



Treball de Fi de Grau
GRAU D'ENGINYERIA INFORMÀTICA
Facultat de Matemàtiques
Universitat de Barcelona

**DEVELINK RED SOCIAL WEB PARA
DESARROLLADORES**

David García Prados

Director: Simone Balocco
Realitzat a: Departament de
Matemàtica Aplicada i Anàlisi. UB
Barcelona, 28 de gener de 2016

Índice

Abstract (Castellano)	5
Abstract (Català)	6
Abstract (English)	7
1. Introducción	8
1.1. Formas de plantear online las relaciones laborales	8
1.2. Análisis del estado actual de las redes sociales profesionales	9
1.2.1. Datos sobre candidatos de empleo	10
1.2.2. Datos sobre reclutadores	12
1.3. Competidores	14
1.3.1. LinkedIn	14
1.3.2. InfoJobs	15
1.3.3. Viadeo	15
1.3.4. Xing	15
1.3.5. Stack Overflow Careers	16
1.4. Motivación	19
1.5. Objetivos de la nueva propuesta	19
1.6. Organización de los contenidos de la memoria	20
2. Análisis de requerimientos	21
2.1. Funcionales	21
2.1.1. Diagramas de casos de uso	21
2.2. No funcionales	27
3. Diseño	30
3.1. Modelo Relacional	30
3.2. Explicación de las tablas	31
4. Tecnologías usadas	33
4.1. Servidor Xampp	33
4.2. Servidor Openshift	33
4.3. Framework PHP	33
4.3.1. Comparativa	33
4.3.2. Elección: Laravel	37
4.4. Composer	38
4.5. Framework Cliente: Bootstrap	38
4.6. Gestor de código: Git	39
5. Planificación	40

5.1. Fases.....	40
5.2. Seguimiento	41
5.3. Diagrama de Gantt	42
6. Arquitectura de Develink	43
6.1. Explicación del MVC usado en Develink	43
6.2. Estructura general creada inicialmente.....	44
6.3. Estructura propia de Develink.....	46
6.3.1. Public	46
6.3.2. App	47
6.3.3. Resources	48
7. Pruebas de funcionamiento y rendimiento.....	50
7.1. Revisión de la base de datos	50
7.2. Validación HTML5	51
7.3. Accesibilidad y usabilidad	51
7.4. Rendimiento	52
8. Conclusiones y propuestas de mejora	55
8.1. Conclusiones	55
8.2. Propuestas de mejora.....	56
8.2.1. Validación de empresas	56
8.2.2. Sistema de certificación de estudios	56
8.2.3. Mejora de las ofertas laborales directas	57
8.2.4. Recomendación de usuarios	57
8.2.5. Ampliar la gestión de proyectos.....	57
8.2.6. Contratos con fecha de caducidad	57
8.2.7. Implementación del panel de administrador	57
8.2.8. Chat interno	58
9. Bibliografía.....	59
10. Anexo	62
10.1. Aplicación Laravel en Openshift.....	62
10.1.1. Creación	62
10.1.2. Acceso remoto y gestión de aplicaciones	65
10.1.3. Colaboradores	65
10.2. Servidor (PHP, MySQL) usando Xampp	66
10.2.1. Configuración general.....	66
10.2.2. Incorporar aplicación	67
10.2.3. Configuración del servicio de correo electrónico	68

10.2.4. Fichero de configuración Apache	68
10.3. Manual de usuario.....	69
10.3.1. Primeros pasos.....	69
10.3.2. Manual de Trabajador.....	73
10.3.3. Manual de Empresa	79

Abstract (Castellano)

El proyecto documentado a continuación llamado “Develink red social web para desarrolladores” es una aplicación web que relaciona laboralmente empresas y desarrolladores.

La función principal es aportar a las empresas una simplificación y automatización de la gestión de sus recursos humanos, en especial del proceso de reclutamiento y selección, mediante la publicación de ofertas. Por lo tanto, se trata también de una útil herramienta de búsqueda de empleo para los desarrolladores. Para lograrlo, se identificarán habilidades y experiencias anteriores (referencias) en proyectos, constituyendo un currículum que será sometido a validación. Esa verificación es su aspecto más innovador, diferenciándose sustancialmente del resto de redes sociales.

La aplicación es escalable y modular, es decir, adaptable a nuevos requisitos y aplicable a otros ámbitos.

El presente documento detalla tanto el proceso de planificación, análisis, diseño e implementación de esta red social como las conclusiones y posibilidades de mejora extraídas al finalizarlo.

Abstract (Català)

El projecte documentat a continuació anomenat “Develink red social web para desarrolladores” és una aplicació web que relaciona laboralment empreses i desenvolupadors.

La funció principal és aportar a les empreses una simplificació i automatització de la gestió dels seus recursos humans, en especial del procés de reclutament i selecció, mitjançant la publicació d’ofertes. Per tant, es tracta també d’una eina útil de cerca de feina pels desenvolupadors. Per aconseguir-ho, s’identificaran habilitats i experiències anteriors (referències) en projectes, constituint un currículum que serà sotmès a validació. Aquesta verificació és el seu aspecte més innovador, diferenciant-se substancialment de la resta de xarxes socials.

L’aplicació és escalable y modular, és a dir, adaptable a nous requisits i aplicable a altres àmbits.

El present document detalla tant el procés de planificació, anàlisi, disseny i implementació d’aquesta xarxa social com les conclusions i possibilitats de millora extretes al finalitzar-ho.

Abstract (English)

The following project called “Develink red social web para desarrolladores” is a web application that establishes a professional relationship between companies and developers.

The main goal is to provide companies a simplification and automation of human resources management, especially for the recruitment and selection process, with respect to job offers publication. Moreover, this is a useful tool for developers as well in order to search for employment. To achieve it, skills and references in projects are identified, constituting a curriculum that will be subjected to validation. This verification is its most innovative feature, differing substantially from other social networks.

The application is scalable and modular, hence, it is adaptable to new requirements and applicable in other areas.

This document details the process of planning, analysis, design and implementation of this social network as well as inferred conclusions and suggestions for improvement.

1. Introducción

El concepto de relación laboral entre empleados y empresas es muy antiguo y, hasta hace relativamente poco tiempo, apenas cambiante. No obstante, últimamente ha sufrido una importante transformación debido a la mejora de las tecnologías que permiten crear páginas web y, sobretodo, al acceso generalizado a una conexión a Internet cada vez más veloz y accesible desde una gran variedad de dispositivos portátiles. Tanto es así, que se ha convertido en prácticamente imprescindible tanto para las personas que buscan empleo como para las que lo ofrecen, dejando atrás a los clásicos diarios y carteles.

1.1. Formas de plantear online las relaciones laborales

Una vez vista la necesidad de herramientas online que faciliten el contacto profesional y, por lo tanto, la viabilidad de su creación y existencia, se pueden plantear de dos formas:

- **Orientadas a ofertas laborales.** En este caso, las empresas exponen sus necesidades (y en algunos casos lo que ofrecen por ellas) y los usuarios interesados se ponen en contacto, pasando a ser candidatos. Este concepto es el más similar a los clásicos y una de las páginas web más conocidas que lo explota es InfoJobs [1].



Figura 1. El buscador de InfoJobs está centrado en ofertas laborales. Fecha: 27/11/2015

- **Orientadas a perfiles profesionales.** Consiste en hacer público (admitiendo diferentes niveles de privacidad) el currículum vitae de las usuarios, el cual incluye intereses, habilidades, formación recibida y experiencias laborales anteriores. El mayor exponente de este concepto es LinkedIn [2]. Actualmente es fácil de encontrar pero anteriormente era muy poco habitual debido al elevado coste que suponía publicitarse para un particular con necesidad de empleo y el breve periodo de visibilidad que le ofrecía.



Figura 2. El buscador de LinkedIn está centrado en los perfiles. Fecha: 27/11/2015

Ambos conceptos son importantes y no excluyentes, por lo tanto, toda página exitosa en este sector los incluye aunque es probable que se centre más en uno de ellos.

1.2. Análisis del estado actual de las redes sociales profesionales

En este apartado analizaré un estudio realizado en 2014 por InfoEmpleo y Adecco titulado “Redes Sociales y Mercado de Trabajo” [3]:

- El ámbito es España.
- La muestra tomada diferencia claramente los candidatos (6646 en los que se incluyen empleados, desempleados y autónomos) y los profesionales de Recursos Humanos o reclutadores (284 personas).
- Residentes en España entre 18 y 67 años.
- Es cuantitativa y 100% online.

Actualmente, es importante tener una gran visibilidad en Internet y buena reputación. A los candidatos les aporta un aumento significativo en las probabilidades de encontrar empleo. Un 77% de los encuestados crearon un perfil en redes sociales por ese motivo. Además, también es útil para las empresas ya que les permite reclutar a más personas que se ajusten mejor a sus requisitos. Un 70% de sus departamentos de Recursos Humanos usaron redes sociales y portales de empleo con ese fin, un 20% más que en 2011. A continuación se analizan por separado los candidatos y los reclutadores.

1.2.1. Datos sobre candidatos de empleo

De la muestra tomada, 61% son desempleados, un 33% empleados y el 6% restante autónomos o freelance. A continuación se muestran resultados relevantes:

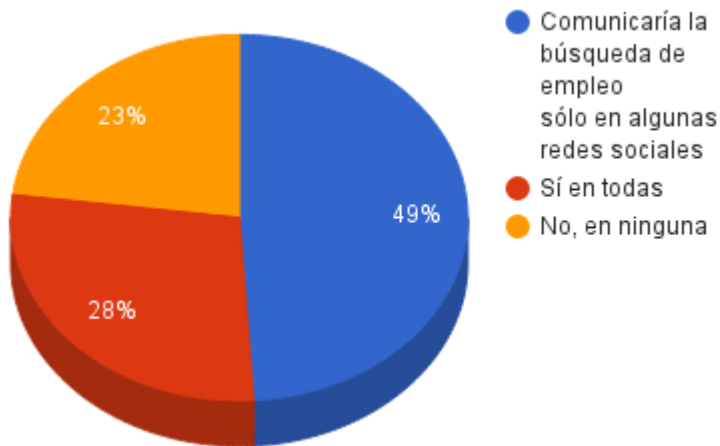


Figura 3. Búsqueda de empleo en redes sociales de los candidatos encuestados. Fecha: 08/12/2015

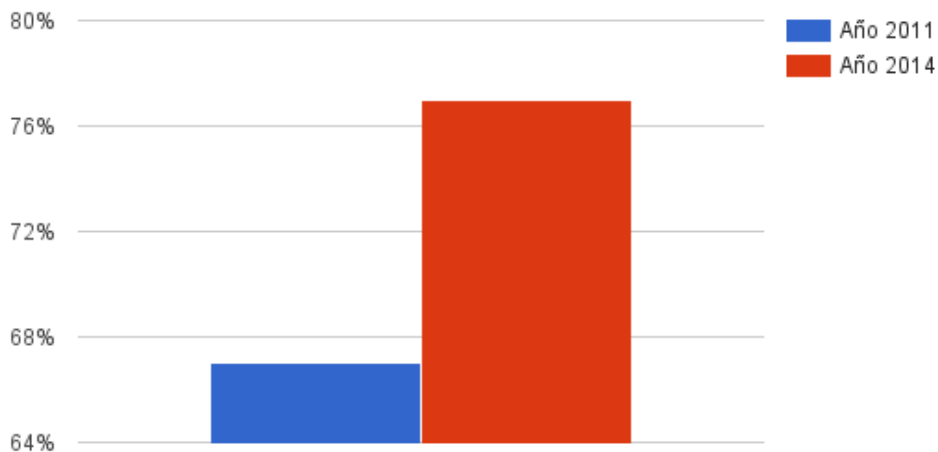
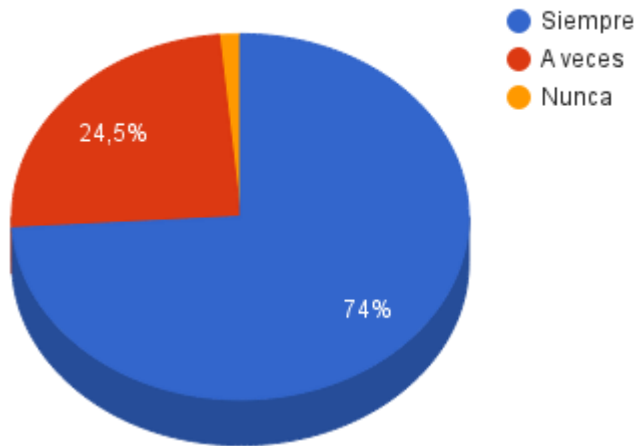
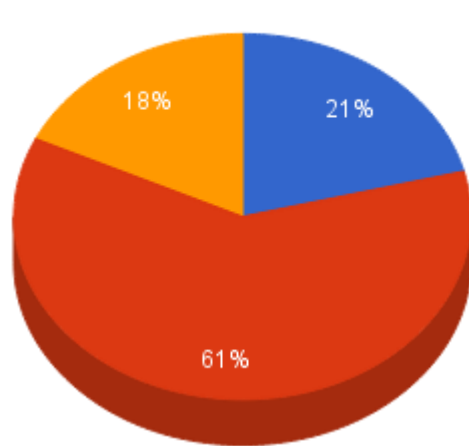


Figura 4. Comparativa de uso de redes sociales como herramienta de empleo. En 2014 fue un 77% (49% más 28% de la Figura 3) y en 2011, un 67%. Hay un incremento de un 10% en 3 años. Fecha: 08/12/2015

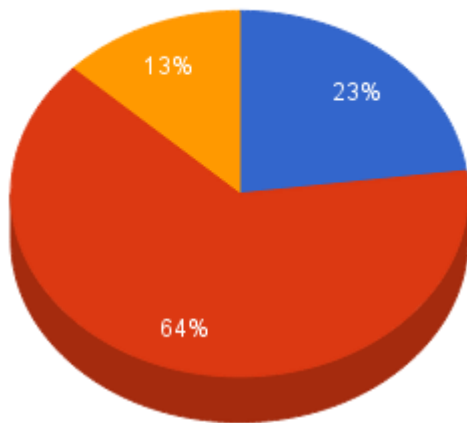
Portales de Empleo



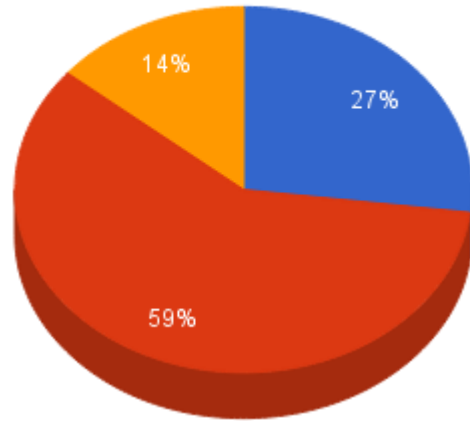
Redes Sociales



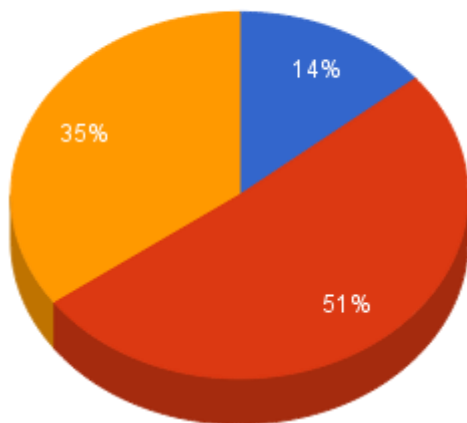
Contactos



Empresas de Selección



Anuncios en Prensa



Cazatalentos

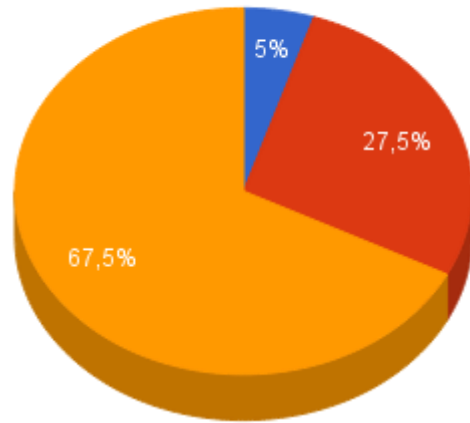


Figura 5. Medios que usan los candidatos para buscar empleo y frecuencia. Fecha: 08/12/2015

Se puede observar una gran popularidad de los portales de empleo y las redes sociales que contrasta con la escasa búsqueda en los anuncios en prensa. También es destacable que se sigue confiando en los contactos pero no en los cazatalentos.

A los usuarios que emplean redes sociales para buscar trabajo, se les preguntó cuáles utilizan. Respondieron lo siguiente:

Red social	Buscan empleo
LinkedIn	74%
Facebook	46%
Twitter	23%
Google+	15%
Foros	12%
Blogs	10%
Xing	8%
Instagram	3%
Tuenti	2%
Pinterest	1%

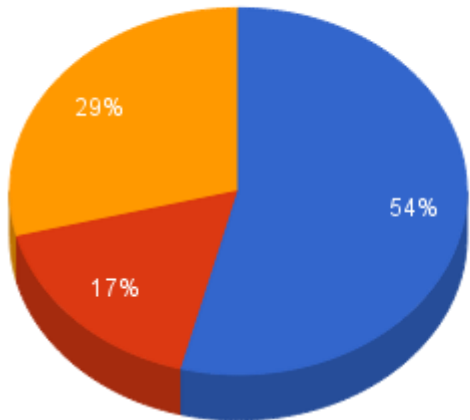
La tabla indica que la más útil para los encuestados es LinkedIn y la segunda Facebook. LinkedIn es mucho más popular que Facebook para buscar empleo a pesar de tener menos usuarios registrados ya que está especializada en el ámbito profesional. También es destacable el reducido porcentaje de Xing ya que comparte objetivo con LinkedIn. En este caso, se debe a que la mayoría de encuestados no la conocían.

1.2.2. Datos sobre reclutadores

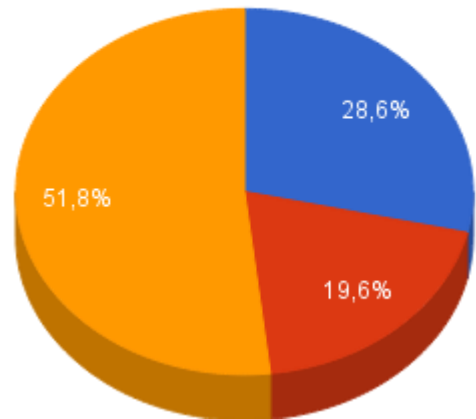
De la muestra tomada, el 32% son directores de departamento, un 28% empleados y un 40% mandos intermedios.

A continuación se muestran cuáles usan para buscar talento y con qué frecuencia.

LinkedIn

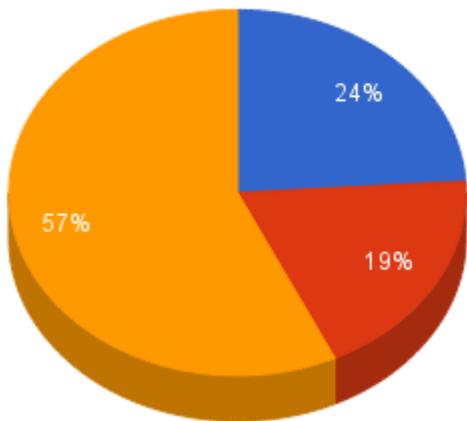


Twitter

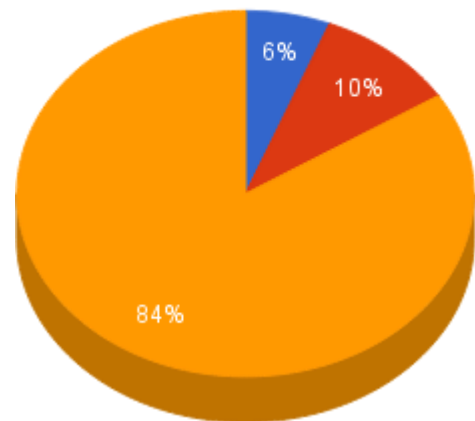


● Siempre
● A veces
● Nunca

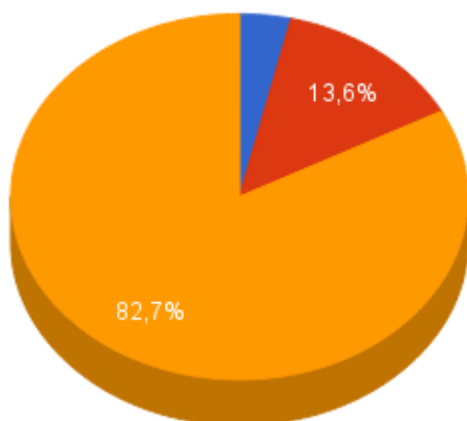
Facebook



Youtube



Xing



Viadeo

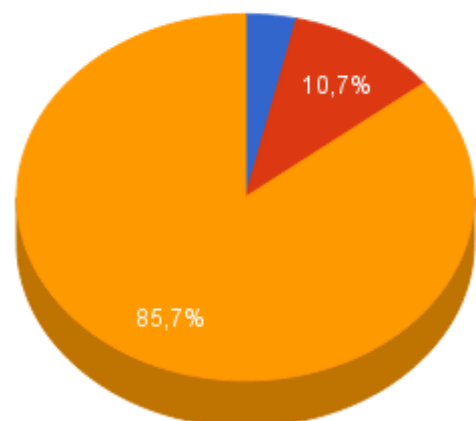


Figura 6. Redes sociales que usan para buscar talento. Fecha: 08/12/2015

En la figura anterior se puede observar que LinkedIn es la más usada para reclutar seguida a bastante distancia por Twitter y Facebook.

Por lo tanto, se extrae de este estudio que, tanto los candidatos como los reclutadores, para lograr sus objetivos usarán medios que estén centrados en el ámbito laboral, en los que se sientan seguros y que sean generalmente usados.

1.3. Competidores

En la actualidad hay gran variedad de páginas web con funcionalidades sociales relacionadas con el mundo laboral. A continuación se comentan algunas relevantes.

1.3.1. LinkedIn

Fundada en diciembre de 2002, lanzada en mayo de 2003 y en 2015 superó los 400 millones de usuarios. Características principales:

- Lista de contactos llamada Conexión.
- La red de contactos y posibles conocidos se forma a partir de las conexiones directas (la lista anterior), las de segundo grado (conexiones de nuestros contactos) y de tercero. No se contemplan más porque crecería enormemente¹.
- Los usuarios pueden invitar a otros como contacto y los receptores aceptarlo o catalogarlo como "spam". Cada rechazo por no deseada puntúa como voto negativo y, si se acumula una gran cantidad, se restringe la cuenta temporalmente. En caso de reincidencia, es cerrada.
- Posibilidad de subir currículum vitae o diseñar un perfil con fotografía, habilidades y experiencias laborales.
- Los empleadores pueden publicar ofertas laborales y buscar a posibles candidatos.
- Los usuarios pueden seguir a empresas y recibir notificaciones cuando ofrecen un puesto.
- Dispone de un centro de ayuda con información sobre cómo empezar a utilizar la página, crear una identidad profesional adecuada, encontrar buenas ofertas de empleo, gestionar correctamente el perfil de empresa, compartir ideas, colaborar con otros usuarios...

¹ Aumenta de forma aproximadamente exponencial debido a la teoría de los seis grados de separación (formulada por Frigyes Karinthy en 1930). Después de un estudio de 2011 de la "Università degli Studi di Milano", en redes con tantos usuarios se consideran aproximadamente 4 grados o saltos [45].

1.3.2. InfoJobs

Bolsa de empleo privada online, especializada en el mercado español. Fue fundada en 1998 y actualmente hay registradas más de 450.000 empresas. Características principales:

- Búsqueda de ofertas laborales sin necesidad de registro.
- Permite publicar ofertas: las empresas registradas muestran detalles de los puestos de trabajo y las personas que consideran que cumplen los requisitos se pueden apuntar pasando a ser candidatos (sólo si están registradas y, por lo tanto, tienen publicado un currículum).
- Las empresas pueden consultar posibles candidatos sin necesidad de publicar ofertas.
- Posibilidad de publicar proyectos y subastarlos a proveedores
- Servicio InfoJobs TV: información actualizada sobre cómo redactar un currículum, afrontar entrevistas laborales...

1.3.3. Viadeo

Fundada en mayo de 2004. Es la segunda red social profesional más grande (a mucha distancia de LinkedIn) ya que en 2014 tenía 65 millones de usuarios. Características principales:

- Gestión de contactos profesionales.
- Búsqueda de oportunidades laborales y alianzas de negocio.
- Consulta a expertos del mismo sector profesional.
- Posibilidad de crear perfil que muestre habilidades.

1.3.4. Xing

Red social profesional creada en 2003 y hasta 2006 llamada OpenBC. En 2010 ya tenía más de 10 millones de usuarios registrados. Características principales:

- Tiene un alto grado de privacidad y control de "spam" ya que cumple de forma estricta la Ley de protección de datos europea.
- Prohíbe de forma explícita actividades de MLM (Multi-Level Marketing) también conocidas como venta piramidal.
- Gestión de contactos.
- Facilita establecer nuevas conexiones entre usuarios.
- Visualización de red de contactos. Permite saber a partir de cuántos está conectado con otros.
- Las empresas pueden publicar ofertas y los usuarios buscarlas mediante la funcionalidad "Xing Empleo".

- Es posible la búsqueda de personas por nombre, ciudad, sector, empresa, área profesional...
- Incluye foros y grupos donde se plantean y resuelven dudas, se comparte información de interés...
- Los usuarios pueden saber cuántas veces se ha consultado su perfil y por parte de quien.

1.3.5. Stack Overflow Careers

Bolsa de trabajo exclusiva para desarrolladores [6]. Como su nombre indica, pertenece a la página web Stack Overflow (en la que la comunidad de usuarios plantea y resuelve dudas de programación). Al compartir sector con el presente proyecto, Develink, será muy útil su análisis para tomar sus puntos fuertes como ejemplo a seguir e intentar mejorar sus carencias. Características principales:

- En la página principal se muestra un buscador de empleo por habilidades, lugares y empresas.

Keywords **Location**

Job Title, Keyword, or Company City, Country or Zip Code Search

Keywords: [iPhone](#) - [JavaScript](#) - [jQuery](#) - [AngularJS](#) - [Linux](#) - [C++](#) - [Python](#) - [Java](#) - [PHP](#)
Locations: [Amsterdam](#) - [Berlin](#) - [San Francisco](#) - [London](#) - [New York](#) - [Chicago](#) - [Remote](#)
Companies: [ING](#) - [eHarmony](#) - [Time Doctor](#) - [ThoughtWorks Deutschland GmbH](#) - [WorldRemit](#)

Figura 7. El buscador de Stack Overflow Careers. Fecha: 03/12/2015

- Dispone de una pestaña “Jobs” en la que se listan ofertas de empleo cercanas a la ubicación del usuario. Para cada oferta se muestra:
 - Si es destacada
 - Título
 - Localización
 - Breve descripción
 - Habilidades en forma de tags
 - Tiempo transcurrido desde su publicación

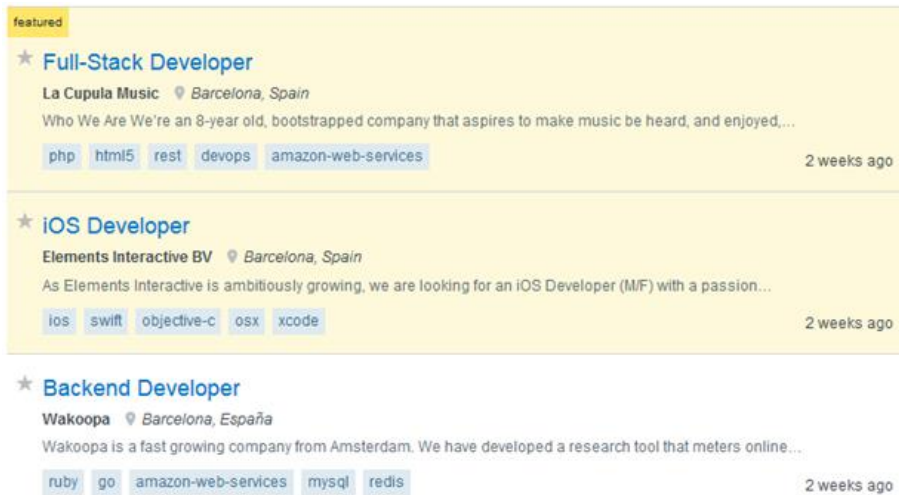


Figura 8. Extracto de 3 ofertas de la lista de Stack Overflow Careers. Fecha: 03/12/2015

- Una vez se hace clic en una oferta concreta, ésta se muestra detalladamente. Además de lo mostrado, hay la descripción de la empresa y el botón “Apply” para mostrar interés por el puesto y pasar a ser candidato.

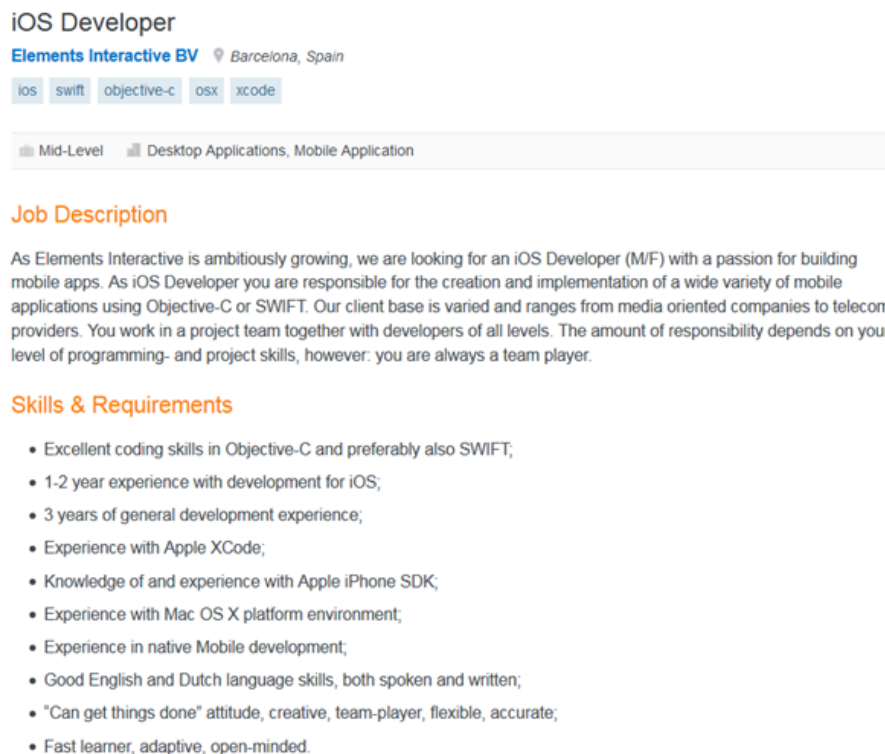


Figura 9. Una oferta detallada en Stack Overflow Careers. Fecha: 03/12/2015

- En la pestaña “Companies” se muestran todas las empresas registradas y las habilidades demandadas. De forma análoga a “Jobs”, permite realizar búsquedas y al hacer clic en una de ellas, se muestra más información.

- En "Cities", se ofrece el servicio "Explore Cities beta" que consiste en mostrar un listado algunas ciudades en las que se ofrece empleo. Como su nombre indica, está en pruebas, y el objetivo de la versión final es ampliarlo a todas las del mundo. Para cada una de ellas, indica:
 - Una descripción del lugar, con especial énfasis en las oportunidades laborales que ofrece.
 - Los sueldos medios por puestos de trabajo.
 - Una enumeración de las industrias más importantes.
 - Sus empresas más importantes.



Figura 10. Ciudad de Nueva York en Stack Overflow Careers.Fecha: 03/12/2015

- Dispone de una función de búsqueda de candidatos. Para mostrar los siguientes datos relevantes es necesaria una suscripción de pago elevada ya que está orientada a empresas:
 - Nombre, apellidos, lugar de residencia, página web personal.
 - Descripción
 - Historial de empleos, estudios y habilidades.
 - Actividad en Stack Overflow. Concretamente, las mejores respuestas.
 - Los proyectos de código abierto en los que ha contribuido (en webs de desarrollo colaborativo como Github).
 - Aplicaciones que ha creado.
 - Blogs y artículos que ha escrito.
 - Libros favoritos.
 - Otra información de interés para posibles empleadores.

1.4. Motivación

El proyecto Develink surge principalmente por los siguientes motivos:

- Mis ganas de aprender un framework PHP moderno, potente, bien documentado y extensamente usado.
- Aplicar mis conocimientos aprendidos a lo largo del grado, especialmente los de programación web, a un proyecto real.
- La necesidad de mejorar las redes sociales profesionales para avanzarse a los nuevos requisitos que pueda demandar el mercado laboral en un futuro cercano.

1.5. Objetivos de la nueva propuesta

A partir del anterior análisis del estado actual de las redes sociales actuales y de los servicios ofrecidos por la competencia, se plantea crear una red social profesional que:

- Se dirigirá inicialmente al empleo de desarrolladores. La elección de este colectivo en particular, es debida a que habitualmente pueden desempeñar todo su trabajo de forma no presencial vía Internet.
- Será suficientemente flexible, para adaptarse a los máximos sectores profesionales posibles.
- Se centrará en las habilidades y experiencias previas de las personas, adquiridas en proyectos de empresas que se incluirán en un currículum vitae virtual.
- Someterá a un proceso de validación la atribución de conocimientos en experiencias por parte de los trabajadores para garantizar su veracidad. Las encargadas de su confirmación o desmentimiento serán las empresas referenciadas.
- Por lo anteriormente mencionado, su control de autenticidad en el registro de empresas será crucial.
- Permitirá a las empresas una gestión de los recursos humanos de sus proyectos de forma online, segura y simple. Podrán publicar ofertas, mostrando u ocultando su identidad, dirigidas a todos los usuarios en general, sólo a los que cumplan sus requisitos o enviarlas directamente de forma privada a unos en concreto. Además, la aplicación les permitirá hacer una selección de uno de los candidatos entre todos los se hayan interesado por la oferta. Una vez se realice la contratación, si tiene fecha de expiración, habrá la posibilidad de renovarla cuando acabe el periodo. También se contemplará, en cualquier momento, la opción de despido.
- Dispondrá de un sistema de valoraciones que establecerá la reputación tanto de empleados como empresas. Para que sea útil y fiable, sólo se podrá utilizar cuando finalice el contrato y estará vetado a usuarios despedidos.

1.6. Organización de los contenidos de la memoria

En esta memoria se tratan en profundidad los siguientes puntos:

- Capítulo 2. Explicación del análisis de los requisitos de la aplicación web Develink.
- Capítulo 3. Se muestra y comenta el Modelo Relacional de la base de datos.
- Capítulo 4. Resumen de las tecnologías usadas.
- Capítulo 5. Explicación de la planificación del proyecto.
- Capítulo 6. Proceso detallado de la construcción de Develink.
- Capítulo 7. Se exponen y analizan los resultados de pruebas de fiabilidad y rendimiento de la aplicación.
- Capítulo 8. Se comentan las conclusiones extraídas del trabajo realizado además de proponer mejoras y extensiones interesantes que se podrían aplicar en un futuro.
- Capítulo 9. Listado de las fuentes de información consultadas.
- Capítulo 10. Se incluye información útil de consulta que no tiene relevancia suficiente para formar parte del contenido principal de la presente memoria.

2. Análisis de requerimientos

Antes del desarrollo se analizó qué requisitos funcionales y no funcionales había que satisfacer para lograr los objetivos marcados. Debido a la complejidad y extensión del proyecto, han sido revisados y refinados en diversas ocasiones a lo largo de la implementación.

El requisito fundamental es ofrecer una red social profesional segura e intuitiva que permita poner en contacto a candidatos y a empresas para realizar contrataciones satisfactorias que permitan llevar a cabo proyectos.

2.1. Funcionales

Se realiza una separación de requisitos según el rol de cada usuario. Existen 5:

- **Anónimo:** Usuario sin identificarse en la web.
- **Identificado:** Usuario que ha iniciado sesión.
- **Trabajador:** Usuario identificado que representa a un empleado o a un candidato.
- **Empresa:** Usuario identificado como una compañía o empresa con capacidad reclutadora y de despido.
- **Administrador:** Usuario identificado que tiene la capacidad de gestionar toda la aplicación web, supervisando lo que realiza el resto y solventando posibles problemas con el objetivo principal de garantizar su correcto funcionamiento.

2.1.1. Diagramas de casos de uso

A continuación se detallan los diagramas de casos de uso (uno por rol).

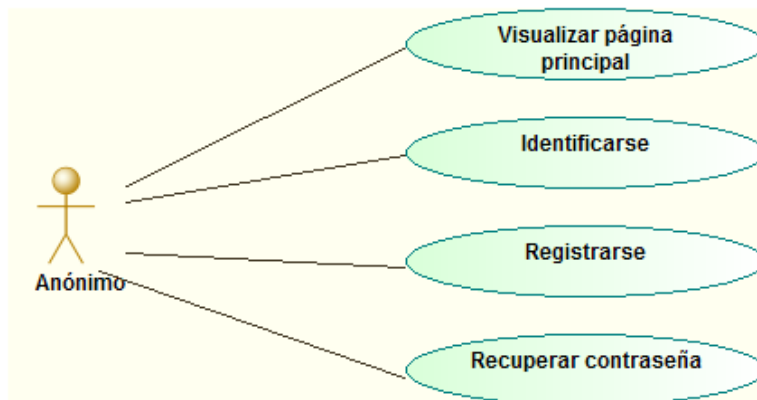


Figura 11. Anónimo: diagrama de casos de uso

Como observa en el diagrama anterior, el **usuario anónimo** puede realizar las siguientes acciones:

- Visualizar la página principal: lee información general sobre el funcionamiento de Develink.
- Identificarse: inicia sesión mediante correo electrónico y contraseña previamente registrados.
- Registrarse: rellena un formulario para formar parte de la red social que le permitirá su posterior identificación.
- Recuperar la contraseña: introduce su correo electrónico y recibirá un enlace para cambiar su contraseña.

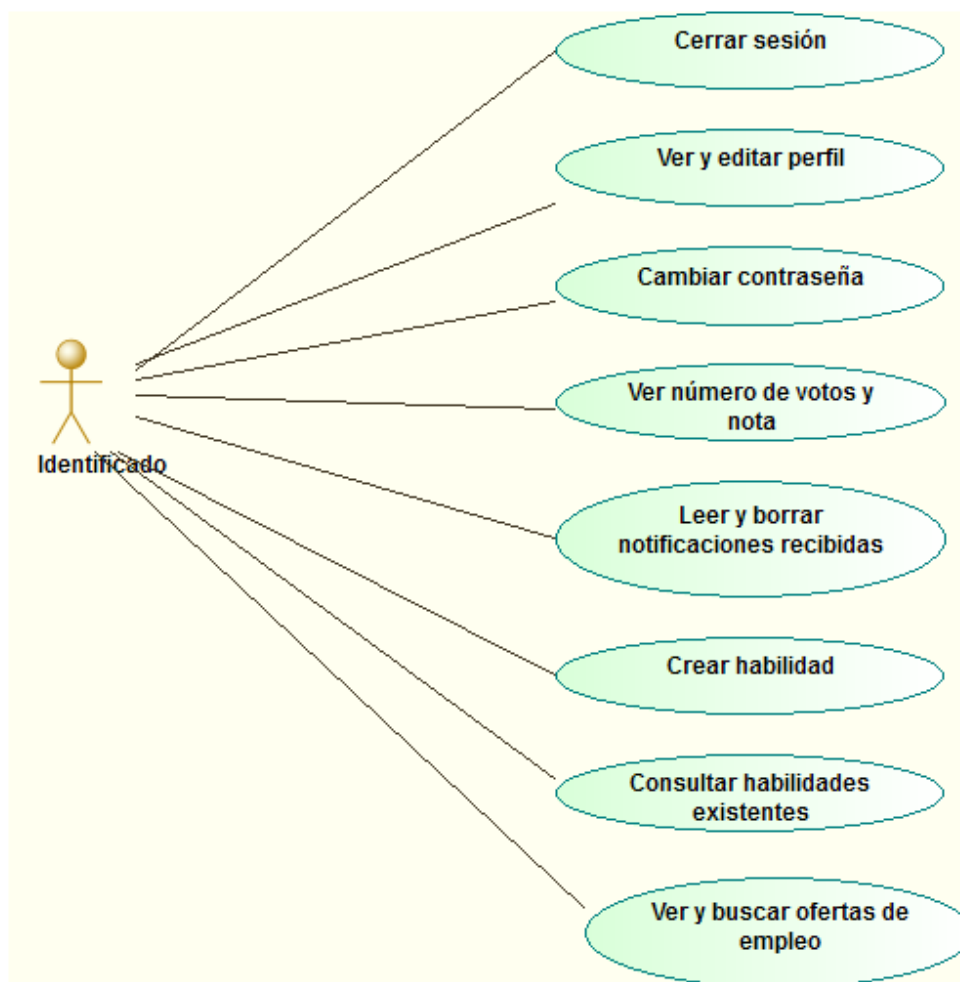


Figura 12. Identificado: diagrama de casos de uso

El **usuario identificado**, además de visualizar la página principal, puede realizar:

- Cerrar sesión: se desconecta cambiando su rol a usuario anónimo.
- Ver y editar perfil: modifica descripción, url de web de contacto y fotografía.
- Cambiar contraseña.

- Ver número de votos y nota: visualiza cuántos usuarios lo han votado y, si son 5 o más, qué valoración tiene entre 0 y 100. También se incluye una no numérica que indica “Bad” si [0,50), “Fair” si [50,70) “Good” si [70, 90) y “Excellent” si [90,100]. Como las notas recibidas son anónimas y de 1 a 4 (para evitar valor central), las aquí mostradas se han calculado aplicando “25 x suma / número”. La suma y el número de valoraciones, son atributos del usuario (se pueden observar en la “figura 16 del apartado 3.Diseño”). La precisión del resultado se ha limitado a 2 decimales.
- Leer y borrar notificaciones recibidas: ve avisos sobre interacciones que le afecten y puede eliminarlos fácilmente uno a uno.
- Crear habilidad: permite crear una nueva habilidad indicando nombre, descripción, url de fotografía y página web con más información. Esta acción debe ser moderada por un administrador para evitar malos usos.
- Consultar habilidades existentes: busca habilidades en la red social. Acepta múltiples, separadas por comas. No hace falta escribir los términos exactos ya que se aplica el concepto de Distancia de Levenshtein [46] para encontrar los resultados más relevantes (ordenados por similitud, de mayor a menor). Al hacer clic sobre una de ellas se mostrará una ficha con su descripción.
- Ver y buscar ofertas de empleo.

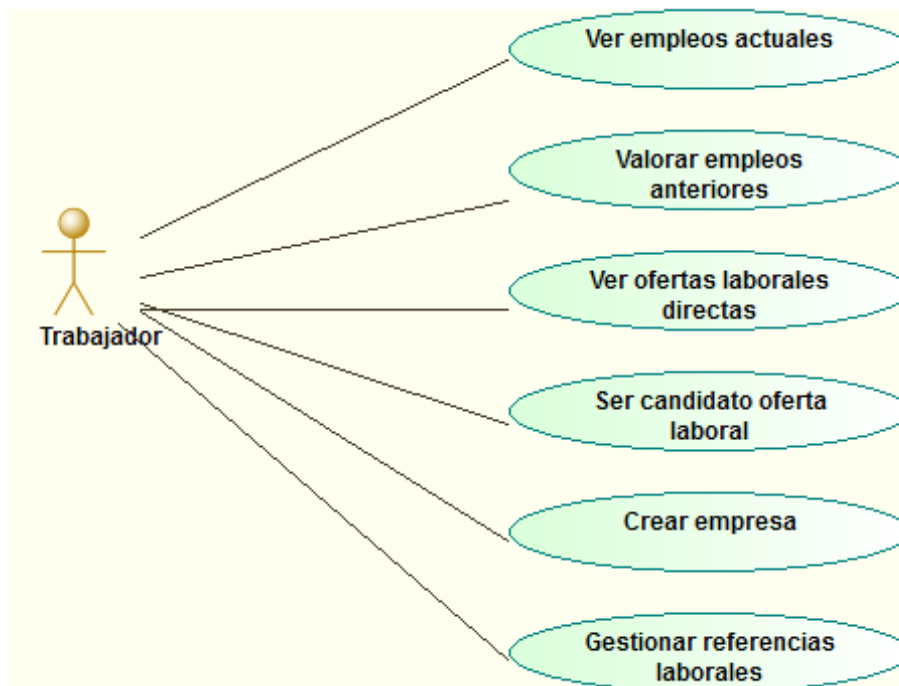


Figura 13. Trabajador: diagrama de casos de uso

El **trabajador**, además de las acciones de usuario identificado, puede realizar:

- Ver empleos actuales: se muestran los lugares en los que está trabajando y su cargo.
- Valorar empleos anteriores: puntúa empresas (valor entre 1 y 4) en las que finalizó un contrato sin despido.
- Ver ofertas laborales directas: visualiza y busca entre las ofertas enviadas de forma personalizada e individualizada por empresas.
- Ser candidato de una oferta laboral: selecciona una oferta entre las del listado general o el directo.
- Crear empresa: permite registrar una cuenta con rol empresa. Esta acción debe ser moderada por un administrador para evitar malos usos.
- Gestionar referencias laborales: permite borrar referencias laborales y solicitar a empresas que les validen sus experiencias anteriores.

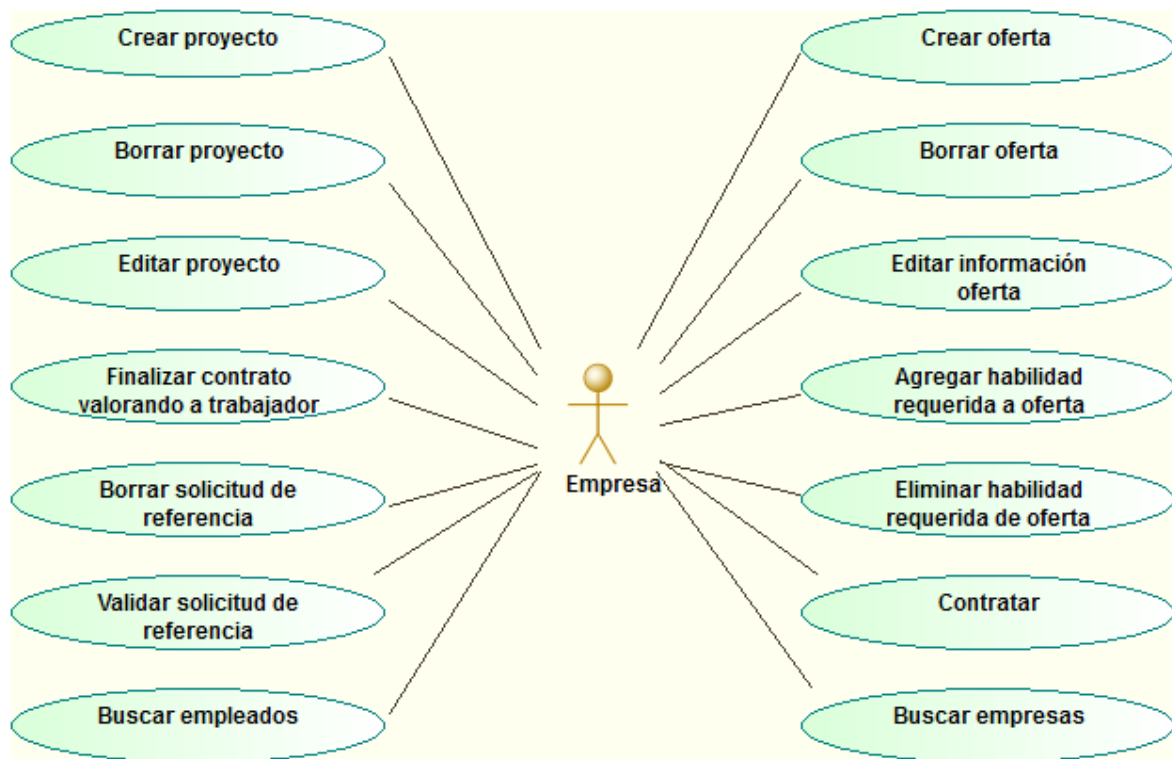


Figura 14. Empresa: diagrama de casos de uso

La **empresa**, además de las acciones de usuario identificado, puede realizar:

- Crear proyecto: crea un nuevo conjunto de ofertas laborales vacío. Se le asigna nombre, descripción, url de fotografía y página web. Utilizado para organizarlas y separarlas por tareas.
- Borrar proyecto: elimina el proyecto y todo su contenido.
- Editar proyecto: permite cambiar la información básica (nombre, descripción...).
- Finalizar contrato valorando a trabajador: permite terminar el contrato con un trabajador y asignarle una nota (entre 1 y 4). Puede finalizar de dos formas: completado (de forma habitual) o con despido (prematuramente). La principal diferencia es que sólo en caso que se complete, la experiencia obtenida pasa a formar parte del empleado (se considera experiencia o referencia a cada habilidad que formaba parte de la oferta con la que fue contratado).
- Borrar solicitud de referencia: elimina una solicitud de experiencia laboral anterior, es decir, la empresa no reconoce que el empleado haya trabajado en su empresa y/o aprendido esa habilidad.
- Validar solicitud de referencia: confirma que el empleado ha trabajado en la empresa y ha adquirido ese conocimiento.
- Buscar empleados: permite encontrar empleados entre todos los existentes. Una vez seleccionado uno, se visualiza su currículum.
- Crear oferta: publica una oferta laboral en la cual se indica título, descripción y si debe mostrar el nombre de la empresa anunciante (el caso en que no se muestra, se le ha denominado "phantom"). Las habilidades demandadas se pueden agregar posteriormente.
- Borrar oferta: elimina una oferta laboral de las que publicó.
- Editar información oferta: permite el cambio de título, descripción y si es "phantom" o no. Además, es posible dirigirla a un listado de usuarios concretos (llamada "direct") y que no sea pública.
- Agregar habilidad requerida a oferta: permite añadir una habilidad de las existentes como requisito a la oferta. Para simplificar la búsqueda, se establecen categorías de habilidades que engloban ramas del conocimiento o áreas profesionales.
- Eliminar habilidad requerida a oferta.
- Contratar: selecciona a uno de los empleados que se han interesado por la oferta y lo contrata.
- Buscar empresas: permite encontrar empresas entre todas las existentes. Una vez seleccionada una, se visualiza su descripción básica.

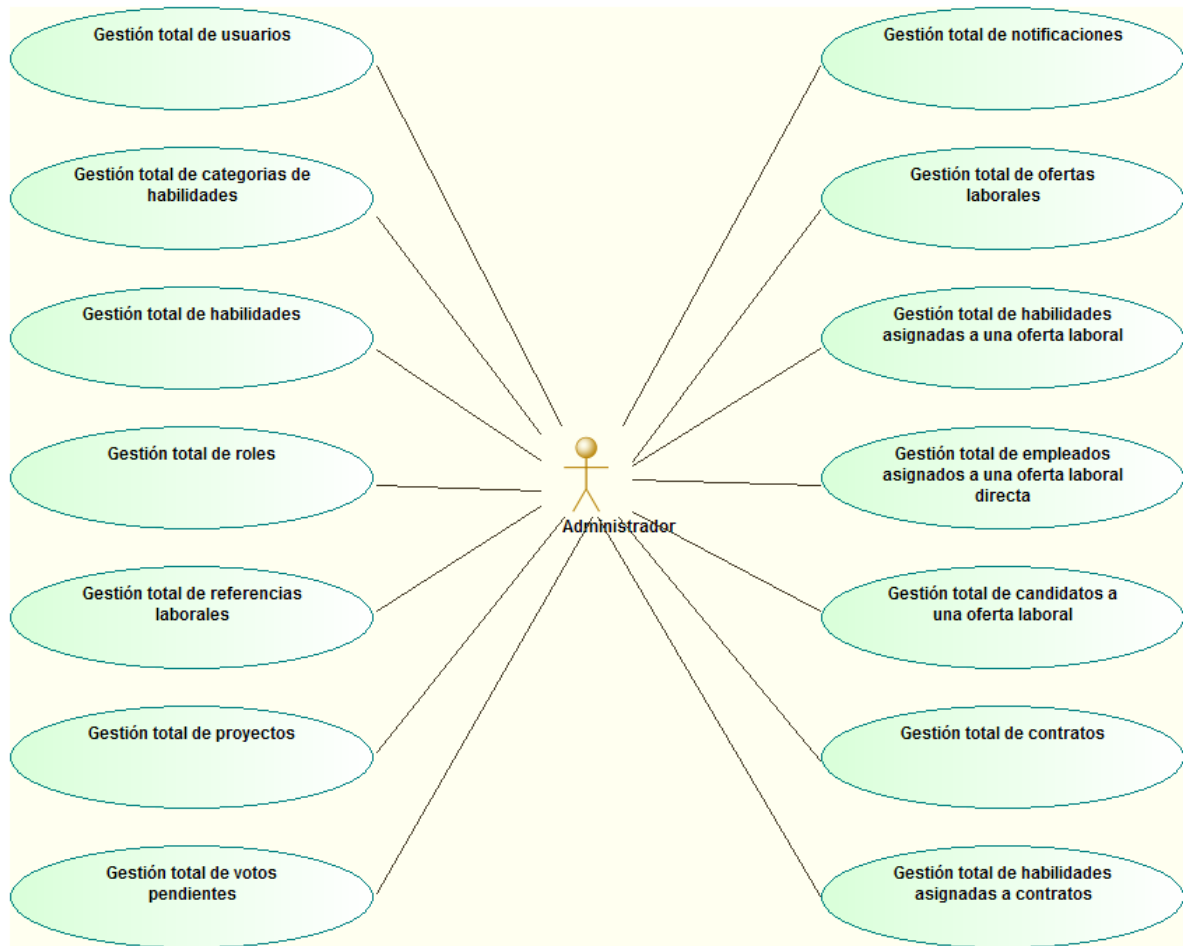


Figura 15. Administrador: diagrama de casos de uso

El **administrador**, además de las acciones de usuario identificado, puede realizar una gestión de todos los componentes de la red social. Es completa ya que dispone de permisos de lectura, creación, modificación y eliminación sobre:

- Usuarios de todos los roles.
- Habilidades.
- Categorías de habilidades.
- Roles: los existentes son trabajador, empresa y administrador pero esta opción permite modificarlos para adaptarse a nuevos requisitos de negocio.
- Referencias laborales.
- Proyectos.
- Votos pendientes.
- Notificaciones.
- Ofertas laborales.
- Habilidades asignadas a una oferta laboral.
- Empleados asignados a una oferta laboral directa.
- Candidatos a un puesto de trabajo

- Contratos
- Habilidades asignadas a contratos

Respecto la modificación de usuarios y roles, hay que restringir o incluso prohibir los cambios referentes a administrador ya que si no se tiene en cuenta, como todos ellos tienen el mismo nivel de privilegio, se podría dar el caso que uno pudiera tomar el control total de la aplicación (eliminando previamente a todos los demás). El problema se podría solucionar fácilmente accediendo y modificando directamente la base de datos. No obstante, es preferible prevenirlo mediante lo explicado anteriormente.

2.2. No funcionales

Como cualquier aplicación que pueda dirigirse al público en general, debe cumplir las “5 es” de usabilidad (empleadas en el “User Centered Design” estudiado en la disciplina Factores Humanos y Computación). Son las siguientes:

- **“Effective”** (efectiva). Que el usuario pueda completar sus tareas de forma precisa y completa logrando así sus objetivos.
- **“Efficient”** (eficiente). Que las realice de forma rápida.
- **“Error tolerant”** (tolerante a errores). Que haya los mínimos errores y recuperarse de ellos sea fácil.
- **“Easy to learn”** (fácil de aprender). Debe ser intuitiva, tanto para los usuarios que se inician como para los más experimentados, permitiendo un aprendizaje rápido y continuo.
- **“Engaging”** (atractiva, estimulante). Que se sienta lo más cómodo posible (disfrute) usándola.

A continuación se detalla qué es imprescindible para lograrlo.

- Efectividad:
 - Un buen análisis funcional y un diseño que se ajuste a él.
 - Realizar tests de usabilidad con expertos usando heurísticas como las de Nielsen [44] y siguiendo sus recomendaciones.
 - Que los usuarios puedan probar desde el principio y en diversas fases de desarrollo la aplicación. Tener en cuenta su opinión.
- Eficiencia:
 - Escalabilidad: La aplicación debe soportar mucho tráfico y, para conseguirlo, poder ejecutarse simultáneamente en diferentes servidores.

- Velocidad de la página web: Reducir el número de peticiones al servidor y el tamaño de los recursos (imágenes, vídeos...).
- Tolerante a errores, fácil de aprender y estimulante:
 - Accesibilidad: Ofrecer una única interfaz adaptable (“responsive”) que se pueda utilizar desde diferentes dispositivos, sistemas operativos y navegadores web.

Otro requisito no funcional de gran importancia en una aplicación web es la seguridad ya que es necesaria para garantizar su correcto funcionamiento, la integridad de la información y la privacidad de los usuarios. Los principales problemas y soluciones son los siguientes:

- **Inyecciones de código SQL:** la introducción de consultas SQL en campos de texto y enviados al servidor provoca que se pueda acceder y alterar fácilmente el contenido de su base de datos. Se puede evitar eliminando el código de los parámetros de las consultas.
- **Peticiones falsas:** si se pasan parámetros por “GET”, la url los contiene y envía peticiones al servidor. Para realizar consultas de información pública puede ser incluso positivo ya que permite a los buscadores de internet indexarla pero, en caso que implique acceso o modificación de datos privados, es una vulnerabilidad. Para evitarla se deben pasar por “POST”.
- **Inserción de HTML y Javascript:** similar a la inyección SQL pero, en este caso, afecta más en el lado del cliente. Permite modificar la página web que ven los usuarios y, por tanto, introducir código malicioso.
- **Exposición de información en la base de datos:** es inevitable guardar datos confidenciales como contraseñas en la base de datos y no es posible asegurar que sus administradores no le den un mal uso. Por lo tanto, hay que encriptarlos y existen funciones que realizan esa tarea.
- **Espionaje de datos:** es posible analizar los datos que circulan por una página web. Para evitar que visualicen los privados, hay que transportarlos de forma segura (encriptados). Se puede lograr mediante el protocolo HTTPS.
- **Ataques de fuerza bruta:** es posible que alguien que conozca el nombre de un usuario intente averiguar su contraseña para acceder a su cuenta. Para evitar que lo consiga, se puede limitar el número de intentos e incluso enviar un correo informando de lo que sucede al usuario en cuestión. Otra opción, que es la que se ha aplicado en la aplicación Develink, consiste en utilizar el correo electrónico (que es privado) para iniciar sesión en lugar del nombre de usuario (que es visible por otros). Ambas opciones pueden ser incómodas pero, en algunos casos, seguridad y usabilidad están reñidos y hay que buscar un equilibrio.

- **Sesiones sin caducidad:** si una sesión no tiene caducidad y alguien la deja abierta, otro usuario con acceso físico al dispositivo puede realizar cambios en su cuenta sin conocer sus credenciales (mientras no la cierre). La solución es establecer un tiempo límite y, cuando transcurra, cerrar la sesión. Otra opción complementaria es pedir la contraseña en cada acción importante.
- **Falsificación de formularios:** como se ha explicado posteriormente, usar POST proporciona mucha más seguridad que GET para el envío de formularios. No obstante, tampoco es suficiente debido a que es posible la creación de una página web falsa en la que se introduzca un formulario que funcione de la misma forma. Para evitarlo, el servidor genera un valor llamado “CSRF token” que también incorpora en el legítimo y los compara. Sólo en la página original coincidirán y aceptará la petición.

3. Diseño

3.1. Modelo Relacional

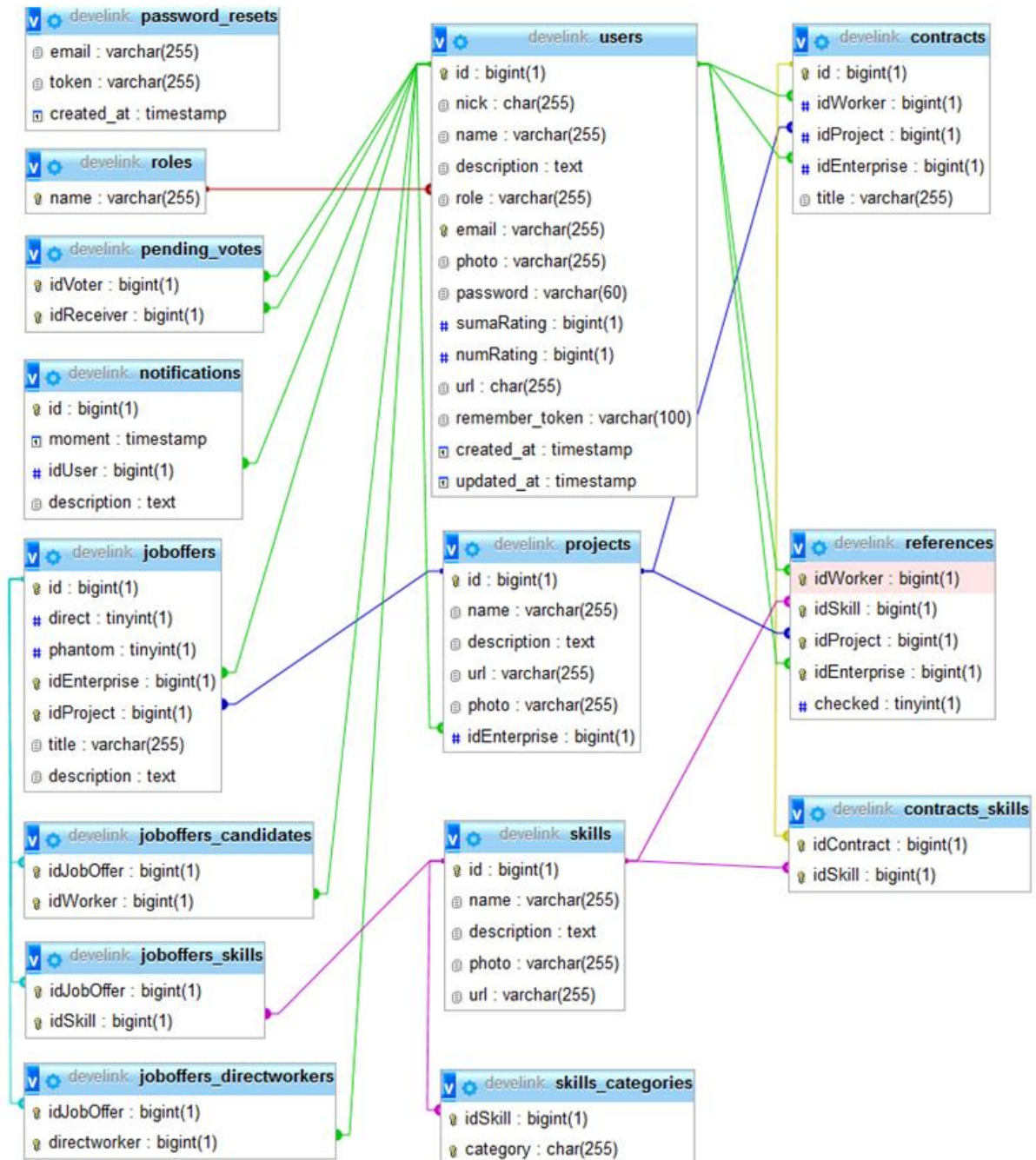


Figura 16. Modelo Relacional de la base de datos usada

Los colores de las referencias son los siguientes: “roles” en rojo, “users” en verde, “projects” en azul, “joboffers” en cian, “skills” en magenta y “contracts” en amarillo.

3.2. Explicación de las tablas

- **password_resets:** utilizada para almacenar los correos electrónicos de los usuarios (tabla users) que han solicitado el recordatorio de contraseña y a la hora que lo realizaron. Es utilizada por el gestor que envía un correo con un enlace para poder cambiarla.
- **roles:** contiene los nombres todos los roles utilizados por la aplicación (el campo “name” es clave primaria). Son referenciados por la tabla users de tal forma que cada usuario tiene un único rol.
- **pending_votes:** votos pendientes de un usuario (votante) a otro (receptor). La aplicación lo limita entre usuarios con roles trabajador y empresa al finalizar un contrato (como se ha explicado en “2.1.1 Diagramas de casos de uso”).
- **notifications:** avisos que un usuario ha recibido y la fecha en que se produjeron.
- **joboffers:** ofertas laborales de la aplicación. Es referenciada por joboffers_candidates, joboffers_skills y joboffers_directworkers
- **joboffers_candidates:** candidatos a una oferta laboral. La aplicación controla que sean usuarios con rol de trabajador.
- **joboffers_skills:** habilidades exigidas en una oferta laboral.
- **joboffers_directworkers:** usuarios con rol trabajador a los que va dirigida una oferta laboral directa.
- **users:** todos los usuarios registrados en la aplicación. Se trata del pilar fundamental de la aplicación ya que es referenciada por la mayoría de tablas, como se puede observar en la figura anterior.
- **projects:** todos los proyectos creados por las empresas (usuarios con rol empresa). Apuntada por joboffers, contracts y references.
- **skills:** habilidades registradas en la aplicación. Apuntada por joboffers_skills, skills_categories, references y contracts_skills.
- **skills_categories:** categorías utilizadas para agrupar habilidades utilizadas en un mismo sector o ámbito.
- **contracts:** contratos realizados a trabajadores por empresas. Está referenciada por contracts_skills.
- **references:** experiencias laborales (referencias). Relacionan empleados, habilidades, proyectos y empresas. Contienen un valor booleano que indica si están validadas (por empresas) o no.

- **contracts_skills:** habilidades que se considerarán aprendidas al completarse los contratos.

A nivel general del funcionamiento de la base de datos, se destaca que la política utilizada es “on delete cascade” y “on update cascade”, es decir, que tanto las modificaciones como las eliminaciones en una tabla, influyen en cascada a todas las que la referencien. De ese modo, se evitan inconsistencias (referencias a valores nulos) y se libera espacio ocupado innecesariamente.

4. Tecnologías usadas

Al inicio del proceso, se decidió qué tecnologías usar. En este apartado se hace una breve explicación de ellas. En el anexo se detallan las configuraciones realizadas.

4.1. Servidor Xampp

Develink se ha desarrollado bajo una arquitectura Xampp (Windows, Apache, MySQL y PHP) ejecutada en el sistema operativo Windows 7 de 32 bits.

Además, se ha comprobado su correcto funcionamiento en Linux. Concretamente, se ha utilizado la distribución Ubuntu 14.04 LTS de 64 bits ya que ofrece una interfaz gráfica muy ligera y al ser Long Term Support (LTS) garantiza un buen soporte a largo plazo (durante 3 años) sin renunciar a la estabilidad requerida en un servidor.

4.2. Servidor Openshift

Para que la aplicación pueda ser probada por el público, ha sido subida a Openshift, propiedad de Red Hat Enterprise.

La cuenta gratuita dispone de 3 pequeños servidores, llamados “small gears”, de 512MB de RAM y 1GB de almacenamiento, más que suficiente para ver su funcionamiento. Uno de sus puntos fuertes es su soporte del protocolo HTTPS, imprescindible si se quiere garantizar un mínimo de seguridad. El principal inconveniente, además de las limitaciones de RAM y almacenamiento, es que si está 48 horas sin recibir peticiones, se para la aplicación y el siguiente acceso es muy lento ya que se tiene que iniciar. A excepción de ese caso, el porcentaje de tiempo de funcionamiento (“uptime”) es prácticamente de un 100%.

La dirección web (url) en la que se ha alojado Develink es la siguiente:

<https://develink-tfgdavid.rhcloud.com>

4.3. Framework PHP

4.3.1. Comparativa

Una de las elecciones más importantes fue el framework a utilizar. Teniendo en cuenta la complejidad de los objetivos marcados, debía:

- Permitir un desarrollo ágil.
- Facilitar un código legible, reutilizable, seguro y que se adapte a nuevos requisitos.

Existen multitud de frameworks con diferentes propósitos. A continuación se analizan algunos de los más conocidos con sus ventajas e inconvenientes. En el siguiente apartado, se comentará la decisión que, finalmente, se tomó.



Medoo

Ventajas	Inconvenientes
Ocupa muy poco espacio (unos 30 Kb)	Demasiado básico
Curva de aprendizaje muy ligera	Por defecto, no proporciona herramientas que faciliten una buena estructuración del código ni su testeo
Admite varios gestores de bases de datos basados en PDO	Sólo recomendable para servicios web sencillos
A pesar de su sencillez, previene de inyecciones SQL	
Buena documentación	



CakePHP

Ventajas	Inconvenientes
Curva de aprendizaje ligera	Hasta hace pocos meses estuvo desfasado: no usaba Composer, la POO no era moderna...
Buena documentación y comunidad	
Ha sufrido recientemente muchas actualizaciones importantes	



Ventajas	Inconvenientes
Curva de aprendizaje ligera	Básico
Muy rápido	Requiere el uso de plugins para muchas funcionalidades básicas
Usa el patrón MVC	Modelo de datos limitado
Buena documentación y gran comunidad de usuarios	



Ventajas	Inconvenientes
Muy completo y moderno	Curva de aprendizaje muy elevada
Buena documentación y comunidad de usuarios	
Ideal para grandes aplicaciones web	



Ventajas	Inconvenientes
Curva de aprendizaje muy ligera	Muy básico
Gestión de rutas muy simple y que permite el uso de expresiones regulares	Por defecto, no proporciona herramientas que faciliten una buena estructuración del código ni su testeo



Ventajas	Inconvenientes
Puntos fuertes de Symfony ya que contiene muchos de sus paquetes	Hay frameworks más fáciles de aprender (con curvas de aprendizaje menores)
Menos curva de aprendizaje que Symfony	
Excelente documentación y comunidad de usuarios	



Ventajas	Inconvenientes
Curva de aprendizaje media	Documentación muy desorganizada
Moderno	



Ventajas	Inconvenientes
Muy rápido (la gran velocidad es debida a que está implementado en C, un lenguaje de más bajo nivel)	Documentación desorganizada
Curva de aprendizaje ligera	Hay que compilar las extensiones PHP
Moderno	Instalación y configuración complicada, especialmente en entornos de producción de servidores compartidos.

4.3.2. Elección: Laravel

Después de analizar los pros y contras anteriores, se descartaron todos los frameworks analizados excepto Laravel.

- **Medoo y Flight:** demasiado simples para un proyecto de tal envergadura ya que no facilitaban la estructuración del código ni su testeo.
- **CodeIgniter:** la estructuración del código que se podía conseguir no era buena ya que el modelo era demasiado simple a expensas de la complejidad (y cantidad de código) del controlador.
- **Symfony:** a pesar de cubrir todas las necesidades, su aprendizaje requería demasiado tiempo (meses) y el disponible para realizar el presente proyecto era limitado.
- **CakePHP:** cuando se inició el proyecto, a pesar de la ligera curva de aprendizaje, veteranía y buena documentación, estaba obsoleto (no usaba Composer, la programación orientada a objetos no era moderna...). No obstante, recientemente ha recibido muchas actualizaciones que han corregido esas carencias y, si su desarrollo sigue a ese ritmo, en un futuro cercano, será una gran opción a tener en cuenta.
- **YiiFramework y Phalcon:** la documentación estaba desorganizada, por lo tanto, complicaba la futura consulta de información. Además, en el caso de Phalcon, hay la dificultad añadida de su instalación y configuración.

A continuación se explica Laravel y se justifica su idoneidad para el proyecto.

Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web. Su objetivo principal es la creación de proyectos PHP de forma elegante y simple. De ese modo, se obtiene rápidamente código fácil de comprender y mantener mejorando la productividad, especialmente en grandes proyectos.

A pesar de ser relativamente joven y de tener una curva de aprendizaje moderada, goza de mucha popularidad debido a su buena documentación y las posibilidades que ofrece. La más destacada es ofrecer una implementación del patrón MVC (Modelo Vista Controlador) intuitiva. También es importante la posibilidad de probar la aplicación usando tests automatizados y que admita tanto configuración de desarrollo (para uso local) como de producción (para uso en servidores públicos).

Por lo tanto, es el framework que mejor se ajusta a las necesidades de la aplicación Develink.

En el proyecto, se usa la versión 5.1 de Laravel y, en Openshift, la 5.0 debido a falta de soporte de la más reciente. La utilización de Composer, que se explica a continuación, corrige los posibles problemas por incompatibilidad entre ambas versiones.

4.4. Composer

Es un gestor de dependencias para PHP que se usa en este proyecto para instalar y mantener actualizado el framework Laravel. Para asegurar portabilidad en las aplicaciones web entre diferentes servidores, trabaja a nivel de proyecto en vez de manejar bibliotecas (libraries) globales del sistema operativo. Su funcionamiento se puede resumir de este modo:

- El proyecto depende de ciertas bibliotecas.
- Algunas de ellas dependen de otras.
- A medida que se van agregando funcionalidades, es probable que se requieran nuevas bibliotecas (y si se eliminan, lo contrario).
- Composer encuentra qué paquetes son necesarios y los instala o actualiza a su última versión (si estaban obsoletos) en la carpeta del proyecto.

4.5. Framework Cliente: Bootstrap

Lo incluyó Laravel hasta la versión 5.0 [31] y está constituido por archivos HTML, CSS y Javascript. Su objetivo es simplificar el desarrollo del aspecto visual de los sitios web de la siguiente forma:

- Adapta los contenidos a todos los tamaños de pantalla (Responsive Design).
- Incluye iconos de libre uso que también son responsive.
- Mejora la accesibilidad de los formularios.
- Detecta la navegación desde un dispositivo móvil y permite cambiar el menú por otro que sea más fácil de usar.
- Procura que la apariencia y funcionamiento del sitio no cambie según el navegador usado.

En el presente proyecto, Develink, se usa debido a las ventajas explicadas anteriormente.

4.6. Gestor de código: Git

Se utiliza Git para mantener un historial de cambios reversible y para subirlo al servidor de Openshift que también lo usa. También es posible agregar, suprimir y descargar archivos mediante gestores FTP como FileZilla pero, en general, no es recomendable porque no disponen del control de versiones de Git y se pueden borrar accidentalmente archivos importantes, provocando que debamos eliminar la aplicación, crear una nueva y volver a subir el proyecto.

5. Planificación

5.1. Fases

El proyecto se planificó en 5 fases: inicial, requisitos, diseño, implementación y final.

Durante la **inicial**, se buscó documentación sobre frameworks que proporcionaran una implementación robusta. Rápidamente se decidió usar Laravel, en lugar de otras alternativas, teniendo en cuenta los motivos explicados en el apartado anterior. También se intentó realizar una captura de requisitos pero, por problemas logísticos debidos a las fechas en las que esta etapa tuvo lugar, se obtuvieron muy pocos y demasiado generales.

Durante la **fase de requisitos**, además de anotar los funcionales, se consideró oportuno alojar la aplicación en un servidor externo llamado Openshift (requisito no funcional), para evitar a los usuarios la instalación y configuración de la aplicación Xampp o similares en sus dispositivos y así facilitar su futuro testeo.

En la **fase de diseño** se capturaron más requisitos, se hicieron diagramas de casos de uso y del modelo de datos, se trabajó bastante en el front-end (apariencia de la web e interacción del usuario) y algo en el back-end (un prototipo simplificado pero funcional de la base de datos de la aplicación y de un controlador que la conectara con la vista).

En la **implementación**, se realizó la mayor parte de la construcción de la aplicación web, se refinó en profundidad el diseño del back-end usando MVC (modelo vista controlador), hubo algún requisito que fue descartado por el tiempo necesario en proporción a su valor de negocio y se introdujeron nuevas funcionalidades que inicialmente no fueron previstas pero que resultaban interesantes. Respecto al front-end, fue aumentando (se crearon más páginas HTML y con más contenido) pero, a diferencia del back-end, lo realizado en las etapas anteriores apenas sufrió modificación. También se inició el presente documento.

La **final** consistió en corregir errores, introducir pequeñas mejoras, realizar pruebas tanto de funcionamiento como de rendimiento del producto obtenido a lo largo del proceso, acabar el redactado de esta memoria y preparar el entorno para realizar la exposición.

5.2. Seguimiento

El proyecto ha tenido una duración estimada total de 135 días. No obstante, se han dedicado en promedio entre 3 y 4 horas diarias ya que, debido a la compaginación con otras asignaturas, la carga de trabajo total ha sido de 36 créditos ECTS en lugar de los 30 habituales y el doble de lo que supondría una dedicación exclusiva al proyecto, 18.

Fase inicial: 4 días (03/07/2015 - 06/07/2015)

- Análisis de requerimientos: 1 día (03/07/2015)
- Documentación acerca de las tecnologías a usar: 3 días (04/07/2015 - 06/07/2015)

Fase de requisitos: 9 días (17/09/2015 - 25/09/2015)

- Planificación: 1 día (17/09/2015)
- Análisis de requerimientos: 1 día (18/09/2015)
- Preparación de los entornos de desarrollo: 2 días (19/09/2015 - 20/09/2015)
- Especificación de funcionalidades: 4 días (21/09/2015 - 24/09/2015)
- Modelo de datos muy básico: 1 día (25/09/2015)

Fase de diseño: 26 días (26/09/2015 - 21/10/2015)

- Frontend: 16 días (26/09/2015 - 11/10/2015)
- Mejora del modelo de datos: 3 días (12/10/2015 - 14/10/2015)
- Análisis de requerimientos: 2 días (15/10/2015 - 16/10/2015)
- Diseño arquitectura: 2 días (17/10/2015 - 18/10/2015)
- Maqueta HTML5 (mejorando frontend previo): 3 días (19/10/2015 - 21/10/2015)

Fase de implementación: 65 días (22/10/2015 - 25/12/2015)

- Desarrollo, pruebas, redactado de la introducción de la memoria y de los aspectos considerados definitivos: 62 días (22/10/2015 - 22/12/2015)
- Gran refactorización de código para corregir errores y mejorar la integración de los requisitos: 3 días (23/12/2015 - 25/12/2015)

Fase final: 31 días (26/12/2015 - 25/01/2016)

- Redactado de la memoria y pruebas: 25 días (26/12/2015 - 19/01/2016)
- Pequeños cambios en la memoria y preparación del entorno para la exposición: 6 días (20/01/2016 - 25/01/2016)

5.3. Diagrama de Gantt

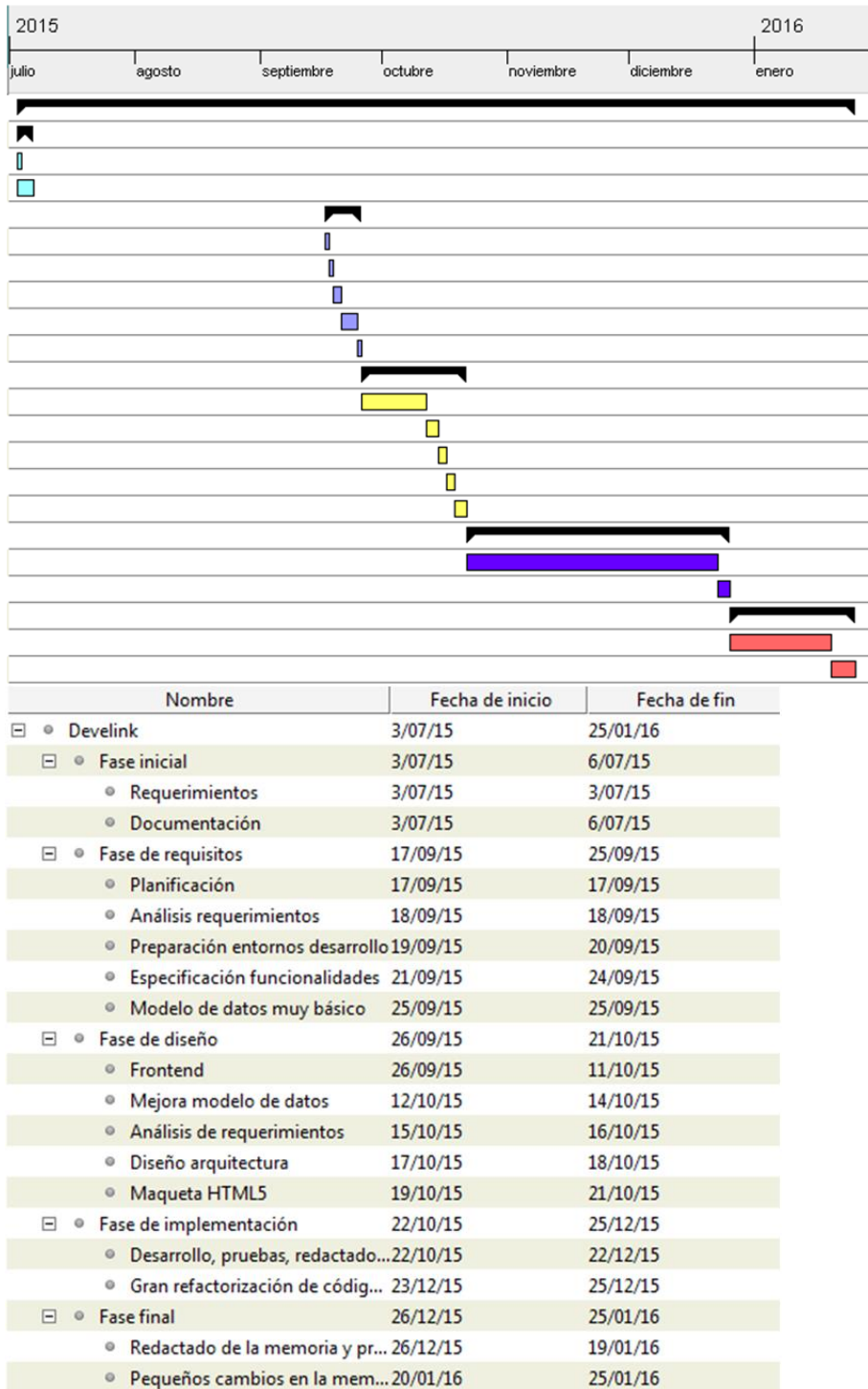


Figura 17. Diagrama de Gantt y calendario de planificación.

6. Arquitectura de Develink

El framework Laravel 5 facilita la construcción de las aplicaciones web. Uno de sus puntos fuertes es la separación de vista, modelo y controlador. No obstante, sólo es un patrón a seguir, la buena estructuración del código es responsabilidad del desarrollador.

6.1. Explicación del MVC usado en Develink

En este apartado se explica de forma simplificada qué sucede internamente cuando se atiende una petición en la aplicación Develink.

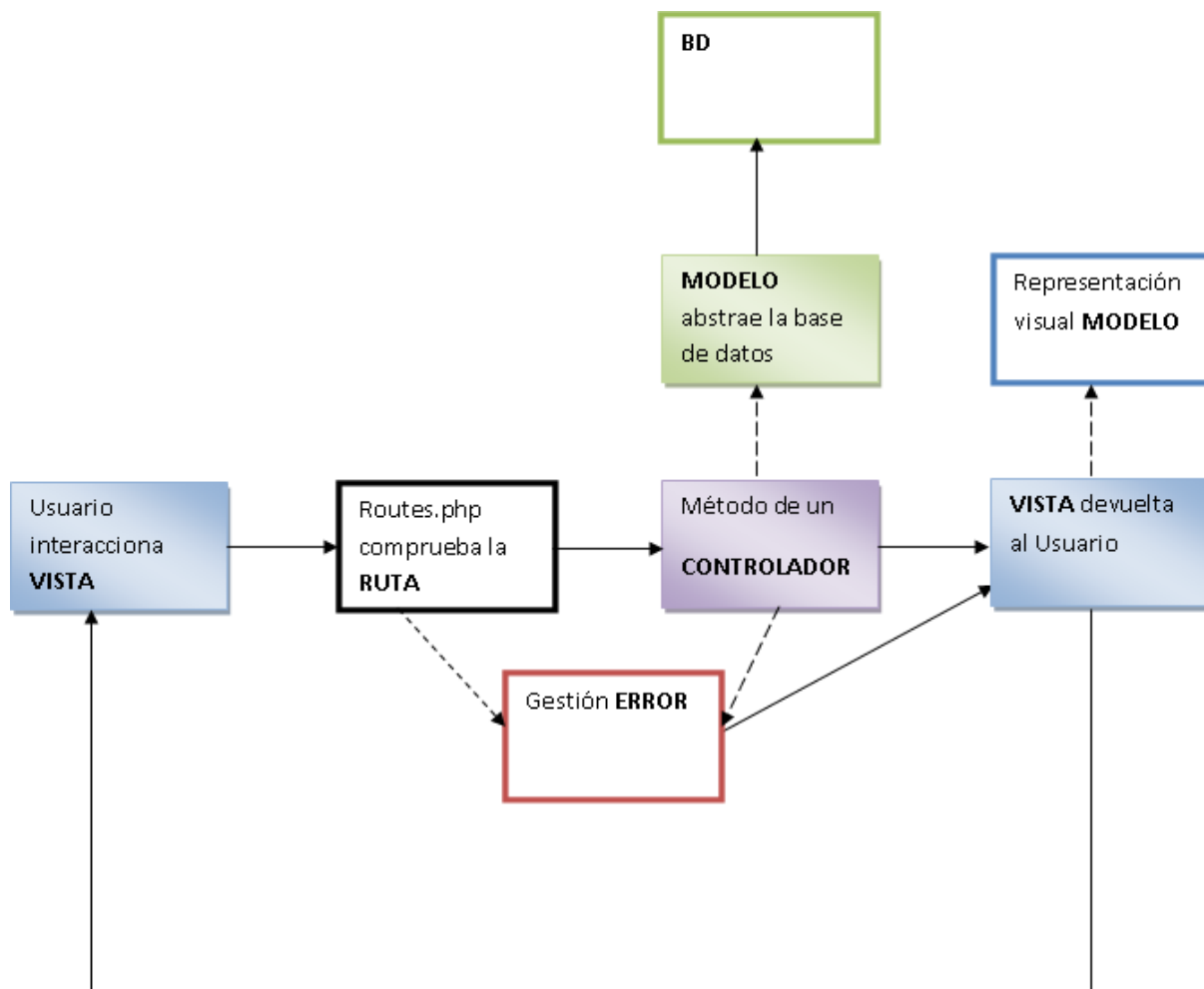


Figura 18. Diagrama de funcionamiento (a grandes rasgos) de Develink

- El usuario interactúa con una vista generando una petición.
- El gestor de rutas (routes.php) se encarga de procesarla y la envía a un método del controlador correspondiente. Si no existe el controlador o la url, muestra el error en una vista.
- El controlador ejecuta la función pedida. En algunos casos, necesitará utilizar el modelo para consultarlo o modificarlo. El modelo es el encargado de acceder (de

forma transparente al programador del controlador) a la base de datos para realizar las consultas pertinentes. Una vez se haya realizado la tarea, se devolverá una vista con el resultado. En caso que la función no exista en el controlador, mostrará otra vista indicando que ha ocurrido un error.

- La vista resultado, si recibió parámetros de controlador (normalmente contenido del modelo), los mostrará. En caso que sean de tipo array o lista los recorrerá y presentará (en html, css...) mediante el uso de Blade Templates [35].
- Finalmente, el usuario puede volver a interactuar con la vista obtenida, comenzando de nuevo el proceso explicado.

6.2. Estructura general creada inicialmente

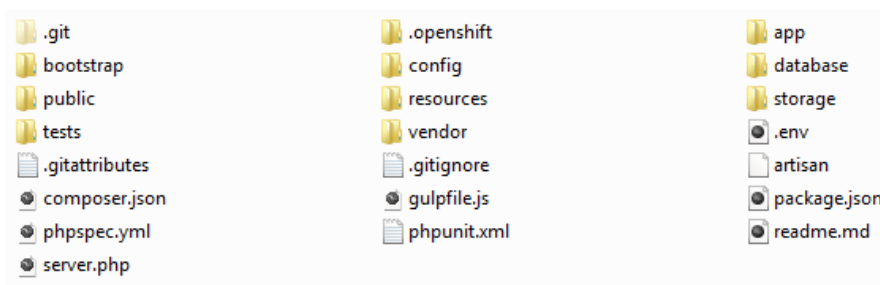


Figura 19. Estructura general de Develink

A continuación se explica cómo se ha estructurado, en líneas generales, Develink. Excepto los archivos y directorios generados tanto por Git como Openshift, todos fueron creados por el framework Laravel al inicio del proyecto:

- **.git:** usada por Git para realizar el control de versiones (commits, push...).
- **.openshift:** contiene archivos para configurar la aplicación del servidor Openshift.
- **app:** incluye los modelos, controladores, middleware y la gestión de rutas (url).
- **bootstrap:** contiene ficheros necesarios para iniciar la aplicación y se definen las rutas de carpetas importantes como app y public. No se debe confundir con el framework de cliente Bootstrap, usado para obtener un diseño web responsive, explicado en “4.Tecnologías Usadas”.
- **config:** se utiliza para configurar los diferentes aspectos de la aplicación (credenciales de la base de datos, caché, sesiones, el servicio de correo electrónico...).
- **database:** se almacenan las migraciones (migrations) y los pobladores (seeds) de las bases de datos.
- **public:** Directorio público que contiene la aplicación de cliente.
- **resources:** contiene las vistas de la aplicación (views), el soporte de idioma (lang) y los archivos necesarios por el framework Bootstrap (en assets).

- **storage:** contiene los datos de caché, sesión y logs.
- **tests:** carpeta con archivos para probar el correcto funcionamiento de la aplicación. Utilizan PHPUnit.
- **vendor:** se almacena el código propio del framework Laravel y sus dependencias. Se gestiona mediante “artisan” y “Composer”.
- **.env:** utilizado para configurar el entorno local de la aplicación web. Concretamente se establece url, base de datos (driver o gestor, puerto, nombre, usuario y contraseña), clave de la aplicación y servicio de correo electrónico (para poder realizar el recordatorio de contraseñas).
- **.gitattributes:** usado por Git para garantizar consistencia entre la modificación de código por parte de múltiples usuarios desde diferentes máquinas y sistemas operativos.
- **.gitignore:** archivo que indica a Git lo que debe ser excluido del repositorio.
- **artisan:** interfaz por línea de comandos para gestionar la aplicación. Incluye un sencillo servidor PHP incorporado.
- **composer.json:** archivo usado por Composer para gestionar los paquetes requeridos por la aplicación.
- **composer.lock:** información sobre el estado de la instalación de Composer y la fecha en la que se actualizaron los paquetes que gestiona. No es recomendable ni necesario editarlo manualmente (de ello se encarga Composer)
- **gulpfile.js:** sistema de construcción (de tareas llamadas Gulp tasks) utilizado para automatizar acciones como la minificación de código Javascript y la compresión de imágenes. En la aplicación Develink no se ha utilizado porque eran pocas tareas y se han realizado manualmente,
- **package.json:** usado para instalar Elixir, una API para definir tareas Gulp de forma sencilla.
- **phpspec.yml:** archivo de configuración de los tests PHPSpec (se diferencia de PHPUnit en que sólo comprueba que el comportamiento de la aplicación sea el esperado sin tener en cuenta sus cambios internos).
- **phpunit.xml:** contiene la configuración general de los tests PHPUnit. La más destacada la ruta donde están situados (por defecto, la carpeta tests).
- **readme.md:** información sobre el framework Laravel. Se puede editar para agregar una descripción del proyecto.
- **server.php:** permite iniciar la aplicación sin un servidor php “real” (por ejemplo, usando el que proporciona “artisan”).

6.3. Estructura propia de Develink

En este apartado se detallan los archivos y carpetas más relevantes que se han creado o modificado durante el desarrollo de Develink.

6.3.1. Public

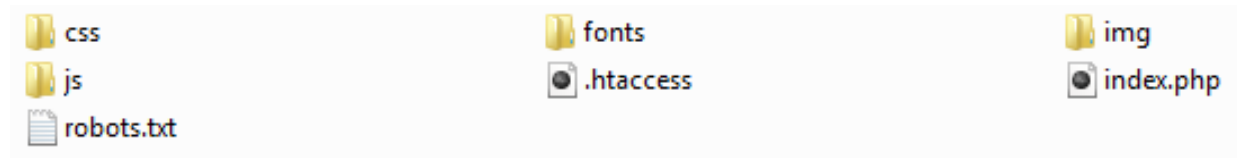


Figura 20. Carpeta public Develink

Contiene los siguientes elementos:

- **css:** carpeta con hojas de estilos. Además de los propios del framework Bootstrap, contiene la de la aplicación (sin minificar).
- **fonts:** incluye fuentes para que la apariencia del texto de la página sea independiente del navegador y del sistema operativo.
- **img:** imágenes usadas por la aplicación
- **js:** incluye archivos Javascript (el propio de la aplicación, el de Bootstrap y de la library JQuery [32]).
- **.htaccess:** fichero utilizado para configurar el servidor Apache [33]. Además de los valores por defecto, se ha establecido una caché (con “set Cache-Control ‘max-age’”) de un año para ficheros estáticos como imágenes y fuentes. Para los javascript y css, sólo una semana ya que deben ser modificados con mucha más frecuencia. De esa manera, supone menos sobrecarga para el servidor y más velocidad en posteriores accesos para el cliente (también un consumo menor para aquellos que tengan contratada una tarifa de datos móvil).
- **index.php:** controlador del frontend, responsable de cargar y ejecutar la aplicación.
- **robots.txt:** Utilizado para indicar a los robots [34] (también llamados spiders o rastreadores automáticos) qué páginas deben consultar y cuáles no. Hay que tener en cuenta que se trata de una información meramente consultiva, es decir, sólo la obedecerán aquellos que sigan unas buenas prácticas. También se podría usar para agregar un enlace a un sitemap (mapa del sitio web).

6.3.2. App

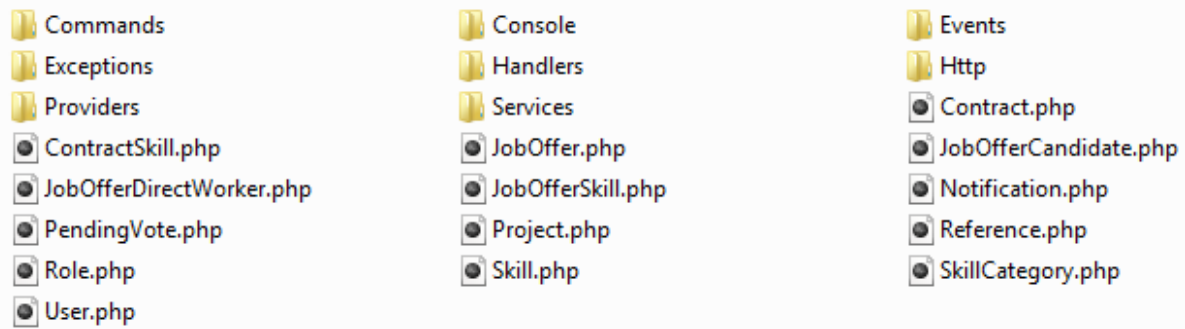


Figura 21. Carpeta app Develink

Los elementos más destacados son los siguientes:

- **Http:** contiene controladores, middleware, las rutas...
- **archivos .php** (Contract, ContractSkill...): contienen el modelo de la aplicación. Se utilizan para que el uso de la base de datos sea transparente al programador.

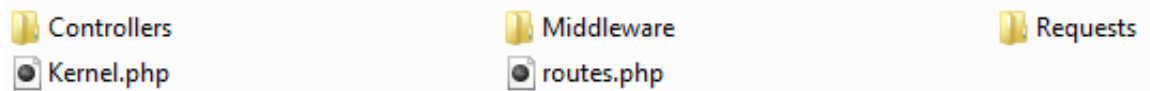


Figura 22. Carpeta app/Http

El contenido más relevante de la carpeta Http es el siguiente:

- **Controllers:** controladores, se explicarán posteriormente.
- **Middleware:** contiene mecanismos que filtran las peticiones http en una aplicación. Por defecto se controla la autenticación y se verifica el CSRF token. Además, se ha creado otro middleware, llamado “HttpsProtocol.php”, que fuerza la conexión segura (HTTPS) de la siguiente forma:
 - Si la conexión no es segura y el entorno es de producción, se redirige a la versión encriptada (segura). El controlar que el entorno sea de producción es importante porque, por defecto, en localhost (entorno de test) no está activado el protocolo HTTPS y, para que la aplicación funcionara, haría falta un certificado “autofirmado”.
 - En caso contrario, se acepta la petición.
- **routes.php:** se definen cuáles son los controladores de autenticación y recuperación de contraseña. Además, se establecen las rutas y cómo se atienden sus peticiones.

```
Route::get('home', 'IdentifiedUserController@home')
```

Figura 23. Ejemplo de ruta Develink: Route:: método (url, controlador@función del controlador)

Una vez abierta la carpeta Controllers, se puede ver lo siguiente:

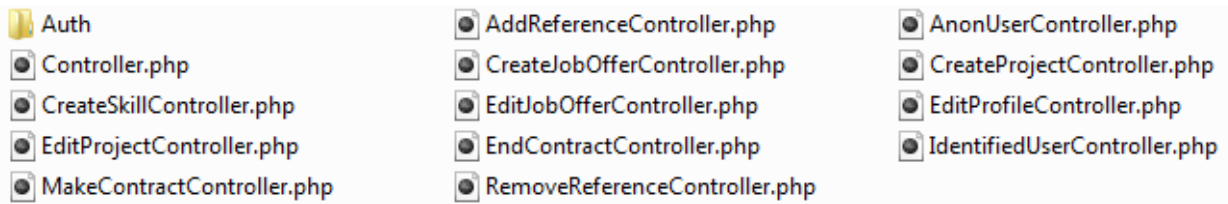


Figura 24. Carpeta app/Http/Controllers

Lo más destacable:

- **Auth:** contiene controladores que gestionan la autenticación y la recuperación de contraseña
- **Controller.php:** clase abstracta de la que heredan el resto de controladores
- **AnonUserController.php:** gestiona todas las acciones del usuario invitado (anónimo, que no ha iniciado sesión).
- **IdentifiedUserController.php:** gestiona todas las acciones que requieren estar identificado.

Se omite la descripción del resto de controladores, debido a que sus nombres indican claramente lo que realizan.

6.3.3. Resources

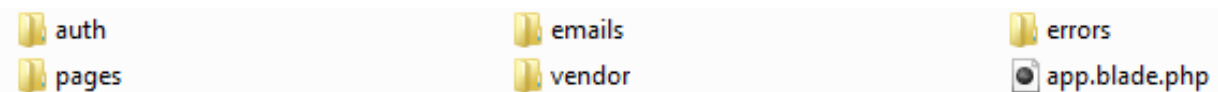


Figura 25. Carpeta resources/views de Develink

A continuación se explican las vistas de la aplicación:

- **auth:** incluye las vistas que muestran los formularios de inicio de sesión, registro...
- **emails:** contiene la vista que permite la recuperación de la contraseña.
- **errors:** páginas de error. La más importante la del error 404 (página no encontrada).
- **pages:** páginas más específicas del proyecto Develink. Se ha realizado una división en dos carpetas, show (las que muestran información) y edit (las que contienen elementos como formularios, que permiten modificar el contenido de la aplicación).
- **vendor:** carpeta vacía, usada por Laravel para mantener consistencia con la carpeta del mismo nombre (vendor) de la raíz del proyecto.
- **app.blade.php:** página base o plantilla de todas las de la aplicación. Incluye las barras de navegación y menús. También las referencias a archivos javascript, favicon, css...

Abriendo la carpeta “show” de pages, se encuentran las siguientes vistas:

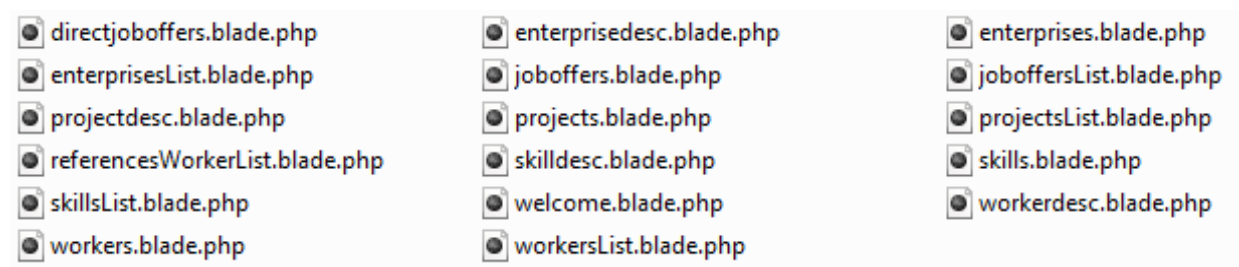


Figura 26. Carpeta resources/views/pages/show de Develink

Los nombres de los archivos describen de forma clara qué información muestran. Los que contienen la palabra “List”, son la representación visual del modelo y se incluyen dentro de otras vistas (como se explicó en la figura 18).

El contenido de la carpeta “edit” es el siguiente:

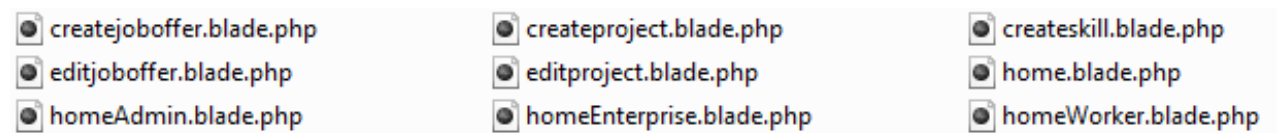


Figura 27. Carpeta resources/views/pages/edit de Develink

Es destacable el funcionamiento de la página home. Hay una parte común a todos los usuarios que permite editar el perfil (home.blade.php) y, en función del rol, se incluye la específica (homeWorker.blade.php, homeEnterprise.blade.php o homeAdmin.blade.php).

7. Pruebas de funcionamiento y rendimiento

Durante el desarrollo de la aplicación web, se han realizado pruebas de funcionamiento y de rendimiento para ver su calidad y mejorarla. El navegador web usado ha sido Firefox 43.

7.1. Revisión de la base de datos

Utilizando un programa del servidor, PHPMYAdmin, se han analizado (Analyze) y revisado (Check) las tablas que conforman la base de datos, con satisfactorio resultado.

Su consulta se ejecutó con éxito.

```
CHECK TABLE `contracts`, `contracts_skills`, `joboffers`, `joboffers_candidates`,
`joboffers_directworkers`, `joboffers_skills`, `notifications`, `password_resets`, `pending_votes`,
`projects`, `references`, `roles`, `skills`, `skills_categories`, `users`
```

[En línea] [Editar] [Crear código PHP]

+ Opciones

Table	Op	Msg_type	Msg_text
develink.contracts	check	status	OK
develink.contracts_skills	check	status	OK
develink.joboffers	check	status	OK
develink.joboffers_candidates	check	status	OK
develink.joboffers_directworkers	check	status	OK
develink.joboffers_skills	check	status	OK
develink.notifications	check	status	OK
develink.password_resets	check	status	OK
develink.pending_votes	check	status	OK
develink.projects	check	status	OK
develink.references	check	status	OK
develink.roles	check	status	OK
develink.skills	check	status	OK
develink.skills_categories	check	status	OK
develink.users	check	status	OK

Figura 28. Revisión de tablas mediante PHPMYAdmin

7.2. Validación HTML5

Se ha usado una herramienta web del W3C (World Wide Web Consortium) para validar la sintaxis [26]. Inicialmente encontró algún pequeño error (tags sin cerrar correctamente) y se solventó siguiendo los consejos indicados.

Showing results for <https://develink-tfgdavid.rhcloud.com/>

Checker Input

Show source outline image report

Check by address

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

1. Info **The Content-Type was** `text/html`. **Using the HTML parser.**
2. Info **Using the schema for HTML with SVG 1.1, MathML 3.0, RDFa 1.1, and ITS 2.0 support.**

Document checking completed. No errors or warnings to show.

Total execution time 699 milliseconds.

Figura 29. Validación HTML5 mediante herramienta de W3C

7.3. Accesibilidad y usabilidad

Como se indicó en el apartado 2.2 (Requisitos no funcionales), la accesibilidad y la usabilidad son una prioridad en toda aplicación y, por lo tanto, se ha considerado importante su análisis. Debido a la falta de tiempo, no se ha podido realizar un test con usuarios reales. Las dos opciones restantes viables eran la autoevaluación mediante heurísticas como las de Nielsen [44] o recurrir a herramientas online. Por objetividad, se decidió la segunda.

Wave [27] y Achecker [28] son dos servicios web que, a pesar de estar más centrados en la accesibilidad, son de gran utilidad. El más intuitivo de ellos es Wave ya que, además de las líneas con incidencias, muestra una vista previa de la página analizada con flags.

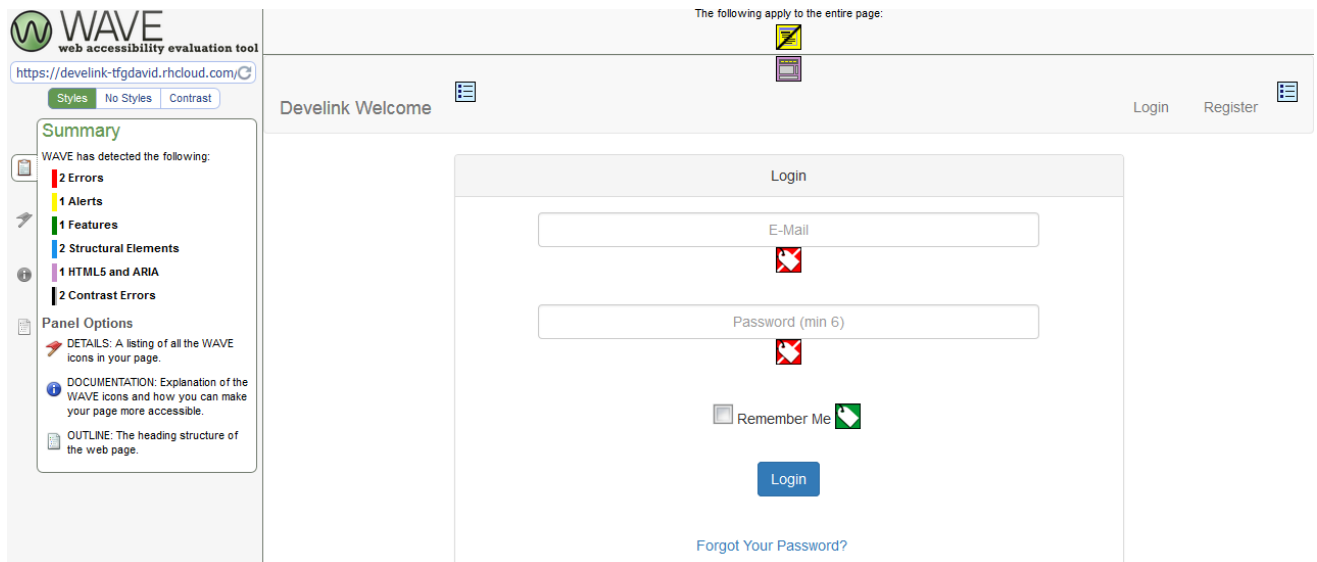


Figura 30. Test de accesibilidad de Wave aplicado al formulario de Login de Develink

De la figura anterior, es destacable positivamente la utilización del elemento HTML label para facilitar la selección de la función de Recordar sesión de usuario (“Remember me”). En cambio, también se detectan aspectos a mejorar como la falta de accesibilidad de los campos de texto para introducir el correo electrónico y la contraseña. El resto de avisos son consejos sobre cómo estructurar la página y la navegación.

7.4. Rendimiento

Para evaluar el rendimiento se ha utilizado PageSpeed Insights [29] de Google. Además de la velocidad, incorpora análisis sobre optimización móvil y experiencia de usuario.

En el primer análisis realizado detectó los siguientes aspectos a mejorar en cuanto a velocidad:

- Establecer una caché de navegador con elementos que cambian con poca frecuencia (como imágenes). De ese modo, el navegador web (cliente), los almacena durante un periodo de tiempo para no tenerlos que descargar cada vez que se abra la página. Aporta la gran ventaja de mejorar el tiempo de carga y un ahorro para los usuarios que usen tarifas de datos móviles. En la aplicación Develink, se decidió aplicar una caché de una semana para los scripts y de un año para imágenes. Esa configuración se establece modificando el archivo público “.htaccess” del servidor.
- Posponer la ejecución de Javascript hasta que se haya cargado la página. Se solventa poniendo los scripts al final del “body” y que el código inicie con el evento “window.onload” o, si se usa JQuery, “\$(document).ready(function)”.

- Minificar CSS, HTML, Javascript: consiste en reducir, manualmente con editores de texto o mediante herramientas especializadas, el tamaño de esos archivos eliminando espacios, saltos de línea y comentarios. La gran desventaja es la falta de legibilidad y, por tanto, el mantenimiento de código. Por ese motivo, es recomendable trabajar con la versión expandida y subir una copia minificada al servidor.
- Evitar el bloqueo que provoca la carga de los archivos CSS: si son de poco tamaño, recomienda su incorporación directa minificada al inicio de HTML.
- Optimizar imágenes: reducir su tamaño (mejorar su compresión). Existe software que realiza esa tarea.

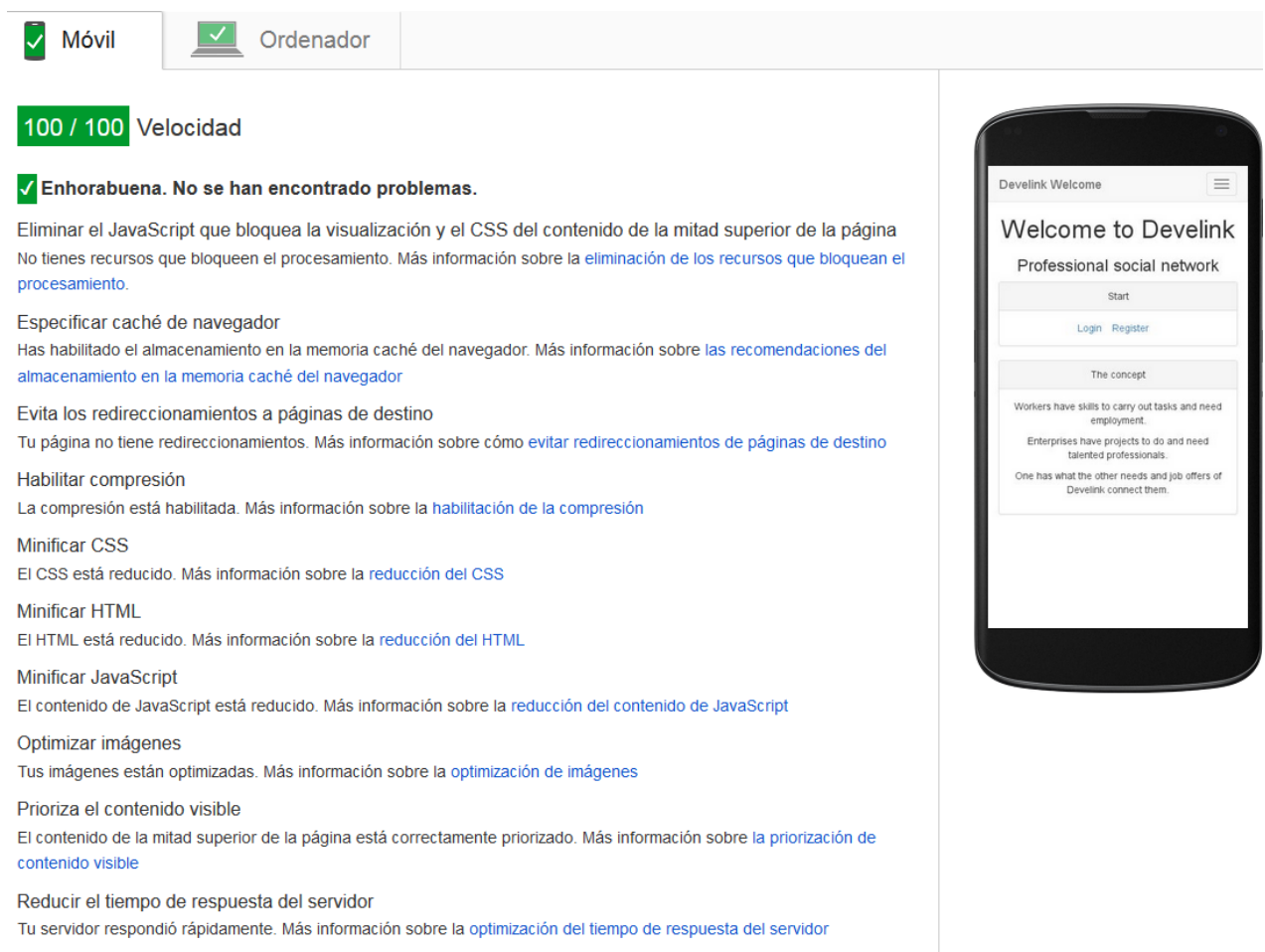


Figura 31. Test de velocidad de PageSpeed Insights una vez aplicadas las mejoras

Respecto a la experiencia de usuario no hubo problemas ya que al usar Bootstrap y seguir sus directrices se cumplían las siguientes reglas:

- Configurar la ventana gráfica.
- Adaptar el contenido a la ventana gráfica.
- Aplicar el tamaño adecuado a los botones táctiles.
- Utilizar tamaños de fuente que se puedan leer.

El consejo de no utilizar plugins se siguió desde el principio ya que actualmente, con las funcionalidades ofrecidas por HTML5 junto a Javascript, no es necesario recurrir a Flash ni a applets Java.

8. Conclusiones y propuestas de mejora

8.1. Conclusiones

El proyecto partía de un objetivo muy ambicioso: construir una red social profesional totalmente funcional con características innovadoras. Por lo tanto, requería un proceso con los siguientes pasos:

- Estudio de la futura competencia, para poder ver qué servicios ofrece y qué aspectos se pueden mejorar.
- Análisis en profundidad de los requisitos obtenidos en la etapa anterior, de propuestas recibidas y de ideas propias.
- Planificación exhaustiva y realista de las tareas a realizar.
- Un diseño flexible para adaptarse a futuros cambios.
- Uso de tecnologías y herramientas adecuadas que proporcionaran agilidad en la implementación.
- Pruebas para evaluar la calidad del producto y obtener consejos para mejorarla.

Después de realizarlo, se puede decir que los objetivos principales están cumplidos. Sin embargo, la aplicación no puede darse por totalmente acabada por los motivos expuestos a continuación:

- Han quedado funcionalidades secundarias pendientes a implementar. Serán concretadas en el siguiente apartado.
- Para que un proyecto de tal magnitud triunfe, debe ser llevado a cabo por un equipo multidisciplinar, competente y cohesionado en lugar de una sola persona. Una individualidad, aunque tuviera muchos conocimientos y fueran transversales, no podría dominar todos los aspectos de su desarrollo ni tampoco lograr avances significativos en breves periodos de tiempo. Una posible división de tareas de ese equipo (multidisciplinar) sería la siguiente:
 - Diseño gráfico.
 - Gestión de la base de datos.
 - Vistas (blade templates, html, css...).
 - Cliente Javascript.
 - Servidor php (controlador, modelo, rutas...).
 - Usabilidad.
 - Administrar la página desde una interfaz web (sin necesidad de conocimientos técnicos).

- La evolución de las nuevas tecnologías es tal que nunca se puede dar por acabada una aplicación ya que, aunque sea de gran calidad y utilidad, si no se actualiza, será superada por las más modernas e innovadoras, quedando en el olvido.

Personalmente, me ha aportado el conocimiento de un framework PHP y la familiarización con grandes proyectos. Considero que ambas experiencias, junto a las obtenidas a lo largo del grado, serán de gran utilidad para mi futuro profesional.

8.2. Propuestas de mejora

A continuación se exponen ideas útiles para incorporar en el proyecto si se continuara su desarrollo.

8.2.1. Validación de empresas

Como se mencionó en el apartado “1.5 Objetivos de la propuesta”, es fundamental controlar el proceso de creación de una cuenta con rol de empresa para que Develink tenga una aplicación real. Debido a su complejidad, no es recomendable que sea automático y, por lo tanto, debería haber intervención por parte de un administrador. La forma que se propone es:

- Usar CIF y correo electrónico de la empresa para demostrar existencia.
- Buscar en internet que coincida con institucionales como primera criba de suplantación.
- Confirmación del correo electrónico (automático con servicios de correo como Mailgun o contactando) para asegurar legitimidad.

8.2.2. Sistema de certificación de estudios

En muchos sectores profesionales, es más importante la titulación académica que la experiencia laboral. Por lo tanto, sería interesante poder automatizar el proceso permitiendo la subida de documentos escaneados que acrediten esos conocimientos. No obstante, plantea una serie de problemas a resolver:

- La verificación de los documentos. Si fuera manual haría falta mucho personal para comprobarlos y habría que confiar en la honestidad de quien lo revisa (que no dieran mal uso a la información). La alternativa sería recurrir al reconocimiento óptico de caracteres (OCR) pero es complejo y tiene limitaciones (no es infalible).
- Cumplir la Ley de Protección de Datos.
- Hay riesgos de ataques externos. Los más comunes:
 - Denegación de servicio, subiéndolos masivamente.

- Virus (que estén infectados).
- Robo de información aprovechando vulnerabilidades.

8.2.3. Mejora de las ofertas laborales directas

Incorporar a las ofertas laborales directas la opción de buscar empleados por características adicionales como edad, nacionalidad, disponibilidad horaria, intención de viajar por cuestiones de trabajo...

8.2.4. Recomendación de usuarios

Esta propuesta consiste en simular las recomendaciones de boca en boca que se han hecho desde tiempos inmemoriales a empresarios sobre personas conocidas que podrían ocupar un puesto de trabajo que ofrecen. El principal riesgo sería el spam pero se podría evitar con las siguientes restricciones:

- Establecer una valoración de las personas que recomiendan teniendo en cuenta si los contratos de sus candidatos sugeridos han finalizado de forma satisfactoria o no para las empresas.
- Limitar el número de recomendaciones que pueden realizar según la valoración anterior (reputación).

8.2.5. Ampliar la gestión de proyectos

Hacer una diferenciación entre proyectos finalizados, activos y futuros mejoraría su gestión en cuanto a usabilidad ya que actualmente, a simple vista (sin abrir todas las ofertas que contiene), no se puede saber su estado exacto. Además, permitiría tener un histórico de conjuntos de ofertas laborales reutilizable.

8.2.6. Contratos con fecha de caducidad

Además de la finalización manual de contratos, podría implementarse la opción de incluir una fecha de caducidad de tal forma que, cuando el periodo expira, haya la opción de renovarlo. Su realización supondría una automatización de los contratos laborales temporales.

8.2.7. Implementación del panel de administrador

Además de las funciones específicas de validación de empresas y certificación de estudios, el administrador debería poder realizar el resto de casos de uso expuestos en el apartado 2.1.1.

8.2.8. Chat interno

Sería interesante crear un chat. De esa manera, se podrían comunicar empresas con trabajadores, característica especialmente útil después de una oferta laboral directa. Además, si se le agregara videoconferencia, posibilitaría la realización de entrevistas laborales dentro de la propia aplicación. A pesar de su gran dificultad (requeriría una gran inversión si no se recurre a herramientas externas o plugins) si se lograra, supondría, junto a la realización de las propuestas anteriores, una simplificación sin precedentes de la gestión de los recursos humanos de una empresa y, para los candidatos, más probabilidades de encontrar empleo.

9. Bibliografía

- [1] "InfoJobs" [Last Accessed: 10/01/16]
<https://www.infojobs.net>
- [2] "LinkedIn" [Last Accessed: 10/01/16]
<https://es.linkedin.com>
- [3] "Adecco, Infoempleo - Redes Sociales y Mercado de Trabajo" [Last Accessed: 10/01/16]
[www.adecco.es/ data/NotasPrensa/pdf/642.pdf](http://www.adecco.es/data/NotasPrensa/pdf/642.pdf)
- [4] "Viadeo" [Last Accessed: 10/01/16]
<http://es.viadeo.com/es>
- [5] "Xing" [Last Accessed: 10/01/16]
<https://www.xing.com/es>
- [6] "Stack Overflow Careers" [Last Accessed: 10/01/16]
<http://careers.stackoverflow.com>
- [7] "Open Web Application Security Project - PHP" [Last Accessed: 10/01/16]
https://www.owasp.org/index.php/PHP_Security_Cheat_Sheet
- [8] "Apache Friends - Xampp" [Last Accessed: 10/01/16]
<https://www.apachefriends.org/es/index.html>
- [9] "OpenShift Developers" [Last Accessed: 10/01/16]
<https://developers.openshift.com/>
- [10] "Genbeta Dev Frameworks PHP" [Last Accessed: 10/01/16]
genbetadev.com/frameworks/y-6-frameworks-php-mas-que-te-haran-la-vida-mas-simple
- [11] "Laravel 5.1 - Docs" [Last Accessed: 10/01/16]
<https://laravel.com/docs/5.1/>
- [12] "Symfony" [Last Accessed: 10/01/16]
<https://symfony.com/doc/current/index.html>
- [13] "Medoo" [Last Accessed: 10/01/16]
<http://medoo.in/doc>
- [14] "Flight" [Last Accessed: 10/01/16]
<http://flightphp.com/learn>
- [15] "CodeIgniter" [Last Accessed: 10/01/16]
<https://www.codeigniter.com/docs>
- [16] CakePHP API [Last Accessed: 10/01/16]
<http://api.cakephp.org>
- [17] "Best PHP Frameworks" [Last Accessed: 10/01/16]
<http://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/>

- [18] "Yii PHP Framework" [Last Accessed: 10/01/16]
<http://www.yiiframework.com/>
- [19] "Phalcon PHP" [Last Accessed: 10/01/16]
<https://phalconphp.com/es/>
- [20] "Composer" [Last Accessed: 10/01/16]
<https://getcomposer.org/>
- [21] "Bootstrap" [Last Accessed: 10/01/16]
<http://getbootstrap.com/>
- [22] "EGit" [Last Accessed: 10/01/16]
<http://www.eclipse.org/egit/>
- [23] "Modelio" [Last Accessed: 10/01/16]
<https://www.modelio.org/>
- [24] "GanttProject" [Last Accessed: 10/01/16]
<http://www.ganttproject.biz/>
- [25] "PHPMyAdmin" [Last Accessed: 10/01/16]
<http://docs.phpmyadmin.net/es/latest/>
- [26] "W3C Validator" [Last Accessed: 10/01/16]
<https://validator.w3.org/>
- [27] "Wave - Web Accessibility Tool" [Last Accessed: 10/01/16]
<http://wave.webaim.org/>
- [28] "AChecker" [Last Accessed: 10/01/16]
<http://achecker.ca/checker/index.php>
- [29] "PageSpeed Insights" [Last Accessed: 10/01/16]
<https://developers.google.com/speed/pagespeed/insights/>
- [30] "Secure Middleware Laravel" [Last Accessed: 10/01/16]
<https://laracasts.com/discuss/channels/tips/secure-middleware-for-laravel-5>
- [31] "Integrar Bootstrap en Laravel 5.1" [Last Accessed: 12/01/16]
<https://styde.net/integrar-bootstrap-3-en-laravel-5-1/>
- [32] "jQuery" [Last Accessed: 15/01/16]
<http://api.jquery.com/>
- [33] ".htaccess, fichero de configuración del servidor Apache" [Last Accessed: 15/01/16]
<https://httpd.apache.org/docs/2.4/howto/htaccess.html>
- [34] "robots.txt, Google Support" [Last Accessed: 15/01/16]
<https://support.google.com/webmasters/answer/6062608?hl=es>
- [35] "Blade Templates Laravel" [Last Accessed: 15/01/16]
<https://laravel.com/docs/5.0/templates>

- [36] "Deploying Laravel on Openshift" [Last Accessed: 18/01/16]
<https://developers.openshift.com/en/php-framework-laravel.html>
- [37] "Openshift - Laravel QuickStart" [Last Accessed: 18/01/16]
<https://hub.openshift.com/quickstarts/115-laravel-5-0>
- [38] "Openshift Login and Register" [Last Accessed: 18/01/16]
<https://openshift.redhat.com/app/login>
- [39] "Openshift Client Tools (rhc)" [Last Accessed: 18/01/16]
<https://developers.openshift.com/en/managing-client-tools.html>
- [40] "Openshift Remote Connection (SSH)" [Last Accessed: 18/01/16]
<https://developers.openshift.com/en/managing-remote-connection.html#keys>
- [41] "Installing Laravel" [Last Accessed: 18/01/16]
<https://laravel.com/docs/5.2#installation>
- [42] "Mail - Laravel" [Last Accessed: 18/01/16]
<https://laravel.com/docs/5.2/mail>
- [43] "Mailgun – QuickStart" [Last Accessed: 18/01/16]
<https://documentation.mailgun.com/quickstart.html>
- [44] "Usability Heuristics of Nielsen" [Last Accessed: 25/01/16]
<https://www.nngroup.com/articles/ten-usability-heuristics/>
- [45] "Anatomy of Facebook- Four Degrees of Separation" [Last Accessed: 25/01/16]
<https://www.facebook.com/notes/facebook-data-team/anatomy-of-facebook/10150388519243859>
- [46] "Levenshtein function - PHP documentation" [Last Accessed: 25/01/16]
<http://php.net/manual/es/function levenshtein.php>

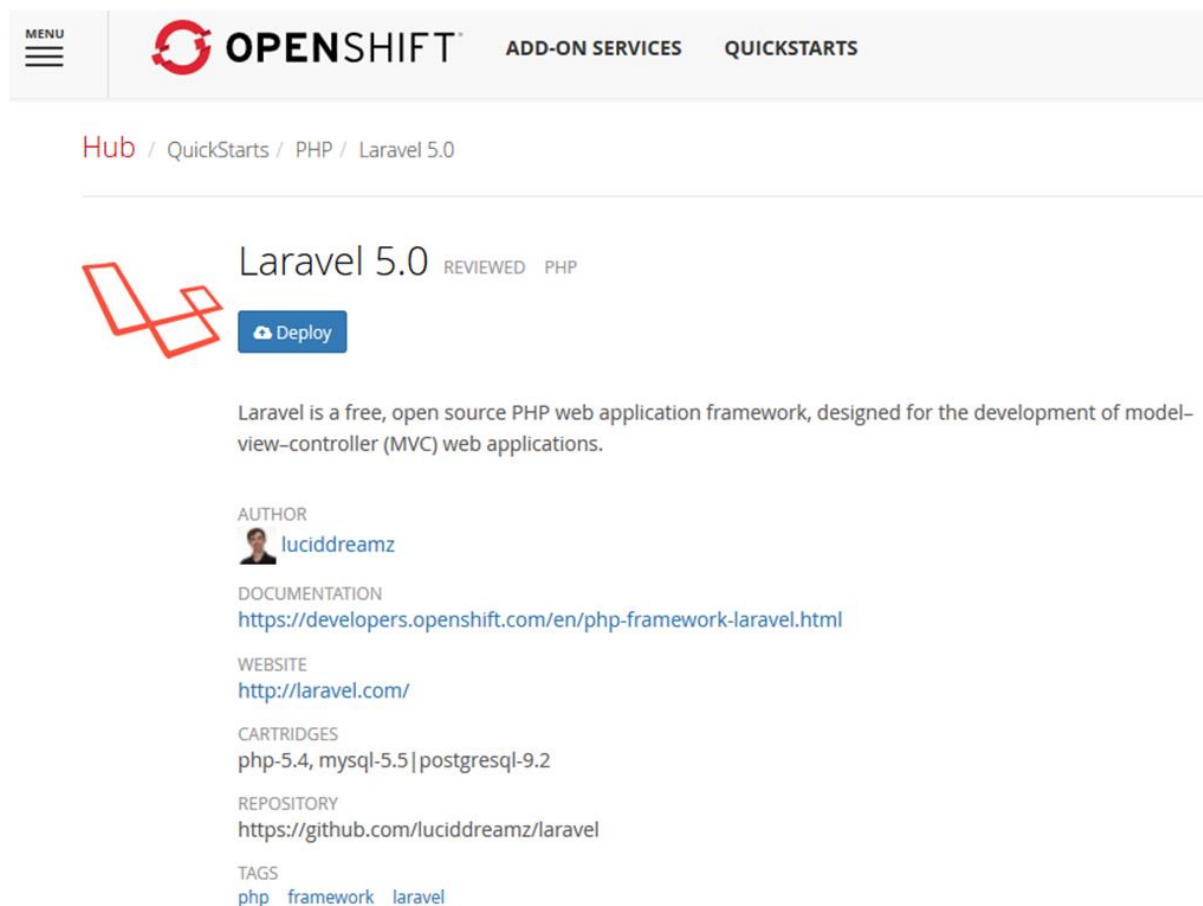
10. Anexo

10.1. Aplicación Laravel en OpenShift

10.1.1. Creación

El servidor OpenShift contiene una documentación algo escasa para instalar manualmente Laravel [36]. Por suerte, ofrece un cómodo “QuickStart Laravel 5.0” [37] que realiza de forma automática gran parte de la configuración. El único problema es que no se trata de la última versión (pero es bastante reciente). A continuación se explican los pasos a seguir:

- Iniciar sesión en OpenShift [38].
- Ir a la página del QuickStart y pulsar el botón “Deploy”.



The screenshot shows the OpenShift QuickStart interface for Laravel 5.0. At the top, there is a navigation bar with 'MENU', 'OPENSHIFT', 'ADD-ON SERVICES', and 'QUICKSTARTS'. Below this, a breadcrumb trail reads 'Hub / QuickStarts / PHP / Laravel 5.0'. The main content area features the Laravel logo, the title 'Laravel 5.0' with 'REVIEWED' and 'PHP' tags, and a blue 'Deploy' button. A descriptive paragraph states: 'Laravel is a free, open source PHP web application framework, designed for the development of model-view-controller (MVC) web applications.' Below this, several metadata fields are listed: 'AUTHOR' (luciddreamz), 'DOCUMENTATION' (https://developers.openshift.com/en/php-framework-laravel.html), 'WEBSITE' (http://laravel.com/), 'CARTRIDGES' (php-5.4, mysql-5.5 | postgresql-9.2), 'REPOSITORY' (https://github.com/luciddreamz/laravel), and 'TAGS' (php, framework, laravel).

Figura 32. QuickStart Laravel 5.0. Fecha: 18/01/2016

- Se mostrará una página con valores por defecto.
El más importante es el nombre de la aplicación que está en “Public URL” y que por defecto es “php” (se recomienda cambiarlo por el que se considere oportuno).

También es relevante seleccionar si se quiere una aplicación escalable o no. Lo ideal es que fuera escalable pero hay que tener en cuenta que, si se quiere añadir posteriormente phpMyAdmin [25], (que sirve para configurar visualmente y de forma fácil la base de datos) genera problemas. Por lo tanto, la elección dependerá de si se quiere usar o no la herramienta. Finalmente, se usa el botón “Create Application”.

Importante: La creación puede durar algunos minutos.

The screenshot shows the 'Configure the application' step in the OpenShift console. At the top, there are three numbered steps: 1. Choose a type of application, 2. Configure the application (current step), and 3. Next steps.

The configuration options are as follows:

- Based On:** Laravel 5.0 Quickstart. Description: Laravel is a free, open source PHP web application framework, designed for the development of model-view-controller (MVC) web applications. Includes links for 'Learn more', 'Community created', and 'Does not receive automatic security updates'.
- Public URL:** http:// -tfgdavid.rhcloud.com. Note: OpenShift will automatically register this domain name for your application. You can add your own domain name later.
- Source Code:** . Note: Your application will start with an exact copy of the code and configuration provided in this Git repository. OpenShift may expect certain files to exist in certain directories, which may require you to update your repository after creation.
- Gears:** small. Note: Gears are the application containers running your code. For most applications, the small gear size provides plenty of resources. You can also upgrade your plan to get access to more gear sizes.
- Cartridges:** PHP 5.4 and . Note: Applications are composed of cartridges - each of which exposes a service or capability to your code. All applications must have a web cartridge.
- Scaling:** . Note: OpenShift automatically routes web requests to your web gear. If you allow your application to scale, we'll set up a load balancer and allocate more gears to handle traffic as you need it.
- Region:**
 - No preference
 - aws-us-east-1 (All gear sizes can be deployed to the US Region.)
 - aws-eu-west-1 (WARNING: Small gears cannot be deployed to this region. Only production gears can be deployed to the EU Region (small,highcpu, medium, and large).)
 - aws-ap-southeast-2 (WARNING: This region is reserved for Dedicated Node Service)
 - aws-us-west-1 (WARNING: This region is reserved for Dedicated Node Service)
 - aws-eu-central-1 (WARNING: This region is reserved for Dedicated Node Service)

Gears within your application will run on servers in the specified region.

At the bottom, there are buttons for 'Back', 'Create Application', and '+1'.

Figura 33. Crear aplicación Laravel 5.0 en OpenShift. Fecha: 18/01/2016

- Una vez creada, se muestra información sobre credenciales de acceso a la base de datos y para gestionar el código mediante un cliente Git. Aunque se puede obtener posteriormente, es recomendable apuntarla en un lugar seguro.

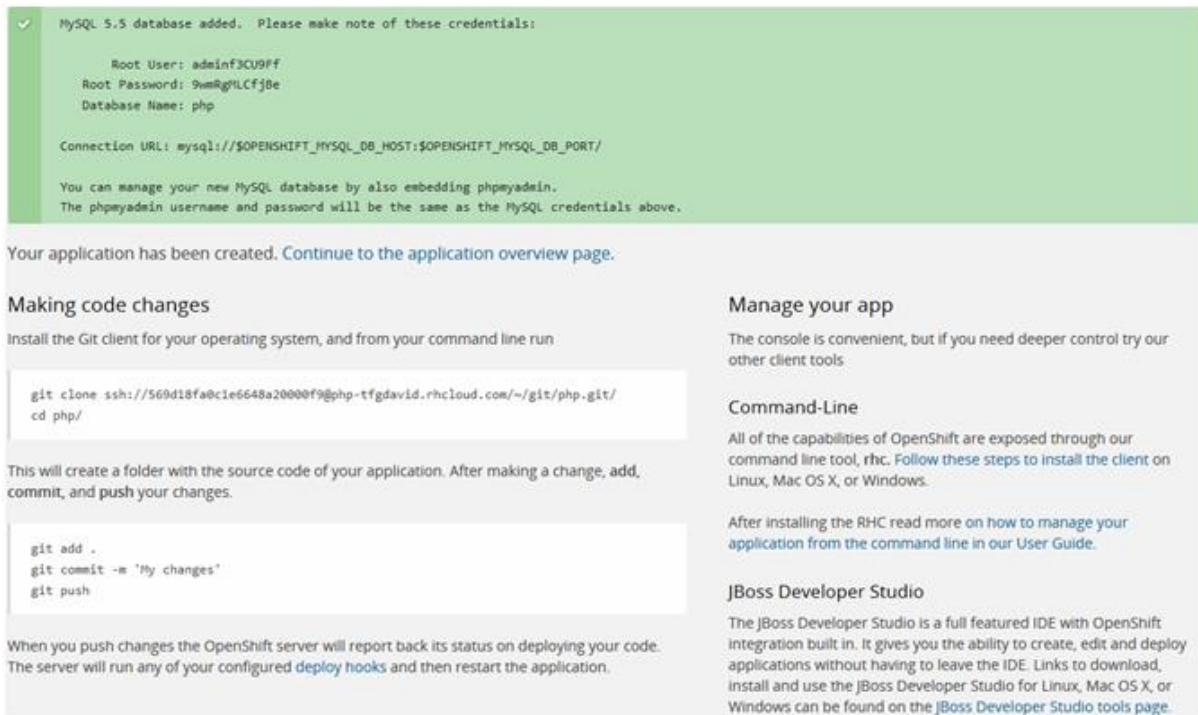


Figura 34. Ejemplo de información sobre credenciales al crear la aplicación. Fecha: 18/01/2016

- A partir de ese momento, cuando se inicie sesión, se podrá acceder a un panel centralizado donde configurar la aplicación. Como se puede observar, sugiere añadir phpMyAdmin en “Tools and Support”.

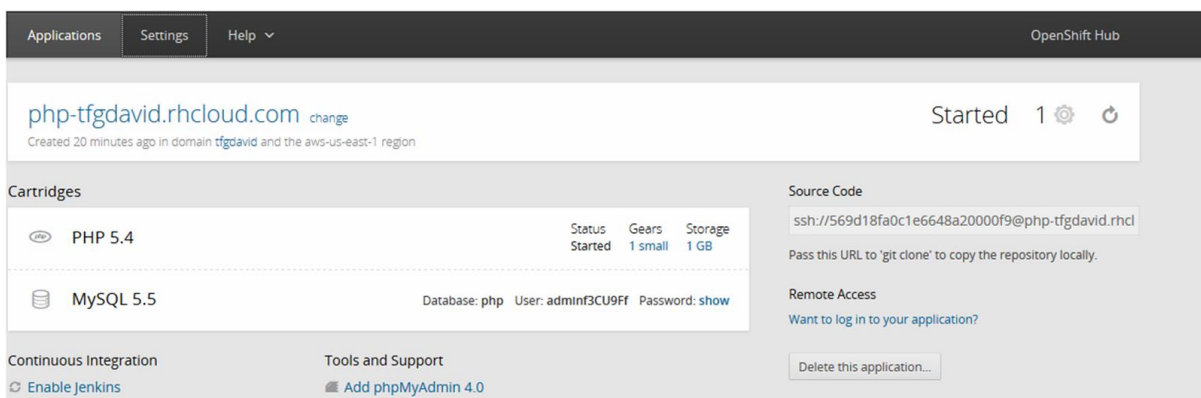


Figura 35. Panel de control para gestionar una aplicación Openshift. Fecha: 18/01/2016

10.1.2. Acceso remoto y gestión de aplicaciones

Debido a que esta parte es más general y está muy bien documentada por parte de Openshift, se procede a mencionar las fuentes de información utilizadas.

- Instalación de las herramientas de cliente, conocidas como Openshift Client tools o rhc [39] que permiten controlar la aplicación web desde terminal de forma más completa que el panel de la figura anterior.
- Configuración de la conexión remota segura para poder acceder desde un cliente Git mediante SSH [40].

10.1.3. Colaboradores

Una opción útil en grandes proyectos es la de agregar colaboradores. Pasos a seguir:

- Hacer clic en el dominio (en este caso "tfgdavid")



Figura 36. Panel principal para gestionar todas las aplicaciones. Fecha: 18/01/2016

- Mediante "Add members..." agregar los colaboradores. Hay los siguientes permisos:
- Ver (View): sólo puede visualizar el dominio y sus aplicaciones.
- Editar (Edit): añadir, modificar y eliminar aplicaciones en el dominio. También gestionar el acceso mediante Git y SSH.
- Administrar (Administer): control total, incluyendo qué usuarios son miembros.

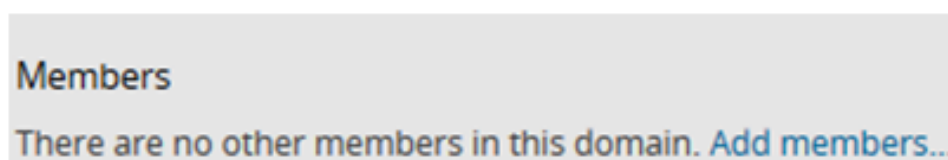


Figura 37. Gestionar los miembros de la cuenta. Fecha: 18/01/2016

10.2. Servidor (PHP, MySQL) usando Xampp

10.2.1. Configuración general

Suponiendo que está Xampp [8] instalado, lo primero que hay que hacer es abrir su panel de control e iniciar todos los módulos que se necesiten.

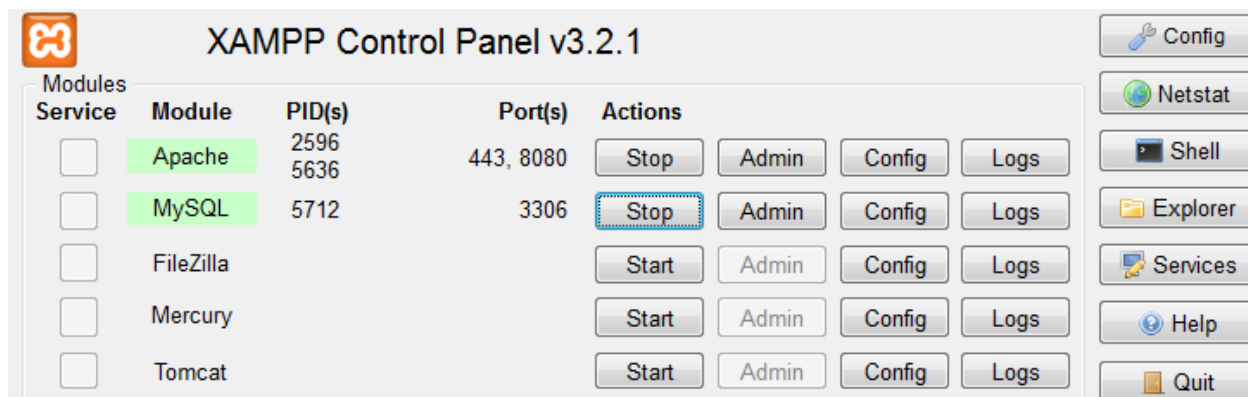


Figura 38. Panel de Control de Xampp con Apache y MySQL iniciados.

Xampp, por defecto, es inseguro para probar localmente nuestras aplicaciones y hay que configurarlo para solventar sus vulnerabilidades.

Cuando se hayan iniciado los módulos que quiera usar, debe ir a “Security”:

<http://localhost:8080/security/index.php>

La forma más rápida de lograr mayor seguridad es siguiendo los consejos que aparecen en el enlace de la siguiente imagen, entre los que se incluye establecer contraseñas seguras de acceso.

To fix the problems for mysql, phpmyadmin and the xampp directory simply use
=> <http://localhost/security/xamppsecurity.php> <= [allowed only for localhost]

Figura 39. Enlace que permite configurar de forma rápida Xampp y sus módulos

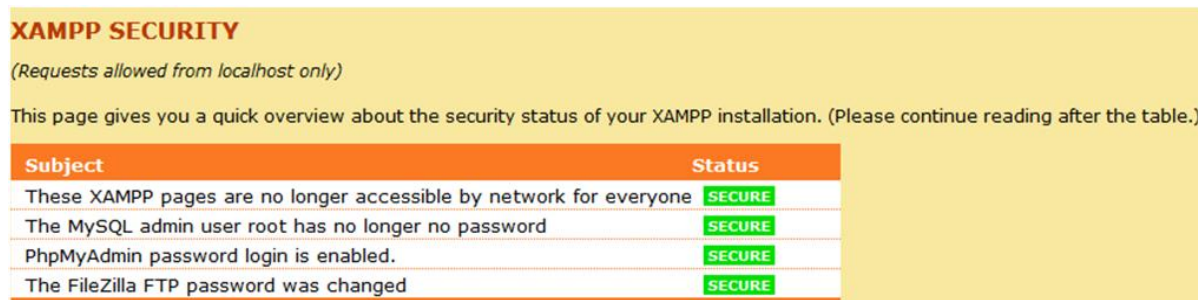


Figura 40. Buen nivel de seguridad en Xampp (todo en verde e indicando “SECURE”)

10.2.2. Incorporar aplicación

En el siguiente ejemplo, se va a considerar “prueba” como el nombre del nuevo proyecto (aplicación) a crear.

Las aplicaciones deben ir en la carpeta htdocs de la instalación de Xampp [8].

Por ejemplo, en Windows, una posible ruta es: C: \xampp \htdocs \prueba

Mediante un cliente Git, se puede bajar una aplicación creada en Openshift, como se ha explicado en el “apartado 10.1”. Nuestra instalación de Composer puede diferir de la del servidor, provocando incompatibilidades. Para solventarlo, hay que realizar los siguientes pasos en la carpeta del proyecto, “prueba” (una vez bajado):

- Si la instalación de Composer es antigua, es recomendable actualizarla antes:
composer self-update
- Comprobar dependencias del proyecto e instalar las que falten: **composer install**
- Actualizar todas sus dependencias: **composer update**

Si lo que se desea es crear una nueva aplicación local, se pueden seguir las instrucciones (dentro de htdocs) mediante Composer [20]:

- En caso de no tener instalado Laravel, se descarga poniendo en terminal:
composer global require "laravel/installer"
- Se crea un nuevo proyecto. Hay dos formas de hacerlo pero la más sencilla es mediante: **composer create-project --prefer-dist laravel/laravel prueba**

El proceso de crear la aplicación es lento porque descarga dependencias y realiza configuraciones.

Importante: Hay que dar permisos de escritura a los elementos de las carpetas del proyecto “storage” y “bootstrap/cache” para evitar errores.

Si se han seguido los pasos correctamente y se inicia en Xampp el servidor Apache, se podrá visualizar una página web con el mensaje “Laravel 5” en una dirección similar a la siguiente: <http://localhost:8080/prueba/public/>

En la dirección web (url) hay el texto “/public”, ya que es donde está alojada la aplicación cliente, tal y como se explica en el apartado “6.Arquitectura de Develink”.



Figura 41. Aplicación Laravel creada correctamente. Muestra la página “welcome” por defecto.

En la documentación de Laravel hay más información sobre la instalación [41].

10.2.3. Configuración del servicio de correo electrónico

Para poder enviar emails a los usuarios que han olvidado contraseña, fue necesario configurar un servidor de correo. Los pasos seguidos están ampliamente documentados en la web de Laravel [42] y la de Mailgun [43].

10.2.4. Fichero de configuración Apache

```
<IfModule mod_rewrite.c>
  <IfModule mod_negotiation.c>
    Options -MultiViews
  </IfModule>

  RewriteEngine On

  # Redirect Trailing Slashes...
  RewriteRule ^(.*)/$ /$1 [L,R=301]

  # Handle Front Controller...
  RewriteCond %{REQUEST_FILENAME} !-d
  RewriteCond %{REQUEST_FILENAME} !-f
  RewriteRule ^ index.php [L]
</IfModule>

# 1 year cache for images and fonts
<FilesMatch "\.(ico|jpg|jpeg|png|gif|eot|svg|ttf|woff|woff2|map|bmp)$">
Header set Cache-Control "max-age=31536000, public"
</FilesMatch>

#1 week javascript and css
<FilesMatch "\.(js|css)$">
Header set Cache-Control "max-age=604800, public"
</FilesMatch>
```

Figura 42. Archivo de configuración Apache “.htaccess” de Develink. Se establece una caché de un año para imágenes y fuentes. De una semana para archivos js y css.

10.3. Manual de usuario

Este apartado contiene el manual de los primeros pasos a realizar y los específicos de los diferentes roles (trabajador y empresa). El de administrador no se incluye debido a que no dio tiempo a realizar sus funciones.

La dirección web de Develink es la siguiente: <https://develink-tfgdavid.rhcloud.com/>

Se proporcionan cuentas para realizar pruebas (tests) en la aplicación (contraseña: 123456)

- **Trabajador:** w@mail.com
- **Empresa:** e@mail.com
- **Administrador** (sin funciones aún): a@mail.com

10.3.1. Primeros pasos

10.3.1.1. Página de Bienvenida

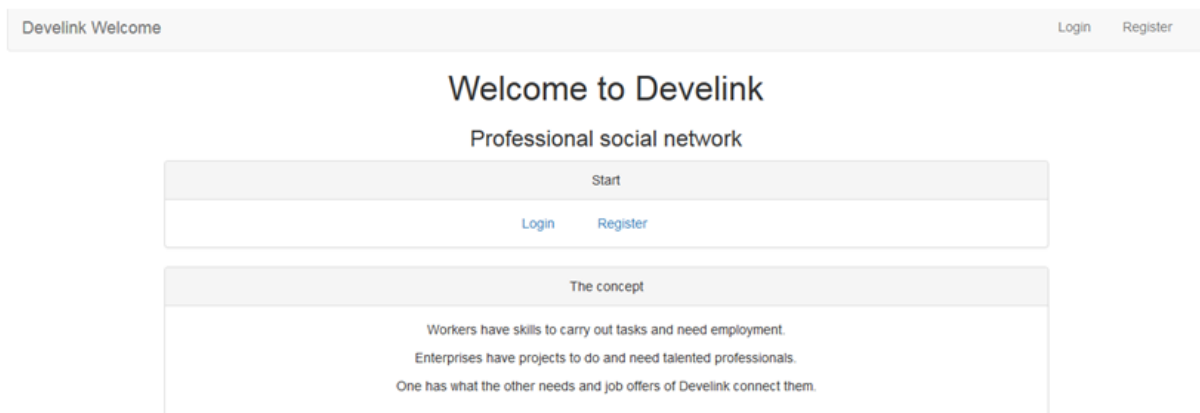


Figura 43. Página de bienvenida

La primera página que puede visualizar es la de bienvenida. Le permite crear una cuenta de empleado (trabajador) mediante la opción “Register”. Además, si está registrado puede iniciar sesión con “Login”. También le explica el concepto de Develink:

“Los trabajadores poseen habilidades para llevar a cabo tareas y buscan empleo. Las empresas tienen proyectos a realizar y requieren unos buenos profesionales. Unos tienen lo que los otros necesitan y Develink los conecta”.

Como puede observar, contiene una barra de navegación superior con enlaces a lugares importantes de la web. Debido a que (esa barra) es un elemento común, se omitirá (a excepción que se mencione de forma explícita) en las siguientes figuras para permitir una mejor visualización del contenido (cuerpo) de las páginas en cuestión.

10.3.1.2. Register

The image shows a registration form titled "Register". At the top, there are two radio buttons: "Worker" (which is selected) and "Enterprise". Below these are six text input fields, each with a placeholder label: "Name", "Nick", "E-Mail", "Photo URL", "Password (min 6)", and "Password Confirmation". At the bottom of the form is a blue button labeled "Register".

Figura 44. Página de registro: "Register"

Debe introducir:

- Si se desea crear una cuenta con rol Trabajador (por defecto) o Empresa.
- El nombre completo.
- El apodo o nickname (no puede coincidir con el de otros usuarios, único).
- El correo electrónico (único).
- Un enlace que lleve a una fotografía o imagen
- Una contraseña de como mínimo 6 dígitos.
- Confirmación de contraseña (que la anterior coincida con la introducida en este campo).

Si se crea satisfactoriamente, iniciará sesión como el usuario acabado de registrar y recibirá un mail de bienvenida.



Figura 45. Ejemplo de correo de bienvenida. Incluye contraseña, página inicial y rol de registro.

10.3.1.3. Login

A screenshot of a login form. At the top is a "Login" header. Below it are two input fields: "E-Mail" and "Password (min 6)". There is a "Remember Me" checkbox below the password field. A blue "Login" button is centered below the form. At the bottom, there is a link "Forgot Your Password?".

Figura 46. Página de inicio de sesión (identificación): "Login"

Introduzca el correo electrónico, el password y pulse el botón Login.

Hay una opción llamada "Remember Me" que, si la selecciona, le permite continuar la sesión aunque cierre el navegador (mientras no borre los datos de navegación). Esta función no la debe usar en dispositivos compartidos o públicos ya que otras personas podrían entrar.

10.3.1.4. Proceso de reinicio de contraseña

La web contempla que pueda olvidar su contraseña ("Forgot Your Password?"). El sencillo y seguro proceso que debe seguir en ese caso se explica a continuación.

Figura 47. Página de solicitud reinicio de contraseña: “Reset password”

Importante: con las cuentas de prueba (test) esta funcionalidad carece de sentido porque los correos electrónicos no son reales.

Cuando realiza la solicitud, la aplicación le envía un enlace por correo electrónico para que pueda cambiar su contraseña. Si el email es correcto, recibirá un mensaje como el mostrado a continuación.

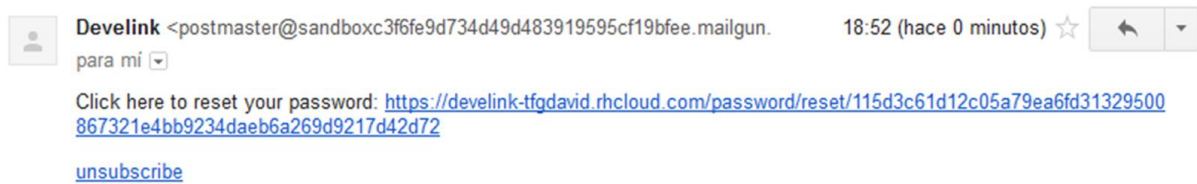


Figura 48. Correo recibido con un enlace para cambiar la contraseña.

Si hace clic en el enlace recibido, muestra un formulario que permite cambiar la contraseña. De esa forma, podrá usar su cuenta con normalidad usando su nueva contraseña.

Figura 49. Página para efectuar el cambio de contraseña

10.3.2. Manual de Trabajador

10.3.2.1. Barra de navegación



Figura 50. Barra de navegación del usuario Trabajador. Aparece en la parte superior.

La barra (de la imagen anterior) le facilita el acceso a las siguientes páginas (algunas de ellas se explicarán a fondo posteriormente):

- **Welcome:** página de bienvenida (explicada en “Primeros Pasos”).
- **Home:** página donde podrá realizar de forma centralizada la mayoría de acciones. En otras webs a esta página se le llama Dashboard.
- **Skills:** listado de habilidades existentes.
- **Job Offers:** lista de ofertas de empleo públicas.
- **Create Skill:** permite crear una nueva habilidad.
- **Direct Job Offers:** lista de ofertas laborales directas, es decir, que se han enviado al trabajador de forma específica.

En la parte derecha de la barra, hay un pequeño menú desplegable que además de mostrar su nombre y rol (trabajador), permite ir a “Home” y cerrar sesión (“Logout”).

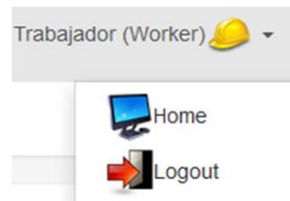


Figura 51. Menú desplegable situado a la derecha de la barra superior.

10.3.2.2. Home

Puede visualizar:

- **My rating:** muestra el número de votos que ha recibido y, si son suficientes, su nota (entre 0 y 100). El color de la barra y el texto que la acompaña dependerá de la valoración.

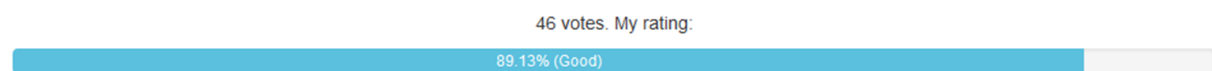


Figura 52. My rating

- **Notifications:** los avisos que ha recibido debido a acciones que le afectan por parte de otro usuario. Si pone el cursor encima de la “X” se muestra la fecha en la que sucedió y, si la pulsa, elimina la notificación



Figura 53. Ejemplo de notificación.

- **Gestor de referencias de “Worker”:** puede visualizar sus referencias (experiencias laborales) que saldrán con un tick de validadas. Además es posible solicitar, mediante el botón “+Add”, a una empresa para la que trabajó que le valide otras experiencias laborales obtenidas. Es necesario introducir habilidad, proyecto y empresa. Se permite su eliminación pero use esta opción con precaución. La skill “unspecified” sirve de para indicar que ha trabajado en una empresa sin indicar la habilidad adquirida.

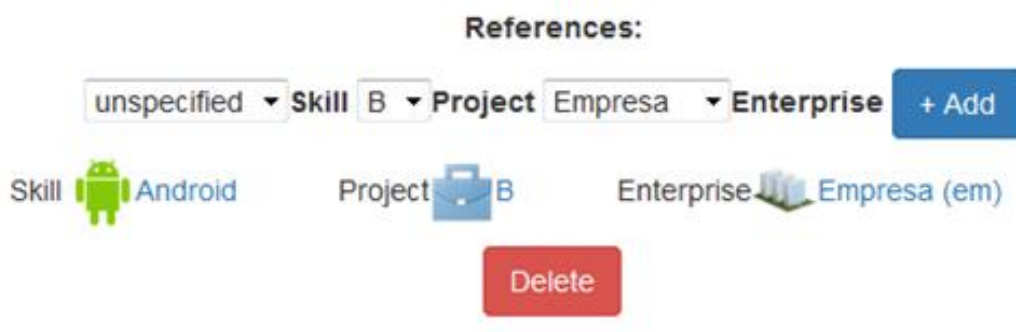


Figura 54. Gestión de referencias.

- **Current jobs:** muestra en qué lugares (“proyecto en empresa”) está trabajando actualmente.



Figura 55. Empleo actual: b es el proyecto, Empresa es el nombre de la empresa y “em” el apodo.

- **Rate previous jobs on enterprises:** permite valorar anónimamente con una nota entre 1 (muy mal) y 4 (excelente) a empresas donde ha finalizado su contrato (excepto si lo despidieron). Hay que pulsar el botón “Rate” para realizarla.



Figura 56. Empleo anterior que se puede valorar

- **Profile:** muestra su perfil, permite editarlo y dispone de una vista previa actual (“Preview”). Se puede modificar la descripción, la página web personal y la fotografía (url o dirección web que lleve a una imagen).

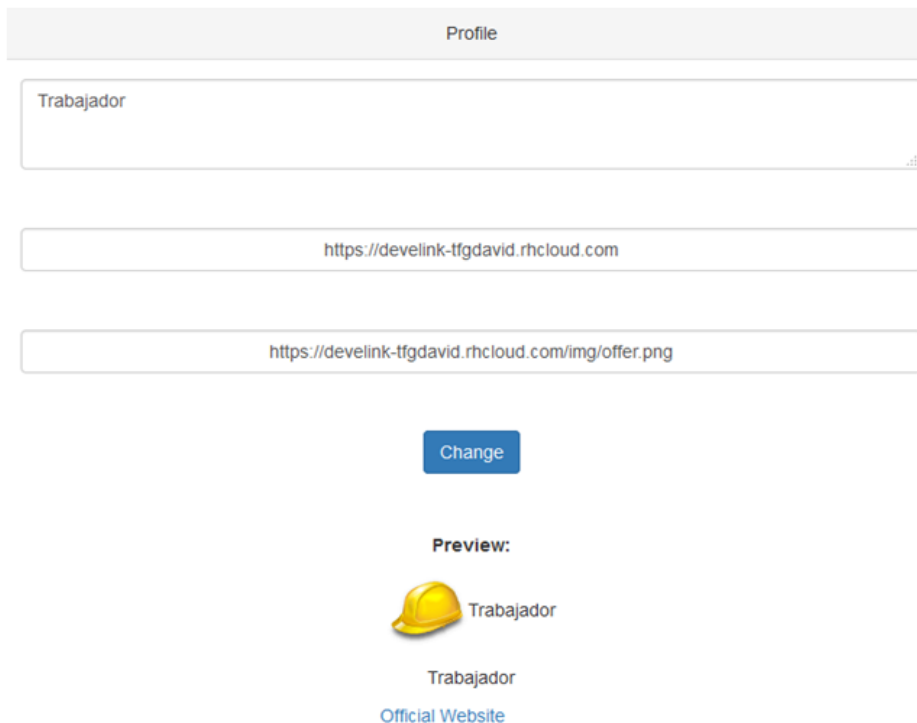


Figura 57. Profile: permite visualizar y editar el perfil

10.3.2.3. Skills

La página muestra, por defecto, todas las habilidades existentes.

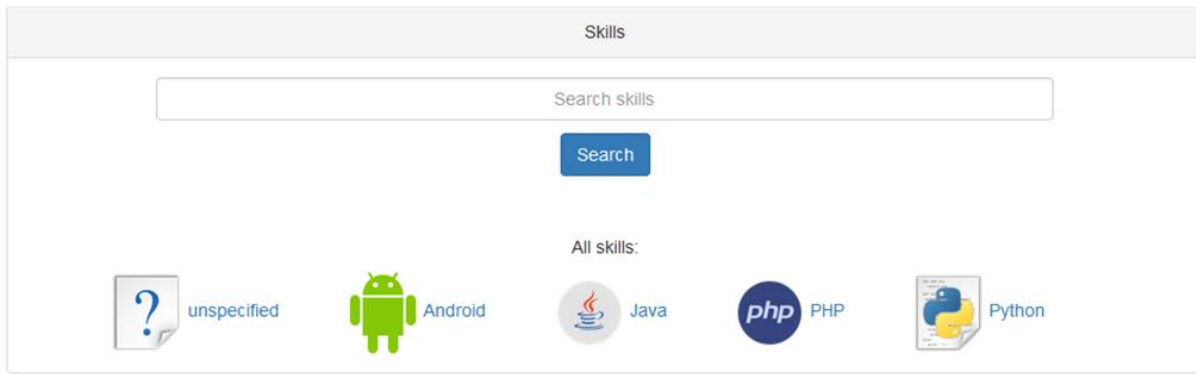


Figura 58. Página Skills con todas las habilidades.

Como puede haber muchos resultados, le permite realizar búsquedas con "Search". Si hay coincidencia, se destaca y el resto se ordenan por similitud (de mayor a menor).

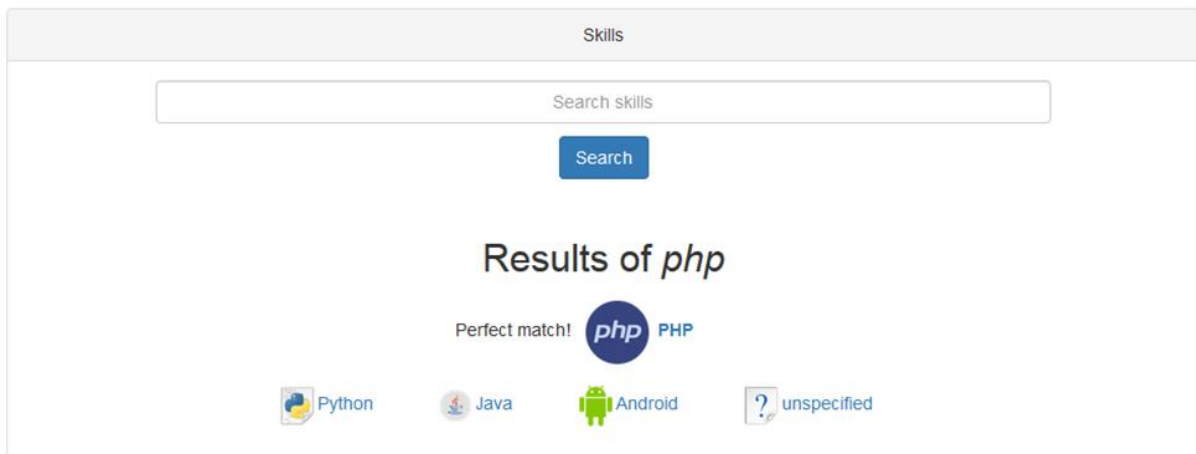


Figura 59. Página Skills en la que se busca "php". Muestra la coincidencia exacta y como segunda python que es la que más se parece al texto buscado.

En caso que no la haya, se visualiza de la siguiente forma:



Figura 60. Página Skills en la que la búsqueda no tiene coincidencia exacta.

Si hace clic en una de ellas, se muestra una ficha con la descripción y un enlace con más información.

Skill description

 Java

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented...

[More information](#)

Figura 61. Ficha descriptiva de una habilidad (en este caso Java).

10.3.2.4. Create Skill

Le permite crear una nueva habilidad (que no exista). Es necesario:

- Nombre
- Descripción
- Dirección web que lleve a una imagen
- Web con más información.

Create a Skill

Figura 62. Formulario para crear habilidad

10.3.2.5. Job Offers

La página muestra, por defecto, todas las ofertas laborales públicas. Le permite realizar búsquedas para acotar los resultados.

Job Offers


Search job offers

Search

All Job Offers:

Enterprise: Empresa **Title:** Mobile [Apply now](#)

I need a mobile developer. Only Android is required but iOS knowledge will be valued.

 Android

Title: Developer [Apply now](#)

We want a developer with the skills specified


 Java

Figura 63. Ofertas de empleo públicas

Como puede observar en la imagen anterior, en la primera oferta se muestra “Enterprise” y “Title” pero en la siguiente sólo el título. El motivo es que la empresa que ha creado la segunda oferta no quiere revelar su identidad.

Para ser candidato a una oferta, debe pulsar el botón verde “Apply now” correspondiente. Cuando lo haga, le llegará un aviso a la empresa indicando que usted tiene interés en su puesto, se apagará el botón y cambiará su texto a “Applied”.

10.3.2.6. Direct Job Offers

La página muestra las ofertas laborales que le han enviado a usted específicamente. Es la única diferencia respecto Job Offers, ya que su funcionamiento es idéntico.

10.3.3. Manual de Empresa

10.3.3.1. Barra de navegación

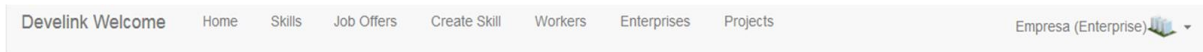


Figura 64. Barra de navegación del usuario Empresa. Aparece en la parte superior.

La barra (de la imagen anterior) le facilita el acceso a las siguientes páginas (algunas de ellas se explicarán a fondo posteriormente):

- **Welcome:** página de bienvenida (comentada en “Primeros Pasos”).
- **Home:** página donde podrá realizar de forma centralizada la mayoría de acciones. En otras webs a esta página se le llama Dashboard.
- **Skills:** listado de habilidades existentes.
- **Job Offers:** lista de ofertas de empleo públicas.
- **Create Skill:** permite crear una nueva habilidad.
- **Workers:** lista de empleados existentes.
- **Enterprises:** lista de empresas
- **Projects:** listado de proyectos de las empresas.

En la parte derecha de la barra, hay un pequeño menú desplegable que además de mostrar el nombre y rol del usuario identificado, permite ir a “Home” y cerrar sesión (“Logout”).

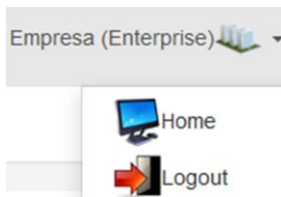


Figura 65. Menú desplegable situado a la derecha de la barra superior.

10.3.3.2. Home

Puede visualizar:

- **My rating:** muestra el número de votos que ha recibido y, si son suficientes, su nota (entre 0 y 100). El color de la barra y el texto que la acompaña dependerá de la valoración.

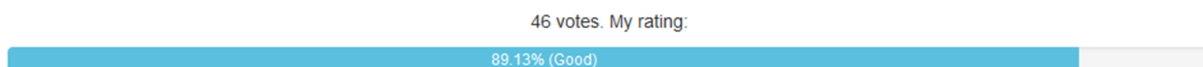


Figura 66. Su valoración obtenida a partir de la nota de otros usuarios.

- **Notifications:** avisos que ha recibido debido a acciones que le afectan por parte de otro usuario. Si pone el cursor encima de la “X” se muestra la fecha en la que sucedió y si la pulsa elimina la notificación.

Notifications: 1

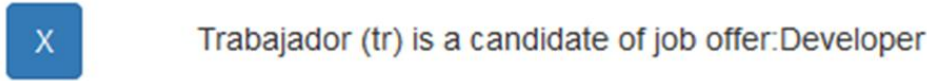


Figura 67. Ejemplo de notificación

- **Projects manager.** Le permite:
 - Ir a un formulario para crear un proyecto.
 - Ver listados los proyectos anteriormente creados. Haciendo clic en el icono o en el nombre de uno de ellos lo puede gestionar y con el botón “Delete”, eliminar.

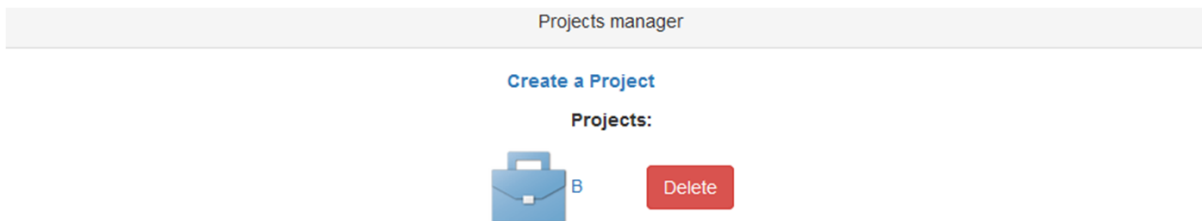


Figura 68. Gestión de proyectos. Se puede ver uno de nombre “B”.

- **Validations.** Muestra todas las referencias laborales relativas a su empresa (incluidas las peticiones). Le permite:
 - Validar o Invaldar la referencia. Para guardar los cambios debe usar el botón “Save”.
 - Borrarlas (botón “Delete”). La eliminación debe usarla con cautela ya que supondrá la supresión de esa experiencia laboral en el currículum del empleado correspondiente.

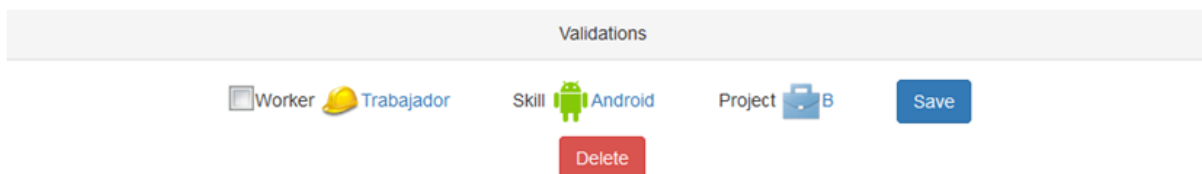


Figura 69. Validación de referencias laborales.

- **Profile:** muestra su perfil, permite editarlo y dispone de una vista previa actual (“Preview”). Se puede modificar la descripción, la página web oficial y la fotografía (url o dirección web que lleve a una imagen).

Profile


descripcion

<https://develink-tfgdavid.rhcloud.com>

<https://develink-tfgdavid.rhcloud.com/img/enterprise.png>

Change

Preview:

 Empresa

descripcion

[Official Website](#)

Figura 70. Profile: permite visualizar y editar el perfil

10.3.3.3. Create a project

Le permite crear y definir un proyecto en su empresa. Este paso es necesario para poder publicar ofertas laborales. Requiere:

- Nombre del proyecto.
- Descripción.
- Dirección web (enlace) que lleve a una imagen.
- Página web oficial del proyecto.

The image shows a web form titled "Create a Project". It consists of a header bar with the text "Create a Project". Below the header are four input fields: "Name", "Description", "Photo URL", and "Website". At the bottom of the form is a blue button labeled "Create".

Figura 71. Creación de un proyecto

Una vez que se ha creado, es posible editarlo (haciendo clic en su nombre o imagen desde el panel "Projects manager" de "Home" comentado anteriormente) y agregarle ofertas laborales (Job Offers). Se explicará detalladamente en el próximo apartado.

10.3.3.4. Edit project

Permite modificar los siguientes aspectos del proyecto seleccionado:

- **Gestionar ofertas laborales:** puede crear una oferta, editar alguna de las existentes (haciendo clic encima del nombre) o borrarla (con el botón "Delete" correspondiente).

The image shows a web interface titled "Job Offers Manager". It features a header bar with the text "Job Offers Manager". Below the header, there is a link "Create a Job Offer" and a section titled "Job Offers:". Under this section, there are two entries: "Mobile" and "Developer". Each entry has a red "Delete" button next to it.

Figura 72. Gestionar ofertas laborales

- **Gestión de contratos:** ve a los trabajadores en plantilla (contratados). Se contempla la posibilidad de finalizar su contrato satisfactoriamente ("Completion contract") y también el despido ("Fire"). En ambos casos es obligatorio puntuar a su antiguo empleado. La nota va comprendida entre 1 (muy mal) y 4 (excelente).

El trabajador en cuestión será notificado automáticamente (de la decisión pero no de la nota).



Figura 73. Gestionar contratos (trabajadores en plantilla).

- **Editar información general del proyecto.** Puede cambiar:
 - Nombre.
 - Descripción.
 - Web oficial.
 - Enlace a una imagen que represente al proyecto.

Hace falta hacer clic en “Save” para guardar los cambios. Se muestra una vista previa.

Figura 74. Formulario para editar proyecto.

10.3.3.5. Create Job Offer

Para publicar una oferta laboral necesita proporcionar la siguiente información:

- Título de la oferta
- Descripción

Además, puede elegir si muestra el nombre de su empresa en la oferta (por defecto) o no (marque “Phantom”). No se olvide de pulsar “Create” para publicarla.

Realizando estos pasos, ha creado una oferta laboral sin especificar habilidades requeridas. Si lo desea hacer, debe editarla (como se explica en el siguiente apartado, en “Gestionar habilidades de una oferta laboral”).

The screenshot shows a form titled "Create a Job Offer". It contains two input fields: "Title" and "Description". Below the "Description" field is a checkbox labeled "Phantom (Hide enterprise name)". At the bottom of the form is a blue button labeled "Create".

Figura 75. Formulario para crear oferta laboral.

10.3.3.6. Edit Job Offer

Le permite editar los siguientes aspectos una oferta laboral:

- **Ver a los candidatos a esa oferta** (personas que se han interesado por ella). Cuando seleccione a uno del listado, le aparecerá una fotografía con el nombre. Si hace clic, visualiza su currículum. De esa manera, puede asegurarse que es el perfil que busca. Cuando haya tomado la decisión, pulse el botón “Contract”. Es importante la decisión ya que sólo puede contratar a uno a la vez (y los demás se descartan).

The screenshot shows the "Candidates of Job Offer" interface. It displays the message "You can only contract 1 person per offer". Below this message is a green button labeled "Contract", a dropdown menu showing "Trabajador (tr)", a yellow hard hat icon, and the word "Trabajador".

Figura 76. Candidatos a una oferta laboral

- **Editar la información y visibilidad de la oferta.** Con “Save & Publish” se guardan los cambios. Se puede modificar:
 - Título
 - Descripción
 - Si se muestra el nombre de la empresa en la oferta o no.
 - Si es pública o no (direct). En ese caso cuando se selecciona “Direct”, hay que usar “Save & Publish” (para ocultar la oferta al público) antes de poder elegir a los empleados a los que se quiere informar de forma privada.

Mobile

I need an Android developer.

Phantom (Hide enterprise name)

Direct (Only for a list of workers)

Save & Publish

+ Add --Workers -- Worker

No workers added

Figura 77. Editar oferta laboral

- **Gestionar habilidades de una oferta laboral:**
 - Permite agregar skills (habilidades) seleccionando la categoría (con la lista desplegable) y luego la habilidad en cuestión. Si no selecciona una categoría, se mostrarán todas las habilidades existentes en la lista desplegable Skill.
 - Si no existe la habilidad que busca, puede usar el enlace “Create skill” (el cual le llevará a un formulario para crearla).

- En “current skills” se muestran las que ha agregado a la oferta y las puede eliminar con “Delete”.

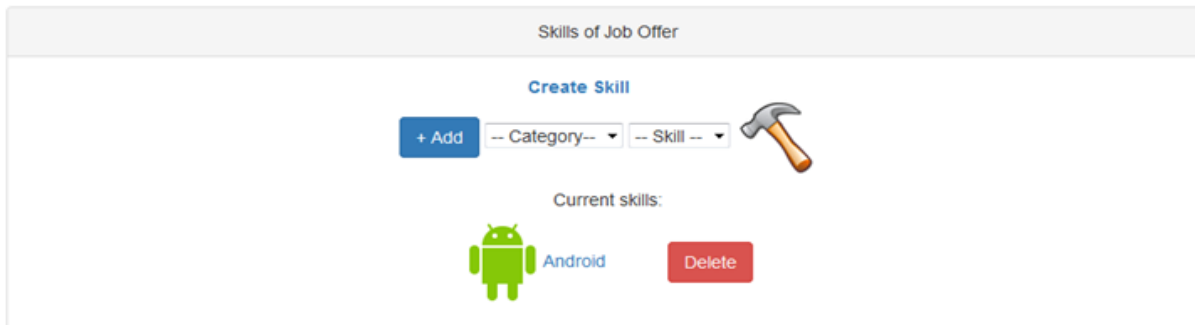


Figura 78. Gestionar habilidades de una oferta laboral

10.3.3.7. Otras páginas de utilidad

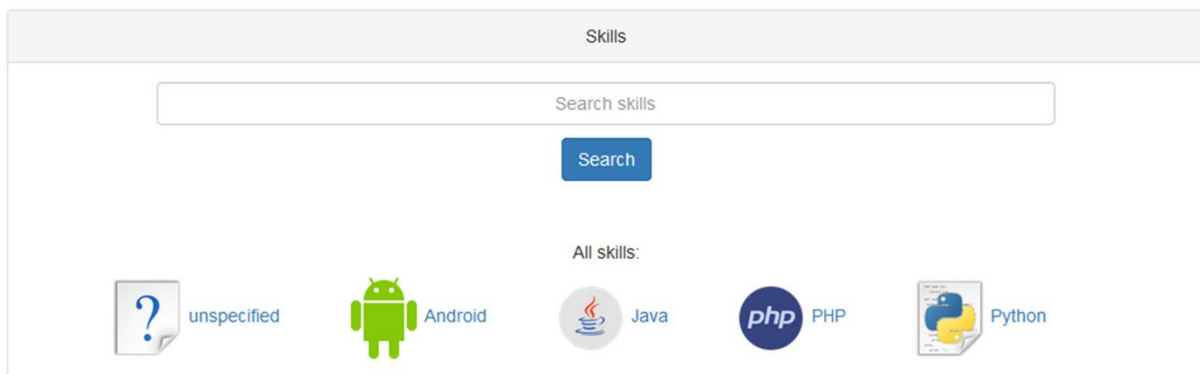


Figura 79. “Skills”. Listado de todas las habilidades existentes. Más información en el apartado “10.3.2.3.Skills” del manual de Trabajador.

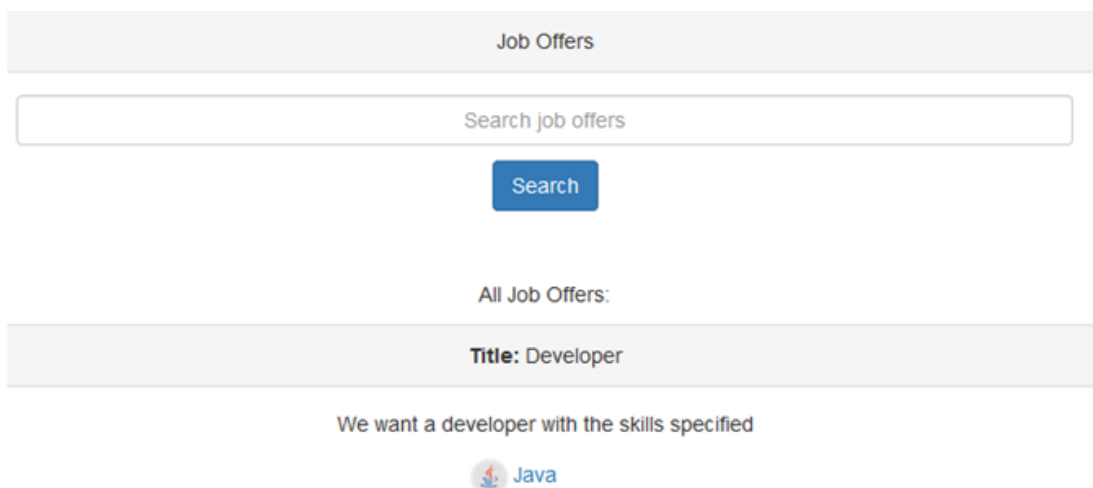


Figura 80. “Job Offers”. Listado de ofertas laborales públicas que permite búsqueda para acotar resultados. La oferta mostrada es “phantom” ya que no se muestra la empresa (Enterprise), sólo título (“Title”) y descripción.

Create a Skill

Figura 81. "Create Skill": Formulario para crear habilidad. Más información en el apartado "10.3.2.4 Create Skill" del manual de Trabajador.

Workers


 Trabajador (tr)

Figura 82. "Workers": Lista de empleados. Con el buscador se pueden acotar los resultados.

Worker information



Trabajador (tr)

Trabajador

48 votes

89.58% (Good)

Personal Website

References:


 Java  B  Empresa

Figura 83. Descripción de un empleado que se visualiza al hacer clic en su fotografía o nombre.

Enterprises

Search




 Empresa (em)  Empresa2 (em2)

Figura 84. "Enterprises": Lista de empresas. Con el buscador se pueden acotar los resultados.

Enterprise information



Empresa (em)

descripcion

11 votes

88.64% (Good)

Official website

Figura 85. Descripción de una empresa que se visualiza al hacer clic en su fotografía o nombre.

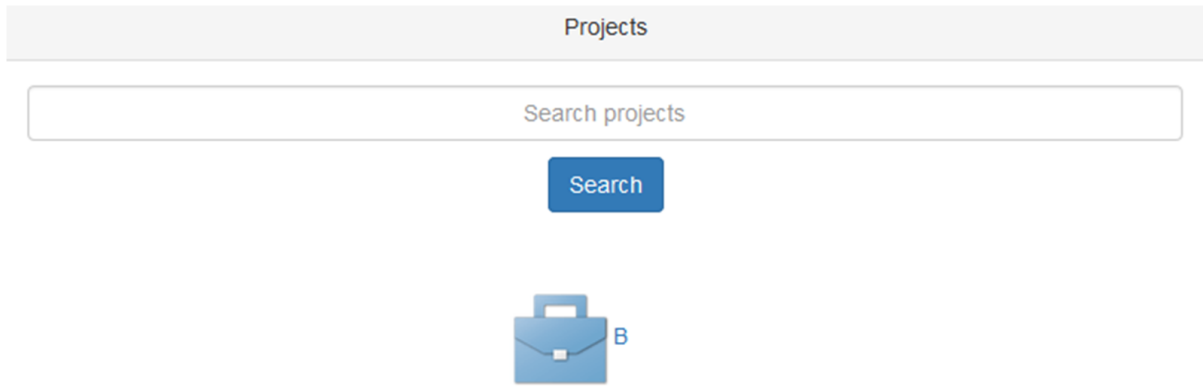


Figura 86. "Projects": Lista de proyectos. Con el buscador se pueden acotar los resultados.

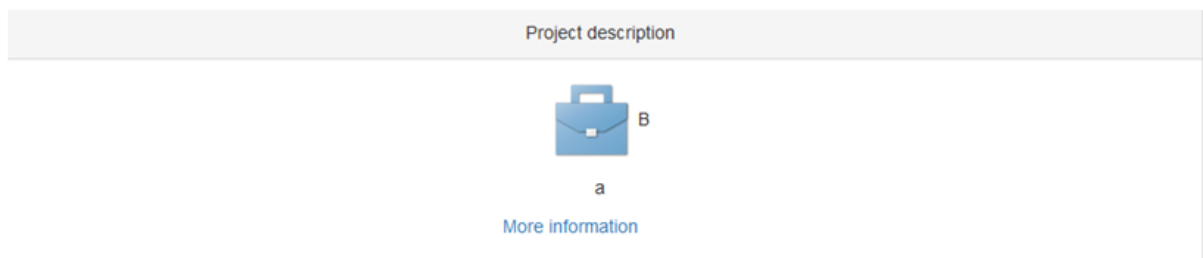


Figura 87. Descripción de un proyecto que se visualiza al hacer clic en su fotografía o nombre.