint(20) NOT NULL AUTO_INCREMENT,  
  USER_ID bigint(20) NOT NULL,  
  FIELD_ID bigint(20) NOT NULL,  
  `VALUE` text NOT NULL,  
  PRIMARY KEY (ID),  
  KEY USER_ID (USER_ID),  
  KEY FIELD_ID (FIELD_ID)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```



```
CREATE TABLE wp_cimy_uf_fields (
  ID bigint(20) NOT NULL AUTO_INCREMENT,
  F_ORDER bigint(20) NOT NULL,
  FIELDSET bigint(20) NOT NULL DEFAULT '0',
  `NAME` varchar(20) DEFAULT NULL,
  LABEL text,
  DESCRIPTION text,
  `TYPE` varchar(20) DEFAULT NULL,
  RULES text,
  `VALUE` text,
  PRIMARY KEY (ID),
  KEY F_ORDER (F_ORDER),
  KEY `NAME` (`NAME`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE wp_cimy_uf_wp_fields (
  ID bigint(20) NOT NULL AUTO_INCREMENT,
  F_ORDER bigint(20) NOT NULL,
  `NAME` varchar(20) DEFAULT NULL,
  LABEL text,
  DESCRIPTION text,
  `TYPE` varchar(20) DEFAULT NULL,
  RULES text,
  `VALUE` text,
  PRIMARY KEY (ID),
  KEY F_ORDER (F_ORDER),
  KEY `NAME` (`NAME`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE wp_commentmeta (
  meta_id bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  comment_id bigint(20) unsigned NOT NULL DEFAULT '0',
  meta_key varchar(255) DEFAULT NULL,
  meta_value longtext,
  PRIMARY KEY (meta_id),
  KEY comment_id (comment_id),
  KEY meta_key (meta_key)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

```

CREATE TABLE wp_comments (
  comment_ID bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  comment_post_ID bigint(20) unsigned NOT NULL DEFAULT '0',
  comment_author tinytext NOT NULL,
  comment_author_email varchar(100) NOT NULL DEFAULT '',
  comment_author_url varchar(200) NOT NULL DEFAULT '',
  comment_author_IP varchar(100) NOT NULL DEFAULT '',
  comment_date datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  comment_date_gmt datetime NOT NULL DEFAULT '0000-00-00
00:00:00',
  comment_content text NOT NULL,
  comment_karma int(11) NOT NULL DEFAULT '0',
  comment_approved varchar(20) NOT NULL DEFAULT '1',
  comment_agent varchar(255) NOT NULL DEFAULT '',
  comment_type varchar(20) NOT NULL DEFAULT '',
  comment_parent bigint(20) unsigned NOT NULL DEFAULT '0',
  user_id bigint(20) unsigned NOT NULL DEFAULT '0',
  PRIMARY KEY (comment_ID),
  KEY comment_post_ID (comment_post_ID),
  KEY comment_approved_date_gmt
(comment_approved,comment_date_gmt),
  KEY comment_date_gmt (comment_date_gmt),
  KEY comment_parent (comment_parent)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```

```

CREATE TABLE wp_links (
  link_id bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  link_url varchar(255) NOT NULL DEFAULT '',
  link_name varchar(255) NOT NULL DEFAULT '',
  link_image varchar(255) NOT NULL DEFAULT '',
  link_target varchar(25) NOT NULL DEFAULT '',
  link_description varchar(255) NOT NULL DEFAULT '',
  link_visible varchar(20) NOT NULL DEFAULT 'Y',
  link_owner bigint(20) unsigned NOT NULL DEFAULT '1',
  link_rating int(11) NOT NULL DEFAULT '0',
  link_updated datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  link_rel varchar(255) NOT NULL DEFAULT '',
  link_notes mediumtext NOT NULL,
  link_rss varchar(255) NOT NULL DEFAULT '',
  PRIMARY KEY (link_id),
  KEY link_visible (link_visible)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```

```

CREATE TABLE wp_options (
  option_id bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  option_name varchar(64) NOT NULL DEFAULT '',
  option_value longtext NOT NULL,
  autoload varchar(20) NOT NULL DEFAULT 'yes',
  PRIMARY KEY (option_id),
  UNIQUE KEY option_name (option_name)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```



```

CREATE TABLE wp_postmeta (
  meta_id bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  post_id bigint(20) unsigned NOT NULL DEFAULT '0',
  meta_key varchar(255) DEFAULT NULL,
  meta_value longtext,
  PRIMARY KEY (meta_id),
  KEY post_id (post_id),
  KEY meta_key (meta_key)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE wp_posts (
  ID bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  post_author bigint(20) unsigned NOT NULL DEFAULT '0',
  post_date datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  post_date_gmt datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  post_content longtext NOT NULL,
  post_title text NOT NULL,
  post_excerpt text NOT NULL,
  post_status varchar(20) NOT NULL DEFAULT 'publish',
  comment_status varchar(20) NOT NULL DEFAULT 'open',
  ping_status varchar(20) NOT NULL DEFAULT 'open',
  post_password varchar(20) NOT NULL DEFAULT '',
  post_name varchar(200) NOT NULL DEFAULT '',
  to_ping text NOT NULL,
  pinged text NOT NULL,
  post_modified datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  post_modified_gmt datetime NOT NULL DEFAULT '0000-00-00
00:00:00',
  post_content_filtered longtext NOT NULL,
  post_parent bigint(20) unsigned NOT NULL DEFAULT '0',
  guid varchar(255) NOT NULL DEFAULT '',
  menu_order int(11) NOT NULL DEFAULT '0',
  post_type varchar(20) NOT NULL DEFAULT 'post',
  post_mime_type varchar(100) NOT NULL DEFAULT '',
  comment_count bigint(20) NOT NULL DEFAULT '0',
  PRIMARY KEY (ID),
  KEY post_name (post_name),
  KEY type_status_date (post_type,post_status,post_date,ID),
  KEY post_parent (post_parent),
  KEY post_author (post_author)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE wp_terms (
  term_id bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(200) NOT NULL DEFAULT '',
  slug varchar(200) NOT NULL DEFAULT '',
  term_group bigint(10) NOT NULL DEFAULT '0',
  PRIMARY KEY (term_id),
  UNIQUE KEY slug (slug),
  KEY `name` (`name`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```

```

CREATE TABLE wp_term_relationships (
  object_id bigint(20) unsigned NOT NULL DEFAULT '0',
  term_taxonomy_id bigint(20) unsigned NOT NULL DEFAULT '0',
  term_order int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (object_id,term_taxonomy_id),
  KEY term_taxonomy_id (term_taxonomy_id)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE wp_term_taxonomy (
  term_taxonomy_id bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  term_id bigint(20) unsigned NOT NULL DEFAULT '0',
  taxonomy varchar(32) NOT NULL DEFAULT '',
  description longtext NOT NULL,
  parent bigint(20) unsigned NOT NULL DEFAULT '0',
  count bigint(20) NOT NULL DEFAULT '0',
  PRIMARY KEY (term_taxonomy_id),
  UNIQUE KEY term_id_taxonomy (term_id,taxonomy),
  KEY taxonomy (taxonomy)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE wp_usermeta (
  umeta_id bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  user_id bigint(20) unsigned NOT NULL DEFAULT '0',
  meta_key varchar(255) DEFAULT NULL,
  meta_value longtext,
  PRIMARY KEY (umeta_id),
  KEY user_id (user_id),
  KEY meta_key (meta_key)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE wp_users (
  ID bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  user_login varchar(60) NOT NULL DEFAULT '',
  user_pass varchar(64) NOT NULL DEFAULT '',
  user_nicename varchar(50) NOT NULL DEFAULT '',
  user_email varchar(100) NOT NULL DEFAULT '',
  user_url varchar(100) NOT NULL DEFAULT '',
  user_registered datetime NOT NULL DEFAULT '0000-00-00
00:00:00',
  user_activation_key varchar(60) NOT NULL DEFAULT '',
  user_status int(11) NOT NULL DEFAULT '0',
  display_name varchar(250) NOT NULL DEFAULT '',
  PRIMARY KEY (ID),
  KEY user_login_key (user_login),
  KEY user_nicename (user_nicename)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```



7.5 Anexo V: Manual de usuario

Se puede acceder a la web del proyecto introduciendo su url en cualquier navegador. De este modo se muestra la página principal donde tendremos los enlaces principales a las demás secciones de la web.

7.5.1 Registro

Sin tener una sesión iniciada, hacemos clic en el enlace Registrarse del menú de cabecera.



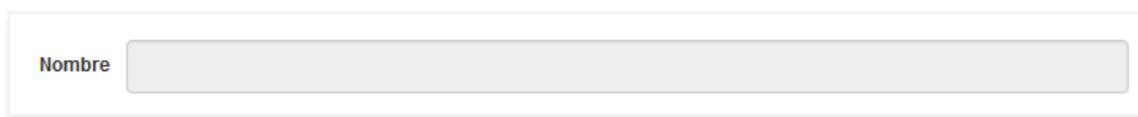
Ilustración 7-1 - Manual de usuario: Registro

Se nos redirige a la página de inicio de sesión/registro, donde tendremos que escoger entre registrarse como cliente, empresa o *freelance*.



Ilustración 7-2 - Manual de usuario: Tipos de registro

Haciendo clic en uno de los enlaces se nos redirige al formulario de registro, con los datos necesarios según el tipo escogido. Los campos obligatorios aparecen resaltados en gris.

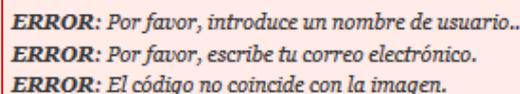


Nombre

Ilustración 7-3 - Manual de usuario: Campos obligatorios

Una vez rellenados todos los datos necesarios y los opcionales que queramos, le damos al botón Registrarse.

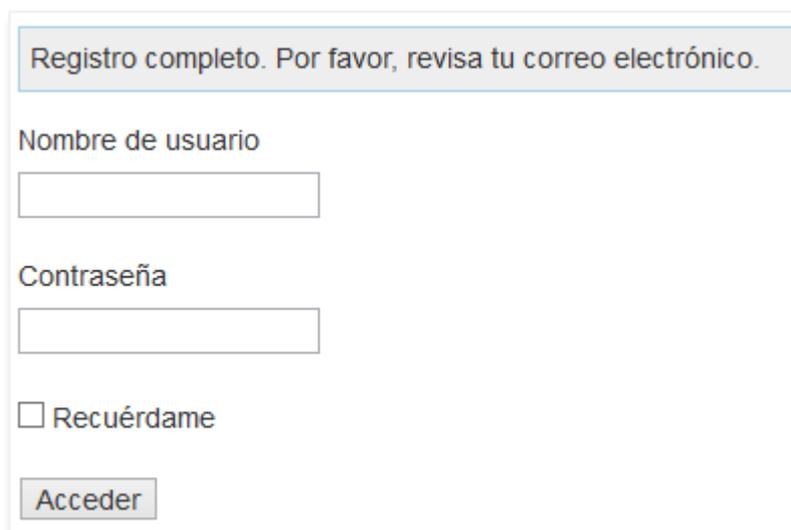
El sistema comprueba los datos antes de registrarnos. Si hay algún error, se nos muestra por pantalla.



ERROR: Por favor, introduce un nombre de usuario..
ERROR: Por favor, escribe tu correo electrónico.
ERROR: El código no coincide con la imagen.

Ilustración 7-4 - Manual de usuario: Errores

Si se completa el registro de forma correcta, nos redirige a la página de inicio de sesión.



Registro completo. Por favor, revisa tu correo electrónico.

Nombre de usuario

Contraseña

Recuérdame

Acceder

Ilustración 7-5 - Manual de usuario: Registro completo

7.5.2 Olvido de contraseña

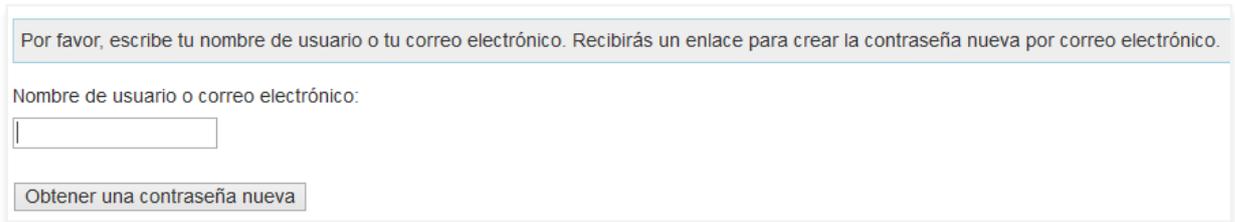
Si se nos pierde o se nos olvida la contraseña de inicio de sesión, tenemos la opción de cambiarla mediante el enlace de la página de inicio de sesión.



¿No recuerdas tu contraseña?

Ilustración 7-6 - Manual de usuario: Olvido de contraseña

Haciendo clic en el enlace nos redirige al formulario que nos permite obtener una nueva contraseña.



Por favor, escribe tu nombre de usuario o tu correo electrónico. Recibirás un enlace para crear la contraseña nueva por correo electrónico.

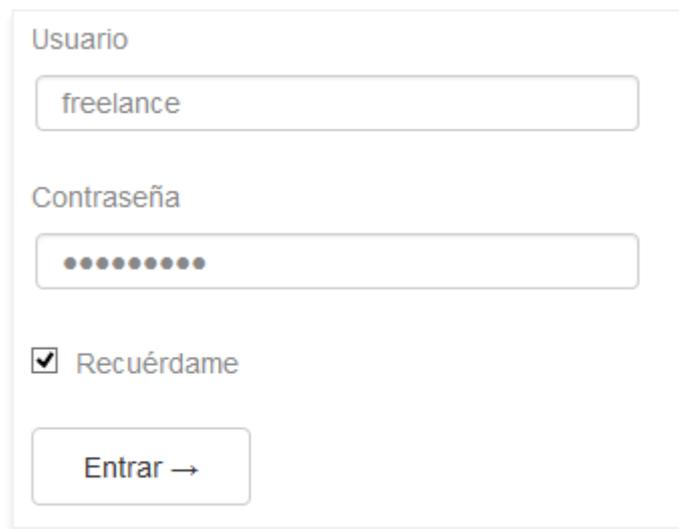
Nombre de usuario o correo electrónico:

Obtener una contraseña nueva

Ilustración 7-7 - Manual de usuario: Resetear contraseña

7.5.3 Iniciar sesión

Para iniciar sesión en la web iremos a la página de inicio de sesión e introduciremos nuestro usuario y contraseña.



Usuario

Contraseña

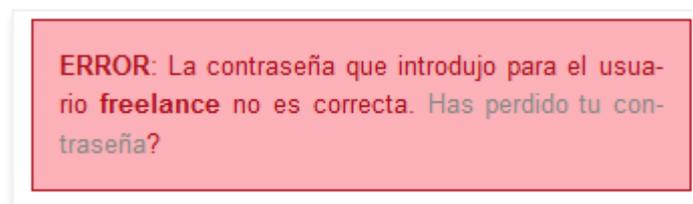
Recuérdame

Entrar →

Ilustración 7-8 - Manual de usuario: Formulario de inicio de sesión

Si los datos son correctos se nos redirige a nuestra zona de usuario.

Si no son correctos se nos muestra un error.



ERROR: La contraseña que introdujo para el usuario **freelance** no es correcta. Has perdido tu contraseña?

Ilustración 7-9 - Manual de usuario: Error de contraseña

7.5.4 Editar perfil

Para editar el perfil de usuario podemos hacer clic en el enlace de los atajos de nuestra zona de usuario o en Perfil, también en la zona de usuario.

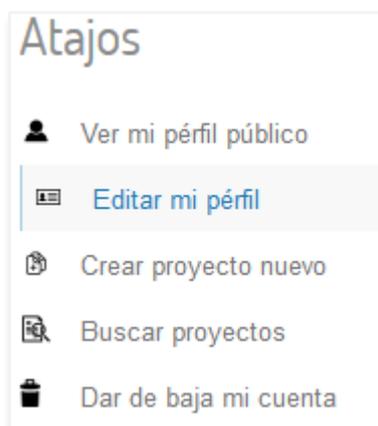


Ilustración 7-10 - Manual de usuario: Atajo a edición de perfil



Ilustración 7-11 - Manual de usuario: Perfil

Para modificar los datos basta con cambiar en el formulario aquellos que queramos y hacer clic en Actualizar perfil.

7.5.5 Página pública de usuario

Todos los usuarios registrados disponen de un perfil público, donde se pueden ver sus datos profesionales y los proyectos en los que colabora.

Se puede acceder a estos perfiles desde la lista de usuarios mejor valorados de la página principal, desde la búsqueda de profesionales o cada usuario desde su menú de atajos.



Ilustración 7-12 - Manual de usuario: Profesionales mejor valorador

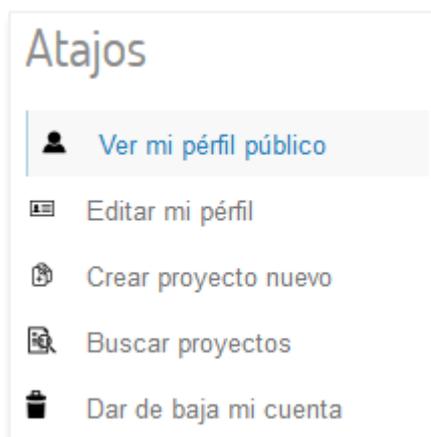


Ilustración 7-13 - Manual de usuario: Atajo a perfil público

7.5.6 Búsqueda de profesionales

Desde el enlace Profesionales del menú de la web o desde el enlace Ver todos los profesionales de la página principal, se puede acceder a un formulario de búsqueda de profesionales registrados en la web. En esta búsqueda los usuarios de tipo Administrador o Cliente no salen.



Ilustración 7-14 - Manual de usuario: Profesionales



Ilustración 7-15 - Manual de usuario: Freelance o Empresa

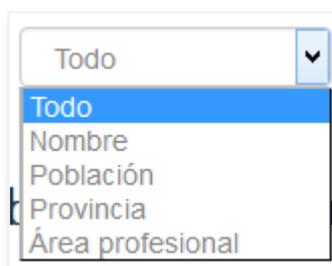


Ilustración 7-16 - Manual de usuario: Opciones de búsqueda de profesionales

El buscador distingue entre Empresas y *Freelance*, y tiene cinco opciones de búsqueda: todo, nombre, población, provincia y área profesional.

Una vez rellenados los campos, la búsqueda muestra los resultados en bloques de diez profesionales.

7.5.7 Crear proyecto nuevo

Cualquier usuario registrado puede crear nuevos proyectos desde su zona de usuario, haciendo clic en el enlace de los atajos.

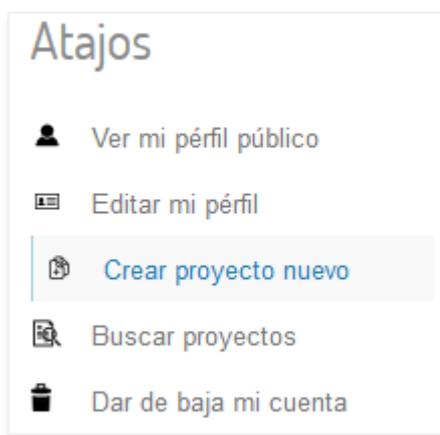


Ilustración 7-17 - Manual de usuario: Crear proyecto nuevo

En el formulario que se abre, aparecen resaltados en gris los campos obligatorios. El proyecto debe tener como mínimo una tarea, pero pueden crearse más mediante el botón de Añadir nueva tarea, o eliminarlas mediante el botón Eliminar tarea de la tarea correspondiente.

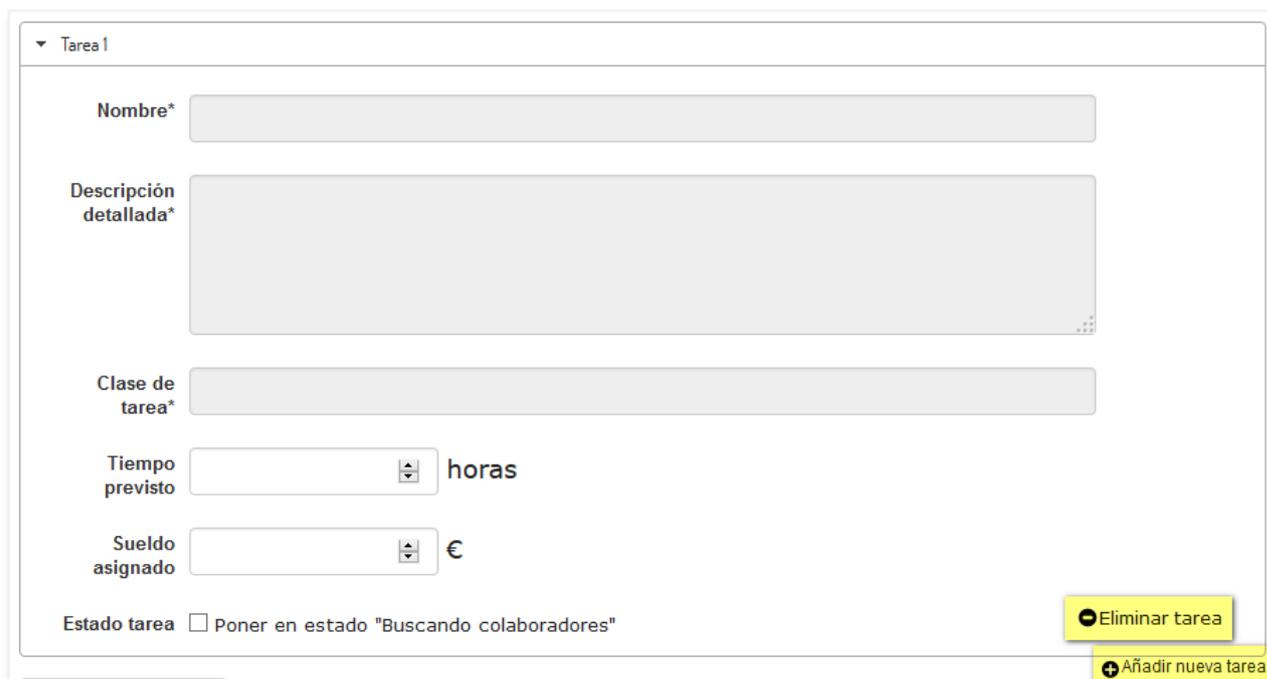
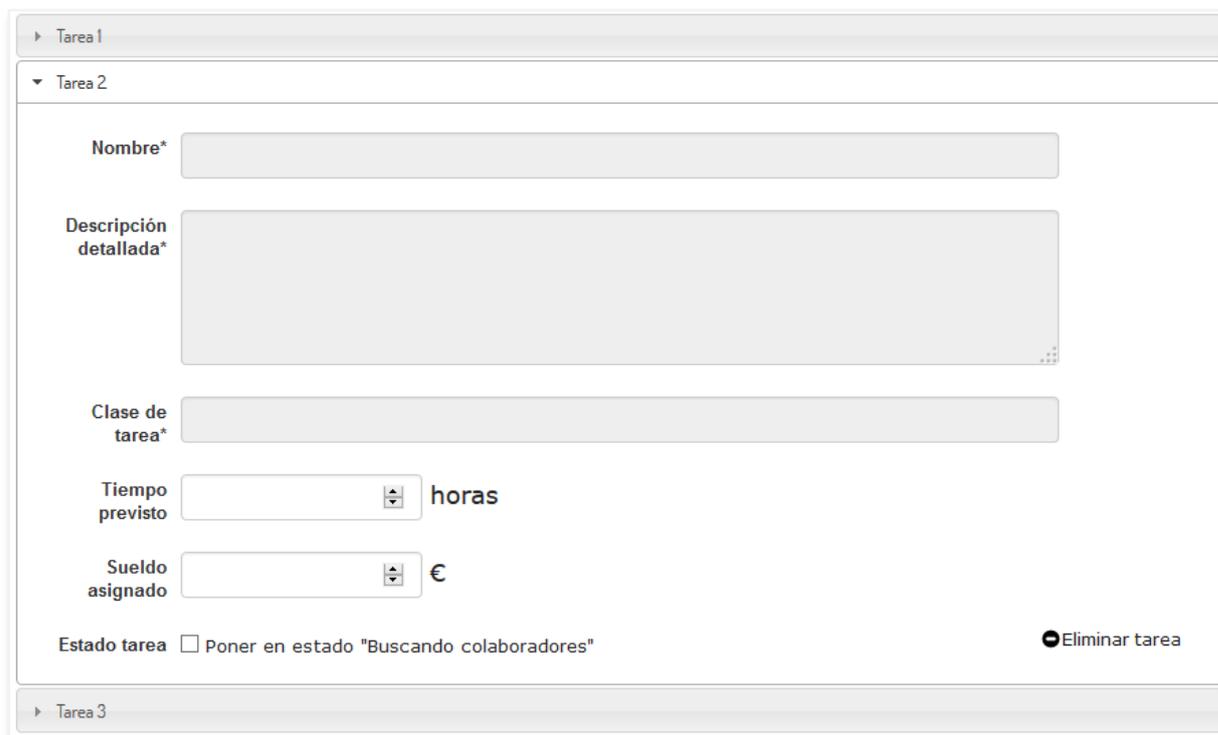
A screenshot of a 'Nueva tarea' (New task) form. The form is titled 'Tarea 1' and contains several fields: 'Nombre*' (required), 'Descripción detallada*' (required), 'Clase de tarea*' (required), 'Tiempo previsto' (with a dropdown arrow and 'horas' label), and 'Sueldo asignado' (with a dropdown arrow and '€' label). At the bottom, there is a checkbox for 'Estado tarea' with the option 'Poner en estado "Buscando colaboradores"'. On the right side, there are two buttons: 'Eliminar tarea' (with a minus sign) and 'Añadir nueva tarea' (with a plus sign).

Ilustración 7-18 - Manual de usuario: Nueva tarea

Una vez creadas varias tareas, para pasar de una a otra solo es necesario hacer clic en la tarea que queramos ver.



The screenshot shows a web interface for task management. At the top, there are three tabs: 'Tarea 1', 'Tarea 2', and 'Tarea 3'. 'Tarea 2' is the active tab. The form for 'Tarea 2' contains the following fields and controls:

- Nombre***: A text input field.
- Descripción detallada***: A large text area for detailed description.
- Clase de tarea***: A text input field.
- Tiempo previsto**: A numeric input field followed by a dropdown menu and the unit 'horas'.
- Sueldo asignado**: A numeric input field followed by a dropdown menu and the unit '€'.
- Estado tarea**: A checkbox labeled 'Poner en estado "Buscando colaboradores"'. To its right is a button with a trash icon labeled 'Eliminar tarea'.

Ilustración 7-19 - Manual de usuario: Tareas

Si hay otros proyectos creados en la web, al escribir en el campo de Tipo de proyecto, se autocompleta con los tipos de proyectos ya creados.



The screenshot shows a dropdown menu for the 'Tipo de proyecto*' field. The dropdown is open, showing a list of project types:

- App
- Jefe de proyecto
- Profesor
- PROFESOR DE GIN TONIC
- Programación
- Programar web

Ilustración 7-20 - Manual de usuario: Tipo de proyecto

Poner una tarea en estado “Buscando colaboradores” hace que el proyecto aparezca en la búsqueda de proyectos y que los profesionales puedan inscribirse como candidatos.

Estado tarea Poner en estado "Buscando colaboradores"

Ilustración 7-21 - Manual de usuario: Poner en estado Buscando colaboradores

Al pulsar en el botón de publicar, el proyecto se publicará si no hay errores.

7.5.8 Buscar proyectos

Para ir a la búsqueda de proyectos hay que hacer clic en Buscar proyectos en la zona de usuario, en Proyectos en el menú superior de la web o haciendo clic en Ver todos los proyectos con búsqueda de colaboradores abierta, en la zona Últimos proyectos de la página principal.

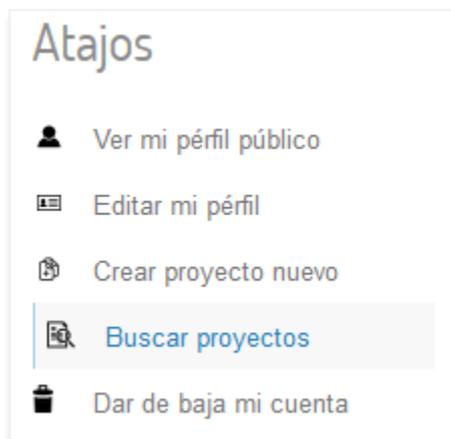


Ilustración 7-22 - Manual de usuario: Atajo a buscar proyectos



Ilustración 7-23 - Manual de usuario: Menú a Proyectos



Ilustración 7-24 - Manual de usuario: Ver todos los proyectos

En la nueva página aparecerá una lista con los proyectos con fase de búsqueda de candidatos abierta, ordenados por fecha de creación descendente.

Hay tres opciones de búsqueda para los proyectos: por nombre del proyecto, por tipo de proyecto y por localización.

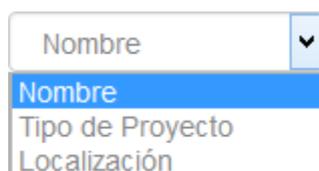


Ilustración 7-25 - Manual de usuario: Opciones de búsqueda de proyectos

Una vez seleccionada una de las opciones y con una o varias palabras en el campo de búsqueda, el sistema devolverá los resultados si los hubiera.



Ilustración 7-26 - Manual de usuario: Resultados de la búsqueda

7.5.9 Página pública de proyecto

La página pública de un proyecto muestra todos los datos del proyecto y permite inscribirse en las tareas.

Se puede acceder a la página de un proyecto de diferentes modos: Desde la página principal, si el proyecto se encuentra en la lista de últimos proyectos, realizando la búsqueda del proyecto o desde los proyectos del usuario en la zona de usuario.

7.5.9.1 Inscripción en una tarea

Una vez dentro de la página de un proyecto, si el usuario ha iniciado sesión y es Profesional, se le permite inscribirse en las tareas como candidato.

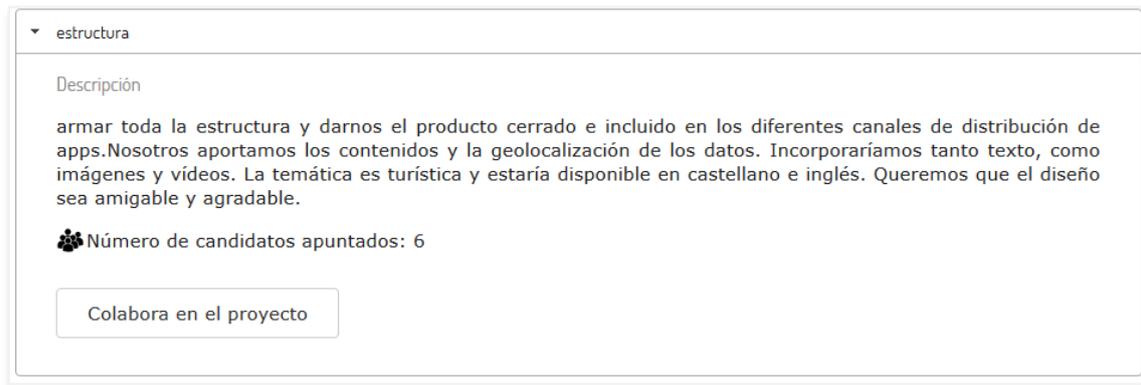


Ilustración 7-27 - Manual de usuario: Colaborar en tarea

Si el usuario no ha iniciado sesión, se le pide iniciar sesión o registrarse para poder inscribirse en las tareas.

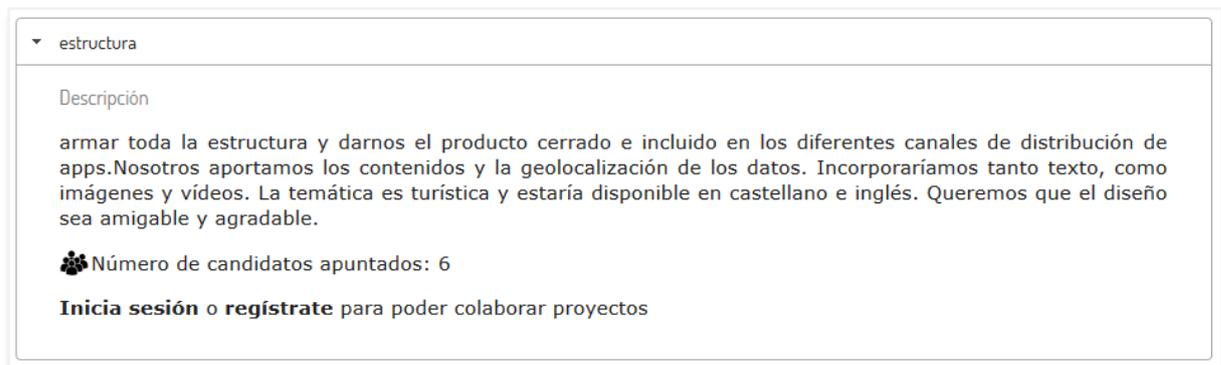


Ilustración 7-28 - Manual de usuario: Inscripción en tareas

Si el usuario se inscribe en la tarea y no hay errores, se le comunicará inmediatamente.

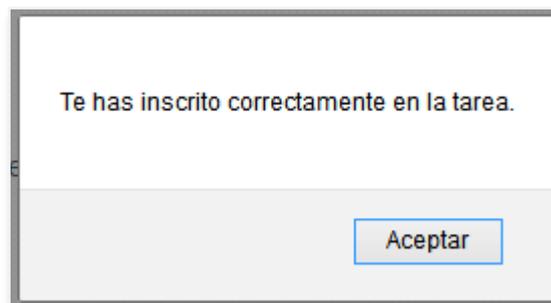


Ilustración 7-29 - Manual de usuario: Inscripción correcta

7.5.10 Proyectos del usuario

Cuando el usuario haya creado proyectos, se haya inscrito en tareas o sea colaborador de alguna de ellas, estos aparecerán en su zona de usuario. Se puede acceder a los proyectos haciendo clic en Mis proyectos, en la zona de usuario.



Ilustración 7-30 - Manual de usuario: Menú a mis proyectos

Para los proyectos creados por el usuario o en los que sea colaborador, aparece un enlace para acceder a la intranet del proyecto.



Ilustración 7-31 - Manual de usuario: Acceder a la intranet del proyecto

En los proyectos en los que esté inscrito como candidato, se puede ver el estado de la candidatura.



Ilustración 7-32 - Manual de usuario: Estado de candidatura

7.5.11 Intranet de proyecto

En la intranet de un proyecto se puede ver el progreso del proyecto, según las tareas completadas, el estado de sus tareas y los colaboradores asignados.

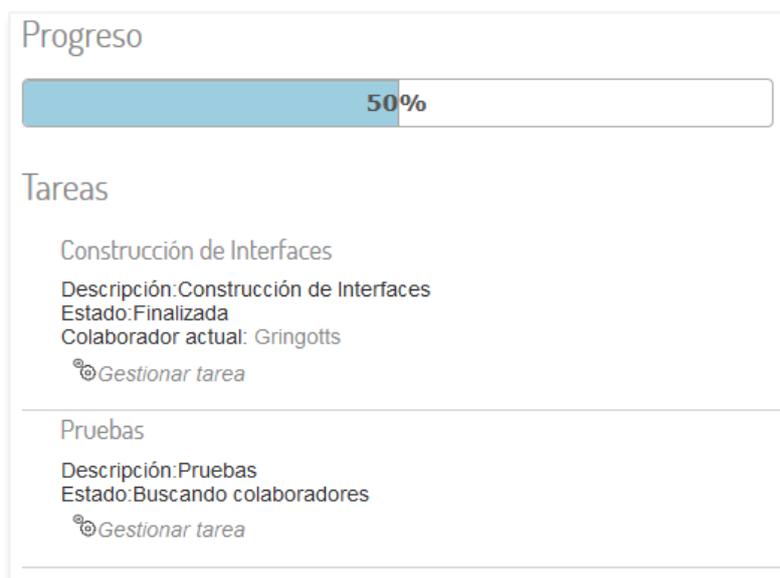


Ilustración 7-33 - Manual de usuario: Proyecto

Si el usuario es el propietario del proyecto aparecen los enlaces Gestionar tarea, con los que accederá a la intranet de las tareas.

Si el usuario es colaborador de alguna tarea, aparece el enlace Accede a la tarea.

integrar un diseño web desde psds a wordpress

Descripción: Entregamos diseño y necesitamos integrarlo al sistema wordpress. Estamos abiertos a la compra de alguna plantilla para acoplar el diseño que hemos creado. El wordpress deberá llevar el plugin de woo commerce que facilitaremos con su correspondiente configuración y mails de autoresponse. En total unas 20 páginas aproximadamente en esta línea.

Estado: En proceso

Colaborador actual: Corporación ACME

[Acceder a la tarea](#)

Ilustración 7-34 - Manual de usuario: Acceder a la tarea

Cualquiera de los usuarios autorizados en un proyecto, puede enviar mensajes tanto en la intranet del proyecto como en la de la tarea que le corresponda. Los mensajes se enviarán y cargarán automáticamente en la página una vez que se envíen.

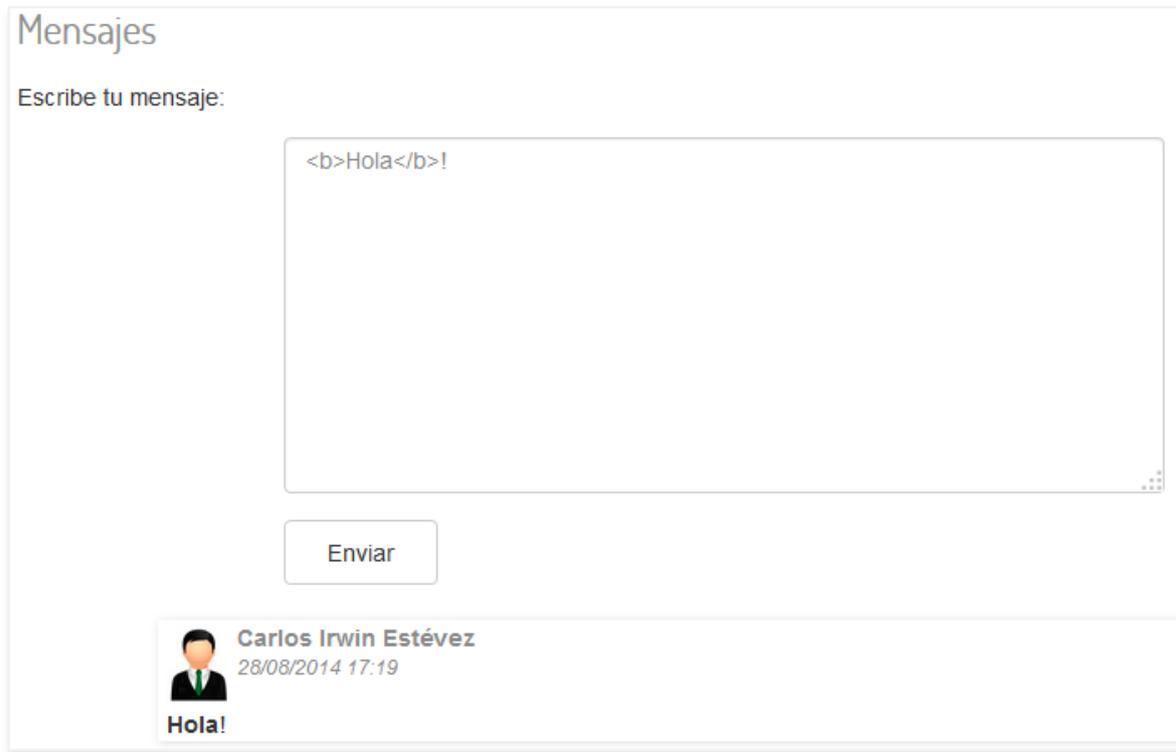


Ilustración 7-35 - Manual de usuario: Mensajes

7.5.12 Intranet de tarea

A la intranet de la tarea puede acceder siempre el propietario del proyecto al que pertenezca la tarea. Si la tarea está en proceso, también podrá acceder el colaborador asignado.

7.5.12.1 Acceso como colaborador

El colaborador de la tarea puede acceder a la tarea para comunicarse con el propietario de la misma y ver o mandar archivos adjuntos, mediante el botón subir archivo



Ilustración 7-36 - Manual de usuario: Subir archivos

7.5.12.2 Acceso como propietario

El propietario además de mensajes y archivos, puede gestionar el estado de la tarea y cambiarlo según sea necesario.

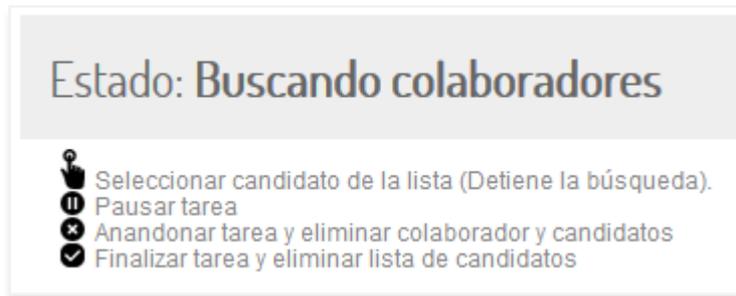


Ilustración 7-37 - Manual de usuario: Estados de la tarea

Cambiar de estado recargará la página de la tarea.

El propietario también puede gestionar la lista de candidatos inscritos a la tarea asignándolos, descartándolos o pasándolos a finalistas.

Candidato	Estado candidatura	Acciones
Umbrella Corporation	Descartada	Asignar Hacer finalista Descartar
Gringotts	Presentada	

Ilustración 7-38 - Manual de usuario: Candidatos de la tarea

Si Asigna a un candidato como colaborador, la tarea pasa a estar en Proceso. Si descarta al actual colaborador, la tarea pasa a modo Pausada.

7.5.13 Valorar a un usuario

Cuando una tarea se finaliza, abandona o se descarta a su actual colaborador, tanto el propietario como el colaborador deben valorarse mutuamente.

Las valoraciones de cada usuario serán accesibles desde el menú de usuario en la zona de usuario.



Ilustración 7-39 - Manual de usuario: Acceso a valoraciones

Si el usuario tiene alguna valoración pendiente le aparece un número al lado de la opción del menú y al entrar en las valoraciones estas aparecerán las primeras.

Valoraciones pendientes				
Colaborador	Proyecto	Tarea valorada	Puntuación (0-10)	
Umbrella Corporation	ANALISTA SAP XI PI	Pruebas	5	Valorar

Ilustración 7-40 - Manual de usuario: Valoraciones pendientes

Si otro usuario ya nos ha valorado antes, las valoraciones aparecen debajo de las pendientes.

Tus valoraciones				
Valoración emitida por	Proyecto	Tarea valorada	Fecha	Puntuación (0-10)
Gringotts	ANALISTA SAP XI PI	Construcción de Interfaces	23-07-2014	9

Ilustración 7-41 - Manual de usuario: Valoraciones

7.6 Anexo VI: Índice de ilustraciones

Ilustración 3-1 – Diagrama de clases	15
Ilustración 3-2 - Diagrama de casos de uso	18
Ilustración 3-3 - Caso de uso: Anónimo	19
Ilustración 3-4 - Caso de uso: Usuario registrado.....	20
Ilustración 3-5 - Caso de uso: Profesional.....	20
Ilustración 3-6 - Caso de uso: Administrador	21
Ilustración 4-1 - Niveles de arquitectura	22
Ilustración 4-2 - Interfaz gráfica: Pantalla principal	23
Ilustración 4-3- Interfaz gráfica: Pantalla principal	24
Ilustración 4-4- Interfaz gráfica: Proyectos	25
Ilustración 4-5- Interfaz gráfica: Profesionales.....	26
Ilustración 4-6 - Interfaz gráfica: Inicio de sesión	27
Ilustración 4-7- Interfaz gráfica: Registro de usuario.....	28
Ilustración 4-8- Interfaz gráfica: Página pública de proyecto	29
Ilustración 4-9- Interfaz gráfica: Página pública de profesional	30
Ilustración 4-10- Interfaz gráfica: Zona de usuario	30
Ilustración 4-11- Interfaz gráfica: Perfil de usuario	31
Ilustración 4-12- Interfaz gráfica: Perfil de usuario.....	32
Ilustración 4-13- Interfaz gráfica: Proyectos de usuario	33
Ilustración 4-14- Interfaz gráfica: Valoraciones.....	33
Ilustración 4-15 - Interfaz gráfica: Nuevo proyecto	34
Ilustración 4-16 - Interfaz gráfica: Intranet de proyecto.....	35
Ilustración 4-17 - Interfaz gráfica: Intranet de tarea.....	36
Ilustración 4-18 - Base de datos: diagrama entidad-relación	38
Ilustración 4-19 - Capa de presentación: Cabeceras y pie de página	43
Ilustración 4-21 - Capa de presentación: Resolución 768px.....	44
Ilustración 4-22 - Capa de presentación: Resolución 320px	44
Ilustración 4-20 - Capa de presentación: Resolución 1024px	45
Ilustración 4-23 - Capa de presentación: Formularios	45
Ilustración 4-24 - Capa de presentación: Formularios	45
Ilustración 4-25 - Capa de aplicación: Páginas de Wordpress.....	47
Ilustración 4-26 - Capa de aplicación: Plantillas de Wordpress	48
Ilustración 4-27 - Capa de aplicación: Roles	48
Ilustración 4-28 - Capa de aplicación: Nuevas entradas.....	50
Ilustración 4-29 - Capa de aplicación: Advanced Custom Fields.....	50
Ilustración 4-30 - Capa de aplicación: Valoraciones pendientes.....	52
Ilustración 4-31 - Capa de aplicación: Opciones del tema	53
Ilustración 4-32 - Capa de aplicación: Cimy User Extra Fields	57
Ilustración 4-33 - Capa de aplicación: Comentarios	64
Ilustración 4-34 - Capa de persistencia: Base de datos.....	70
Ilustración 7-1 - Manual de usuario: Registro.....	90
Ilustración 7-2 - Manual de usuario: Tipos de registro.....	90
Ilustración 7-3 - Manual de usuario: Campos obligatorios.....	91
Ilustración 7-4 - Manual de usuario: Errores.....	91
Ilustración 7-5 - Manual de usuario: Registro completo	91
Ilustración 7-6 - Manual de usuario: Olvido de contraseña.....	91
Ilustración 7-7 - Manual de usuario: Resetear contraseña	92



Ilustración 7-8 - Manual de usuario: Formulario de inicio de sesión.....	92
Ilustración 7-9 - Manual de usuario: Error de contraseña.....	92
Ilustración 7-10 - Manual de usuario: Atajo a edición de perfil.....	93
Ilustración 7-11 - Manual de usuario: Perfil.....	93
Ilustración 7-12 - Manual de usuario: Profesionales mejor valorador.....	93
Ilustración 7-13 - Manual de usuario: Atajo a perfil público.....	94
Ilustración 7-14 - Manual de usuario: Profesionales.....	94
Ilustración 7-15 - Manual de usuario: Freelance o Empresa.....	94
Ilustración 7-16 - Manual de usuario: Opciones de búsqueda de profesionales.....	94
Ilustración 7-17 - Manual de usuario: Crear proyecto nuevo.....	95
Ilustración 7-18 - Manual de usuario: Nueva tarea.....	95
Ilustración 7-19 - Manual de usuario: Tareas	96
Ilustración 7-20 - Manual de usuario: Tipo de proyecto.....	96
Ilustración 7-21 - Manual de usuario: Poner en estado Buscando colaboradores.....	96
Ilustración 7-22 - Manual de usuario: Atajo a buscar proyectos	97
Ilustración 7-23 - Manual de usuario: Menú a Proyectos	97
Ilustración 7-24 - Manual de usuario: Ver todos los proyectos	97
Ilustración 7-25 - Manual de usuario: Opciones de búsqueda de proyectos	98
Ilustración 7-26 - Manual de usuario: Resultados de la búsqueda.....	98
Ilustración 7-27 - Manual de usuario: Colaborar en tarea	99
Ilustración 7-28 - Manual de usuario: Inscripción en tareas.....	99
Ilustración 7-29 - Manual de usuario: Inscripción correcta	99
Ilustración 7-30 - Manual de usuario: Menú a mis proyectos	100
Ilustración 7-31 - Manual de usuario: Acceder a la intranet del proyecto	100
Ilustración 7-32 - Manual de usuario: Estado de candidatura.....	100
Ilustración 7-33 - Manual de usuario: Proyecto.....	101
Ilustración 7-34 - Manual de usuario: Acceder a la tarea	101
Ilustración 7-35 - Manual de usuario: Mensajes.....	102
Ilustración 7-36 - Manual de usuario: Subir archivos.....	102
Ilustración 7-37 - Manual de usuario: Estados de la tarea.....	103
Ilustración 7-38 - Manual de usuario: Candidatos de la tarea.....	103
Ilustración 7-39 - Manual de usuario: Acceso a valoraciones.....	104
Ilustración 7-40 - Manual de usuario: Valoraciones pendientes	104
Ilustración 7-41 - Manual de usuario: Valoraciones	104