

PLANIFICACIÓN Y GESTIÓN ACTIVA DE PROYECTOS EN GNU/LINUX CON REDMINE. (Memoria final).

Estudiante: José María Campos Lahoz, I.T.I.S.
Consultor UOC: Joaquin Lopez Sanchez Montañés

Fecha: 13 de enero de 2013



Índice

I	Introducción.	7
1.	Descripción básica.	7
2.	Funcionalidades de Redmine.	7
3.	Objetivos	8
4.	Problemas encontrados y su Solución	8
4.1.	LiveCD us LiveUSB	8
II	Instalación.	9
5.	Sistema operativo CentOS 6.3	9
5.1.	Repositorios y mirrors	10
5.2.	Más información	10
6.	Ruby.	10
6.1.	Objetivo	11
6.2.	Semántica	11
6.3.	Características	11
6.4.	Interacción	12
6.5.	Sintaxis	12
6.6.	Licencia	12
7.	Ruby on Rails.	12
7.1.	Filosofía	13
7.2.	Historia	13
7.3.	Arquitectura MVC de Rails	13
7.3.1.	Modelo	13
7.3.2.	Vista	14
7.3.3.	Controlador	14
7.4.	Otros módulos	14
7.5.	Gemas	14
7.6.	Soporte de servidores Web	15
7.7.	Soporte de Bases de Datos	15
7.8.	Requisitos	15
7.9.	Entorno de Trabajo	15
8.	RubyGems.	15
9.	Bundler.	16
9.1.	¿Qué es Gemfile.lock?	16
10.	GIT.	16
11.	MUA y MTA.	17
11.1.	MUA o (Mail User Agent).	17
11.2.	MTA (Mail Transfer Agent).	18
12.	NTP (Network Time Protocol).	18

13. Servidor HTTP Apache.	19
13.1. Ventajas	19
13.2. Uso	19
13.3. Configuración	19
13.4. Licencia	20
14. MySQL.	20
14.1. Desarrollo del proyecto	20
14.2. MySQL en cifras	20
15. Redmine.	20
15.1. Características	21
III Seguridad	23
15.2. Acceso restringido a Mysql	23
16. IPTables.	25
17. LiveUSB, Características	26
17.1. Grabación “liveUSB”	27
17.2. Pre-requisitos	27
IV Utilización de RedMine	29
18. Guía de instalación	29
18.1. Descargar RedMine	29
18.2. Configuración	29
18.3. Configuración de RedMine	29
18.4. Crear la estructura de base de datos:	30
18.5. BitNami, instalador todo en uno.	30
19. Actualizar una instalación existente	31
19.1. Actualización	31
20. Guía del administrador	33
20.1. Configuración de la aplicación.	33
20.1.1. Mostrar	33
20.1.2. Autenticación	33
20.1.3. Proyectos	33
20.1.4. Seguimiento de asuntos	33
20.1.5. Repositorios	34
20.2. Roles y permisos	34
20.2.1. Roles propiedades	34
20.2.2. Funciones del sistema	34
21. Guía del usuario	34
21.1. Cuentas de usuario	34
21.1.1. Mi cuenta	34
21.2. Ingreso	36
21.3. Registro	37
21.4. Búsquedas en Redmine	37
21.4.1. Caja de búsquedas	38
21.4.2. Configuración de la búsqueda	38
21.4.3. Resultados	38

21.4.4. Controlador de la búsqueda	38
21.5. Vista del proyecto	39
21.6. Actividad del proyecto	40
21.7. Diagramas de Gantt	41
21.8. Calendario	41
21.8.1. Filtros	41
21.9. Noticias	42
21.10 Documentos Redmine	42
21.10.1.Introducción	42
21.10.2.Añadir Documentos	42
21.10.3.Visor de Documentos	42
21.11Ficheros en Redmine	42
21.11.1.Módulo de ficheros.	42
21.11.2.Almacenamiento de los ficheros.	44
21.11.3.Formulario de nuevo fichero	44
21.12Foros	44
21.12.1.Añadir un tema	45
22.Repositorio	45
V Conclusiones.	47
23.Redmine, gestor de proyectos de código libre para nuestras empresas	47
24.Principales características de Redmine	47
25.Iniciando el trabajo con Redmine	47
26.Conclusiones	49
VI Anexos	51
A. Nomenclatura	51
B. Instrucciones de uso	55
B.1. Accesos	55
B.2. Redmine	55
B.3. Login	55
B.4. IPTables	55
C. Bibliografía de URL's	57

Parte I

Introducción.

Redmine es una herramienta que nos permite trabajar de manera colaborativa en el seguimiento y la gestión de proyectos, y que además incluye un sistema de control de incidencias. Para ello posee diversas herramientas que son, calendario de actividades, diagramas de Gantt, wiki, foro, visor del repositorio de control de versiones, RSS, control de flujo de trabajo basado en roles, integración con correo electrónico, etcétera.

Está escrito usando el framework Ruby on Rails. Es software libre y de código abierto, disponible bajo la Licencia Pública General de GNU v2..

1. Descripción básica.

Redmine es un gestor y planificador de proyectos con interfaz web, orientado a la coordinación de tareas entre varios participantes. Mi trabajo va a consistir en una instalación de una aplicación Redmine contenida en una distribución LiveUSB. En ella, diversos grupos de personas podrán alcanzar unos objetivos comunes, maximizando los resultados óptimos y minimizando la pérdida de tiempo y costes.

Para su puesta en marcha se deberá implementar diverso software, aplicaciones, base de datos, paquetes y comandos de sistema operativo, etc. que son los que darán el soporte necesario a Redmine.

Pero usar esta herramienta no es tarea sencilla porque tras ella hay un sinfín de tecnologías diversas que se deben dominar. Las mismas nos permitirán la comunicación entre las personas componentes de los diversos grupos de trabajo, mantener un mínimo de seguridad contra ataques externos al sistema en su explotación continua, deberán guardar la información de forma persistente para su uso continuado en el tiempo, y otras muchas más que enumeraré más adelante.

2. Funcionalidades de Redmine.

Redmine permite gestionar múltiples proyectos desde una sola interfaz de una ventana de navegador. Los proyectos pueden ser totalmente independientes unos de otros, o complementarios siguiendo una estructura jerárquica. La navegación es muy sencilla y se puede saltar y cambiar de proyecto en cualquier momento. Además cada proyecto puede tener una configuración totalmente diferente y el usuario tener un rol distinto en cada uno. Los proyectos pueden definirse como privados, en los que el administrador debe dar acceso a cada miembro, permitiendo el cambio de ciertas propiedades de cada proyecto, o todo lo contrario, restringiéndolas. O pueden ser proyectos públicos, visibles para todo el mundo. También dentro de cada proyecto pueden definirse varios subproyectos.

- **Personalización de proyectos:** En Redmine cada proyecto es totalmente personalizable, pudiendo encontrar proyectos muy distintos entre sí según sus objetivos. Lo más importante son los módulos que se pueden desactivar o activar para cada proyecto: wiki, foro, noticias, peticiones, control del tiempo, documentos, ficheros o repositorio, aunque hay módulos comunes a todos los proyectos como el de actividad y vistazo.
- **Sistema flexible de seguimiento de tareas:** Una de las mecánicas más útiles para el desarrollo de un proyecto en Redmine son las peticiones y su visualización. Estas peticiones se dividen en 3 tipos iniciales (errores, tareas y soporte) y pueden asignarse a un miembro del proyecto. Se puede indicar una fecha de inicio y fin para esa petición, e incluso llevar un control del tiempo y porcentaje realizado. Además dentro de un proyecto pueden establecerse versiones y asignar tareas a determinadas versiones, así conforme se marquen tareas completadas, las versiones irán completando su porcentaje automáticamente.
- **Uso de calendario y diagrama de Gantt:** Redmine incluye un calendario para visualizar todas las peticiones a lo largo de un mes elegido, marcando claramente el día de inicio y de

fin de cada petición. Igualmente ocurre con la vista en diagrama de Gantt, que va marcando el porcentaje completado conforme avanzan los días.

- **Notificaciones:** Configurando previamente el servidor de correo SMTP, Redmine permite enviar notificaciones por correo electrónico en todos los proyectos, definiendo antes los eventos que activan estos avisos. Además cada usuario en su configuración puede elegir recibir notificaciones de cualquier evento, o solo las relacionadas con él (por ejemplo uno de los campos de las peticiones son las personas en seguimiento).
- **Exportación a distintos formatos:** Los informes de peticiones que pueden generarse añadiendo filtros, y que permiten visualizar las diferentes tareas de un proyecto, pueden exportarse en PDF o formato CSV, pudiendo así imprimirlos posteriormente en un formato organizado. Las páginas de la wiki en cambio, pueden exportarse en HTML o TXT.
- **Otras características:** Redmine es una herramienta llena de funcionalidad y donde casi cualquier cosa puede configurarse o contiene opciones. Algunas otras funcionalidades que habría que destacar son la página personal de cada usuario, que ofrece una vista personalizable con información de todos los proyectos donde está participando, como un calendario global, o peticiones asignadas. Además cabe mencionar que admite como bases de datos MySQL, SQLite y PostgreSQL.

3. Objetivos

El **Objetivo principal** es crear una plataforma, Redmine, que apoye y gestione el trabajo de un conjunto de personas. La plataforma deberá tener un adecuado control y seguimiento de los proyectos que de forma paralela gestione. Cuando un proyecto se planifica se debe fiscalizar su desarrollo para comprobar que marcha sobre el plan previsto, es decir, debe satisfacer los objetivos de calidad, coste y tiempo.

En la actualidad muchas empresas carecen de un ambiente colaborativo de trabajo y que integre herramientas para lograr un control sobre la planificación, la documentación y en consecuencia el trabajo final asociado al proyecto.

4. Problemas encontrados y su Solución

En este momento la instalación elegida funciona pero al intentar instalar las novedosas últimas versiones del aplicativo me he encontrado con problemas de versiones de código. La solución ha pasado por rebajar el número de versión de diversos complementos para adecuarlos a una versión mas estable, probada y menos experimental. Las versiones del diverso software utilizado son las siguientes:

Entorno	
Redmine	2.1.0. stable
Ruby	1.8.7
Rails	3.2.8
entorno del desarrollo	'production'
MySQL	5.1.66
CentOS	6.3

4.1. LiveCD us LiveUSB

También en un principio la idea era la de realizar un liveCD para la entrega del trabajo, pero entre la cantidad de problemas aparecidos y la mejor idoneidad de realizarlo sobre un liveUSB al final cambié de idea y se ha presentado en un liveUSB. Todas las características sobre este tema están ampliamente explicadas en la página 26 del presente trabajo.

Parte II

Instalación.

La instalación del sistema Redmine comprende varias fases.

1. Una primera es la instalación de un Sistema Operativo.
2. Instalación de diverso software específico para la utilización de Redmine. Lenguaje de programación *Ruby*, Framework *Rails*, Base de datos *Mysql*, Servidor de páginas web *Apache*, etc...
3. Configuración de la seguridad. Al ser un servidor público su seguridad puede verse comprometida, se deben de seguir unas cuantas normas para asegurarlo contra ataques externos a la LAN.
4. Grabación del sistema en un LiveUSB para su presentación y evaluación como TFC.

A continuación veremos una enumeración de cada elemento necesario en la instalación de Redmine clasificado por características y comentado someramente.

5. Sistema operativo CentOS 6.3

La instalación del Sistema CentOS 6.3 la he realizado directamente en una memoria liveUSB [1]. En estos momentos existen en Internet un sinfín de aplicaciones que te lo facilita [2].

Con estas acciones conseguimos instalar un sistema mínimo. Podemos ver ampliada la información de la instalación en la sección que contiene la “Figura 11 (Fedora, Live-USB creator)”. Este sistema mínimo es fruto de la imagen que nos hemos bajado del servidor de Software de CentOS, a partir de ahora se debe ir instalando todo el resto de paquetes de software necesarios para poner en funcionamiento el Redmine.

La instalación del Sistema CentOS 6.3 la he realizado directamente en una memoria USB. En estos momentos existen un sinfín de aplicaciones que lo facilitan. Sin embargo esta ha sido la parte del trabajo que mas dificultades me ha dado. El sistema instalado en la memoria USB al ser un sistema “live” es mínimo, esto quiere decir que para completarlo se debe instalar un sinfín de paquetes de software que le faltan para alcanzar nuestros objetivos.

Esta versión de sistema, está orientada a empresas y se basa en el código fuente de Red Hat Enterprise Linux 6.3. Está disponible en 32 y 64 bits y se incluyen paquetes de todas las variantes de RHEL 6.3. Destaca la inclusión de LibreOffice, y hay nuevas herramientas para migrar máquinas virtuales y físicas desde su formato original a un formato de máquina virtual KVM.

CentOS tiene una compatibilidad garantizada con RHEL (Red Hat Enterprise Linux), y todo manual que sirva para RHEL sirve en CentOS sin modificación. En muchos casos, es la principal herramienta de trabajo en servidores, y su estabilidad es envidiable.

Las imágenes oficiales del Sistema originalmente son estas:

```
sha1sum i386:
```

```
541f98e36a7034ab3b470ddf5e2232df3829983e CentOS-6.3-i386-bin-DVD1.iso
44d9e8652af683b844138ea2ea03e6772c18a613 CentOS-6.3-i386-bin-DVD2.iso
c596411085110dbb67fb030e667ae054afb413c8 CentOS-6.3-i386-minimal.iso
51dcbf68ddc0fc2907ecbba055bf041dc8ae7ca9 CentOS-6.3-i386-netinstall.iso
7cb1ea7eb35e4ee7112d4cd6b56528562cab6c65a CentOS-6.3-i386-LiveCD.iso
f34c3f9c6880bffe6344aa6a0381275689f0bf02 CentOS-6.3-i386-LiveDVD.iso
```

```
sha1sum x86-64:
```

```
ea52f3d1ccf2df3b03e064fb0fa6168be28a4d4 CentOS-6.3-x86-64-bin-DVD1.iso
29eee5946372ed29de509b9604d895b99d36a33f CentOS-6.3-x86-64-bin-DVD2.iso
09801301433c8dc1a93d732790f9a4d8ca7895e6 CentOS-6.3-x86-64-minimal.iso
```

```
c85069ee289d6eb279ba1c1ab0c259513302b8fd CentOS-6.3-x86-64-netinstall.iso
e04fbc13598800e5b71a3d2eb62af66100acfcfc CentOS-6.3-x86-64-LiveCD.iso
e810015efaa0e3adf0c0a3c100f79eb8f8e83e97 CentOS-6.3-x86-64-LiveDVD.iso
```

Yo me he decantado por la “*CentOS-6.3-i386-LiveCD.iso*” por su relación entre tamaño y prestaciones que da. No es la más ligera, que sería lo aconsejable para la instalación sobre un USB, pero contiene de serie la mayoría de paquetes que vamos a necesitar.

5.1. Repositorios y mirrors

Todos los paquetes compilados y fuentes de CentOS están ahora alojados en `vault.centos.org`. En esta dirección podremos encontrar desde las versiones más antiguas a las más actuales 6. En la dirección **CentOSPlus** encontraremos diversos kernels, en **OS** tenemos todos los paquetes en formato standard *'rpm'* para compilar, y en **Updates** las últimas versiones del software importante que cambia antes de que aparezca otra revisión del sistema, si no fuera así posiblemente algún paquete de software se volvería obsoleto antes de modernizar todo el sistema con una nueva versión. Los **Extras** son paquetes de terceras partes que por problemas en la licencia no se pueden poner juntos en el repositorio oficial.

CentOSPlus: <http://vault.centos.org/6.3/centosplus/Source/SPackages/>

Extras: <http://vault.centos.org/6.3/extras/Source/SPackages/>

OS: <http://vault.centos.org/6.3/os/Source/SPackages/>

Updates: <http://vault.centos.org/6.3/updates/Source/SPackages/>

5.2. Más información

Las siguientes páginas web tienen una gran cantidad de información para ayudar a la gente en sus problemas con sus sistemas CentOS.

- Notas de las versiones del proveedor y documentación. [3]
- Qué hay de nuevo en Enterprise Linux 6.3. Descripción detallada de las nuevas funcionalidades que posee la nueva versión 6.3. [4, 5, 6]
- Foros de discusión de CentOS-6. Lugar donde se habla sobre los temas que interesa a la comunidad CentOS. [7, 8, 9, 10]

6. Ruby.

Ruby [11] es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro "Matz" Matsumoto, quien comenzó a trabajar en Ruby en 1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en Python y Perl con características de programación orientada a objetos similares a Smalltalk. Comparte también funcionalidad con otros lenguajes de programación como Lisp, Lua, Dylan y CLU. Ruby es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre.

En el círculo de amigos de Matsumoto se le puso el nombre de "Ruby" (en español rubí) como broma aludiendo al lenguaje de programación "Perl" (perla).



Figura 1: Logotipo de Ruby

6.1. Objetivo

El creador del lenguaje, ha manifestado que Ruby está diseñado para la productividad y la diversión del desarrollador, siguiendo los principios de una buena interfaz de usuario. Sostiene que el diseño de sistemas necesita enfatizar las necesidades humanas más que las de la máquina:

'A menudo la gente, especialmente los ingenieros en computación, se centran en las máquinas. Ellos piensan, "Haciendo esto, la máquina funcionará más rápido. Haciendo esto, la máquina funcionará de manera más eficiente. Haciendo esto..." Están centrados en las máquinas, pero en realidad necesitamos centrarnos en las personas, en cómo hacen programas o cómo manejan las aplicaciones en los ordenadores. Nosotros somos los jefes. Ellos son los esclavos'.

Ruby sigue el "principio de la menor sorpresa", lo que significa que el lenguaje debe comportarse de tal manera que minimice la confusión de los usuarios experimentados.

6.2. Semántica

Ruby es un lenguaje orientado a objetos: todos los tipos de datos son un objeto, incluidas las clases y tipos que otros lenguajes definen como primitivos, (como enteros, booleanos, y "nil"). Toda función es un método. Las variables siempre son referencias a objetos, no los objetos mismos. Ruby soporta herencia con enlace dinámico, mixins y métodos singleton (pertenecientes y definidos por un sola instancia más que definidos por la clase). A pesar de que Ruby no soporta herencia múltiple, la clases pueden importar módulos como mixins. La sintaxis procedural está soportada, pero todos los métodos definidos fuera del ámbito de un objeto son realmente métodos de la clase Object. Como esta clase es padre de todas las demás, los cambios son visibles para todas las clases y objetos.

Ruby ha sido descrito como un lenguaje de programación multiparadigma: permite programación procedural (definiendo funciones y variables fuera de las clases haciéndolas parte del objeto raíz Object), con orientación a objetos, (todo es un objeto) o funcionalmente (tiene funciones anónimas, clausuras o closures, y continuations; todas las sentencias tiene valores, y las funciones devuelven la última evaluación). Soporta introspección, reflexión y metaprogramación, además de soporte para hilos de ejecución gestionados por el intérprete. Ruby tiene tipado dinámico, y soporta polimorfismo de tipos (permite tratar a subclases utilizando la interfaz de la clase padre). Ruby no requiere de polimorfismo de funciones al no ser fuertemente tipado (los parámetros pasados a un método pueden ser de distinta clase en cada llamada a dicho método).

6.3. Características

- Orientado a objetos
- cuatro niveles de ámbito de variable: global, clase, instancia y local.
- manejo de excepciones iteradores y clausuras o closures (pasando bloques de código)
- expresiones regulares nativas similares a las de Perl a nivel del lenguaje
- posibilidad de redefinir los operadores (sobrecarga de operadores)
- recolección de basura automática altamente portable.

- Hilos de ejecución simultáneos en todas las plataformas usando green threads
- Carga dinámica de DLL/bibliotecas compartidas en la mayoría de las plataformas
- introspección, reflexión y metaprogramación amplia
- librería estándar
- soporta inyección de dependencias
- soporta alteración de objetos en tiempo de ejecución
- continuaciones y generadores

Ruby actualmente no tiene soporte completo de Unicode, a pesar de tener soporte parcial para UTF-8.

6.4. Interacción

La distribución oficial de Ruby incluye "irb" (Interactive Ruby Shell), un intérprete interactivo de línea de comandos que puede ser usado para probar código de manera rápida.

6.5. Sintaxis

1. La sintaxis de *Ruby* es similar a la de *Perl* o *Python*. La definición de clases y métodos está definida por palabras clave. Sin embargo, en *Perl*, las variables no llevan prefijos. Cuando se usa, un prefijo indica el ámbito de las variables.
2. La mayor diferencia con *C* y *Perl* es que las palabras clave son usadas para definir bloques de código sin llaves. Los saltos de línea son significativos y son interpretados como el final de una sentencia; el punto y coma tiene el mismo uso. De forma diferente que *Python*, la indentación no es significativa.
3. Una de las diferencias entre *Ruby*, *Python* y *Perl* es que *Ruby* mantiene todas sus variables de instancia privadas dentro de las clases y solo la expone a través de métodos de acceso (*attr_writer*, *attr_reader*, etc). A diferencia de los métodos "getter" y "setter" de otros lenguajes como *C++* o *Java*, los métodos de acceso en *Ruby* pueden ser escritos con una sola línea de código. Como la invocación de estos métodos no requiere el uso de paréntesis, es trivial cambiar una variable de instancia en una función sin tocar una sola línea de código o refactorizar dicho código.

6.6. Licencia

El intérprete y las bibliotecas están licenciadas de forma dual (inseparable) bajo las licencias libres y de código abierto GPL y Licencia pública Ruby.

A partir de la versión 1.9.3 se opta por una licencia dual bajo las licencias BSD de dos cláusulas y Licencia pública Ruby.

7. Ruby on Rails.

Ruby on Rails[12], también conocido como RoR o Rails es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC). Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros frameworks y con un mínimo de configuración. El lenguaje de programación Ruby permite la metaprogramación, de la cual Rails hace uso, lo que resulta en una sintaxis que muchos de sus usuarios encuentran muy legible. Rails se distribuye a través de RubyGems, que es el formato oficial de paquete y canal de distribución de bibliotecas y aplicaciones Ruby.



Figura 2: Logotipo de Rails

7.1. Filosofía

Los principios fundamentales de Ruby on Rails incluyen No te repitas (del inglés Don't repeat yourself, DRY) y Convención sobre configuración.

1. No te repitas significa que las definiciones deberían hacerse una sola vez. Dado que Ruby on Rails es un framework de pila completa, los componentes están integrados de manera que no hace falta establecer puentes entre ellos. Por ejemplo, en ActiveRecord, las definiciones de las clases no necesitan especificar los nombres de las columnas; Ruby puede averiguarlos a partir de la propia base de datos, de forma que definirlos tanto en el código como en el programa sería redundante.
2. Convención sobre configuración significa que el programador sólo necesita definir aquella configuración que no es convencional. Por ejemplo, si hay una clase *Historia* en el modelo, la tabla correspondiente de la base de datos es *historias*, pero si la tabla no sigue la convención (por ejemplo *blogposts*) debe ser especificada manualmente (`set_table_name "blogposts"`). Así, cuando se diseña una aplicación partiendo de cero sin una base de datos preexistente, el seguir las convenciones de Rails significa usar menos código (aunque el comportamiento puede ser configurado si el sistema debe ser compatible con un sistema heredado anterior)

7.2. Historia

Ruby on Rails fue escrito por David Heinemeier Hansson a partir de su trabajo en Basecamp, una herramienta de gestión de proyectos, por *37signals*. Fue liberado al público por primera vez en julio de 2004.

7.3. Arquitectura MVC de Rails

Las piezas de la arquitectura Modelo Vista Controlador en Ruby on Rails son las siguientes:

7.3.1. Modelo

En las aplicaciones web orientadas a objetos sobre bases de datos, el Modelo consiste en las clases que representan a las tablas de la base de datos.

En Ruby on Rails, las clases del Modelo son gestionadas por ActiveRecord. Por lo general, lo único que tiene que hacer el programador es heredar de la clase ActiveRecord::Base, y el programa averiguará automáticamente qué tabla usar y qué columnas tiene.

Las definiciones de las clases también detallan las relaciones entre clases con sentencias de mapeo objeto relacional.

El modelo representa:

- Las tablas de la Base de Datos.
- Migraciones (Expresan Cambios en las DDBB)

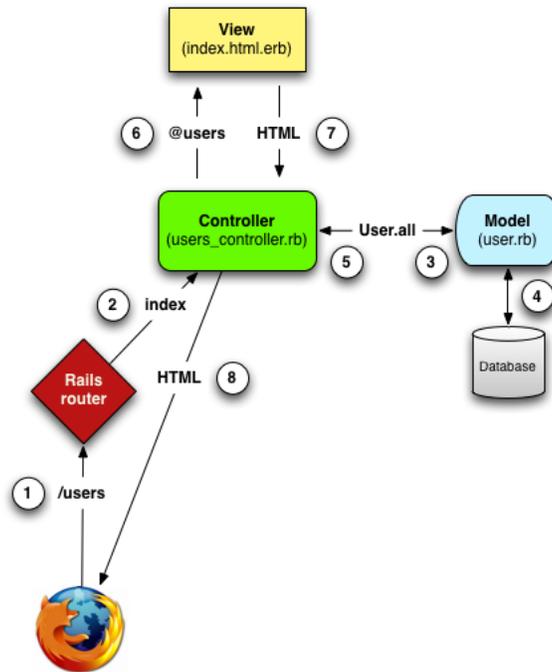


Figura 3: Diagrama detallado del MVC en Rails.

- Observadores

7.3.2. Vista

En MVC, Vista es la lógica de visualización, o cómo se muestran los datos de las clases del Controlador. Con frecuencia en las aplicaciones web, la vista consiste en una cantidad mínima de código incluido en HTML.

Existen en la actualidad muchas maneras de gestionar las vistas. El método que se emplea en Rails por defecto es usar Ruby empotrado, que son básicamente fragmentos de código HTML con algo de código en Ruby, siguiendo una sintaxis similar a JSP.

7.3.3. Controlador

En MVC, las clases del Controlador responden a la interacción del usuario e invocan a la lógica de la aplicación, que a su vez manipula los datos de las clases del Modelo y muestra los resultados usando las Vistas. En las aplicaciones web basadas en MVC, los métodos del controlador son invocados por el usuario usando el navegador web.

La implementación del Controlador es manejada por el ActionPack de Rails, que contiene la clase ApplicationController.

7.4. Otros módulos

Además, Rails ofrece otros módulos, como Action Mailer (para enviar correo electrónico) o Active Resource que proporciona la infraestructura necesaria para crear de manera sencilla recursos REST.

7.5. Gemas

Las gemas son plugins y/o códigos añadidos a nuestros proyectos Ruby on Rails, que nos permiten nuevas funcionalidades como nuevos create, nuevas funciones pre-escritas (como login de usuarios) o

nuevas herramientas para el desarrollo.

7.6. Soporte de servidores Web

Para desarrollo y pruebas, se utiliza Mongrel o WEBrick, incluido con Ruby. Para utilizar Rails en servidores en producción se está extendiendo el uso de Passenger, una suerte de mod_rails para Apache desarrollado en 2008 por la empresa holandesa Phusion. El módulo de Apache “Passenger” se utiliza en este TFC.

7.7. Soporte de Bases de Datos

Dada que la arquitectura Rails favorece el uso de bases de datos se recomienda usar un SGDBR para almacenamiento de datos. Rails soporta la biblioteca SQLite por defecto. Rails intenta mantener la neutralidad con respecto a la base de datos, la portabilidad de la aplicación a diferentes sistemas de base de datos y la reutilización de bases de datos preexistentes. Sin embargo, debido a la diferente naturaleza y prestaciones de los SGDBR’s el framework no puede garantizar la compatibilidad completa. Se soportan diferentes SGDBR’s, incluyendo MySQL, PostgreSQL, SQLite, IBM DB2 y Oracle.

7.8. Requisitos

Servidor web como Apache, lighttpd, algún servidor web compatible con FastCGI. Para desarrollo, Rails permite utilizar WEBrick (un pequeño servidor a medida de rendimiento limitado y no recomendado para su uso en producción). Base de datos (por ejemplo, MySQL, PostgreSQL, o SQLite)

7.9. Entorno de Trabajo

Hay muchas alternativas para trabajar con Ruby on Rails, tanto libres y gratuitas como de pago. A continuación se listan las principales:

Aptana: Nació como plugins de eclipse para la edición y desarrollo web. Actualmente se puede instalar como plugin o autónomo de forma independiente. Las últimas versiones están muy bien integradas con Ruby on Rails. En este momento Aptana 3 es la versión estable.

Netbeans: Uno de los más usados, libre y totalmente gratuito. Viene muy bien integrado con JRuby (lo cual es algo lógico pues es un programa de Oracle).

TextMate: Sólo para Mac. Es el entorno más usado entre la comunidad Rails. Es de pago pero su potencia y forma de trabajo favorece la producción y desarrollo con Ruby on Rails.

Gmate: Un proyecto libre y gratuito para convertir Gedit –el editor de texto de escritorio Gnome de GNU/Linux– en un clon muy aproximado de Textmate. Esto se consigue instalando diferentes plugins, temas y retocando algunas opciones. Al ser gratuito es una opción que está cogiendo muchos adeptos hoy en día.

Existen otros muchos, sólo es necesario usar algún buscador para encontrar más alternativas.

8. RubyGems.

RubyGems [13] es un gestor de paquetes para el lenguaje de programación Ruby que proporciona un formato estándar y autocontenido (llamado gem) para poder distribuir programas o librerías en Ruby, una herramienta destinada a gestionar la instalación de éstos, y un servidor para su distribución.

En resumen. Las ruby gems permiten instalar librerías de Ruby con un simple comando desde cualquier máquina que interprete Ruby y tenga ruby gems instalado.



Figura 4: Logo de Bundler

9. Bundler.

Bundler [14] es una herramienta que permite gestionar de forma sencilla y eficaz las dependencias de las aplicaciones en una aplicación Rails.

- No es específico de Rails (aunque nos centraremos en él)
- Simplifica la distribución de código
- Ofrece mecanismos para instalar/actualizar dependencias de forma cómoda
- Se asegura de que la aplicación siempre cuenta con las gemas necesarias para funcionar sin problemas.
- Todo se define en el fichero Gemfile de la aplicación Rails.

El comando:

```
1 $ bundle install
```

Instala todas las dependencias necesarias, y guarda las dependencias en un fichero llamado Gemfile.lock

9.1. ¿Qué es Gemfile.lock?

Gemfile.lock contiene una lista exhaustiva de todas las gemas que necesita la aplicación

- Aparecen las gemas que declaramos en Gemfile y sus dependencias
- Se registra su número de versión exacto (ref, tag o rama en el caso de Git)
- Este fichero debe estar siempre en nuestro sistema de control de versiones
- Permite que, independientemente de la máquina, se usen las mismas versiones de todo

10. GIT.

Git [15] es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. Al principio, Git se pensó como un motor de bajo nivel sobre el cual otros pudieran escribir la interfaz de usuario o front end como Cogito o StGIT. Sin embargo, Git se ha convertido desde entonces en un sistema de control de versiones con funcionalidad plena. Hay algunos proyectos de mucha relevancia que ya usan Git, en particular, el grupo de programación del núcleo Linux.



Figura 5: Logotipo de GIT.

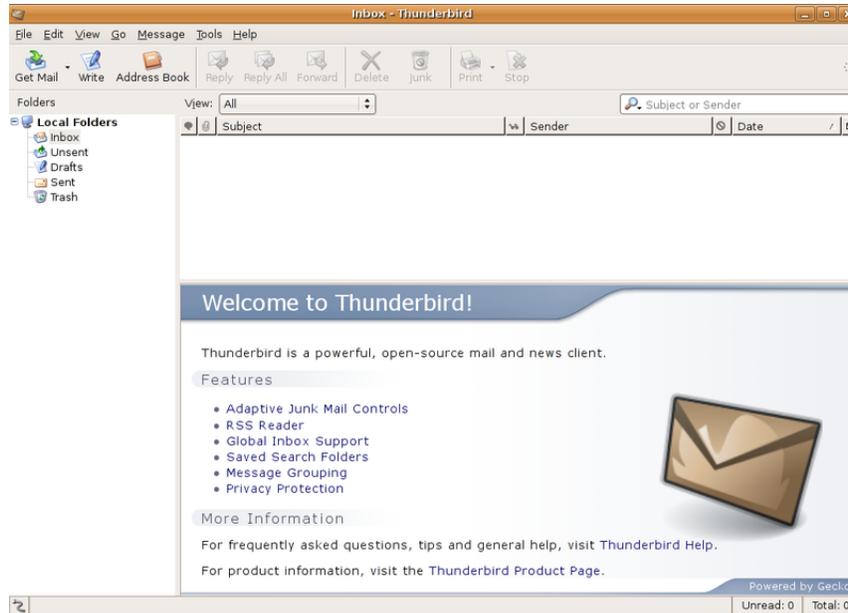


Figura 6: Cliente de correo electrónico.

En los últimos tiempos han proliferado servicios en la nube que albergan repositorios de código, uno de estos repositorios que están de moda es GitHub, pues bien, lo que hace la gente es, depositar en el servidor GitHub el desarrollo de Software que ha realizado en local, en su ordenador, para que de esta manera pueda ser accedido por cualquiera. El uso de GIT en la aplicación Rails es el siguiente. Accedemos al despliegue depositado por un programador en GitHub. Con comandos GIT, recordemos que es un software de control de versiones, y utilizando un fetch, te lo bajas a local en el directorio destino en Rails, y a funcionar.

11. MUA y MTA.

Redmine utiliza los servicios de correo electrónico con profusión, por lo tanto la utilización de un cliente de correo electrónico es obligatorio.

11.1. MUA o (Mail User Agent).

Un cliente de correo electrónico o MUA [16], es un programa de ordenador usado para leer y enviar mensajes de correo electrónico.

Originalmente, los clientes de correo electrónico fueron pensados para ser programas simples para



Figura 7: Logotipo NTP.

leer los mensajes del correo de usuario, enviados por el agente de reparto de correo (MDA) conjuntamente con el agente de transferencia de correo (MTA) a un buzón local.

Los formatos de buzón de correo más importantes son mbox y Maildir. Estos simplísimos protocolos para el almacenamiento local de los mensajes de correo electrónico realizan de una forma muy sencilla la importación, exportación y copia de seguridad de las carpetas de correo.

Los mensajes de correo electrónico pendientes de envío serán entregados al MTA, tal vez a través de un agente de correo saliente, de forma que el cliente de correo electrónico no necesita proporcionar ninguna clase de función de transporte.

IMAP está optimizado para almacenar correos electrónicos en el servidor, mientras que el protocolo POP3 asume generalmente que los mensajes de correo electrónico se descargan al cliente. La gran mayoría de clientes de correo electrónico emplean el Protocolo de Transferencia Simple de Correo (Simple Mail Transfer Protocol, SMTP) para enviar los mensajes de correo electrónico.

Un importante estándar soportado por la mayoría de los clientes de correo electrónico es MIME, que se emplea para el envío de archivos binarios adjuntos al correo. Los adjuntos son ficheros que no forman parte del correo electrónico propiamente dicho, pero que se envían junto con éste.

11.2. MTA (Mail Transfer Agent).

Un servidor de correo o MTA [17], es una aplicación de red ubicada en un servidor en Internet, cuya función es parecida al Correo postal tradicional, sólo que en este caso lo que se maneja son los correos electrónicos (a veces llamados mensajes o e-mails), a los que se hace circular a través de redes de transmisión de datos.

Los servidores de correo a menudo realizan diferentes funciones según sea el uso que se planifique para el mismo.

Agente de Transferencia de Correo (del inglés Mail Transport Agent o MTA; también Message Transport Agent, Agente de Transporte de Mensajes) es uno de los programas que ejecutan los servidores de correo, y tiene como fin transferir un conjunto de datos de una computadora a otra.

El MTA, tiene varias formas de comunicarse con otros servidores de correo:

Recibe los mensajes desde otro MTA. Actúa como "servidor" de otros servidores. Envía los mensajes hacia otro MTA. Actúa como un "cliente" de otros servidores. Actúa como intermediario entre un "Mail Submission Agent" y otro MTA.

Algunas soluciones de correo que incluyen un MTA son: Sendmail, qmail, Postfix, Exim, Mdaemon, Mercury Mail Transport System, Lotus Notes (IBM) y Microsoft Exchange Server.

Por defecto el protocolo estándar para la transferencia de correos entre servidores es el SMTP, o Protocolo Simple de Transferencia de Correo. Está definido en el RFC 2821[18] y es un estándar oficial de Internet.

12. NTP (Network Time Protocol).

Network Time Protocol (NTP) es un protocolo de Internet para sincronizar los relojes de los sistemas informáticos a través del ruteo de paquetes en redes con latencia variable. NTP[19] utiliza UDP como su capa de transporte, usando el puerto 123. Está diseñado para resistir los efectos de la latencia variable.

NTP utiliza el Algoritmo de Marzullo con la escala de tiempo UTC, incluyendo soporte para características como segundos intercalares. NTPv4 puede mantenerse sincronizado con una diferencia máxima de 10 milisegundos (1/100 segundos) a través de Internet, y puede llegar a acercarse hasta 200 microsegundos (1/5000 segundos) o más en redes de área local sobre condiciones ideales.



Figura 8: Logotipo de Apache.

NTP es uno de los protocolos de Internet más viejos que siguen en uso (desde antes de 1985). NTP fue diseñado originalmente por David L. Mills de la Universidad de Delaware, el cual lo sigue manteniendo, en conjunto con un equipo de voluntarios.

13. Servidor HTTP Apache.

El servidor HTTP Apache[20] es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, implementa la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que Behelendorf quería que tuviese la connotación de algo que es firme y enérgico, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU.

Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA.

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

13.1. Ventajas

Modular, Código abierto, Multi-plataforma, Extensible, Popular (fácil conseguir ayuda/suporte).

13.2. Uso

Apache es usado principalmente para enviar páginas web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizan características propias de este servidor web.

Este servidor web es redistribuido como parte de varios paquetes propietarios de software.

Apache es usado para muchas otras tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable.

Los programadores de aplicaciones web a veces utilizan una versión local de Apache con el fin de previsualizar y probar código mientras éste es desarrollado.

13.3. Configuración

La mayor parte de la configuración se realiza en el fichero `apache2.conf` o `httpd.conf`, según el sistema donde esté corriendo. Cualquier cambio en este archivo requiere reiniciar el servidor, o forzar la lectura de los archivos de configuración nuevamente.



Figura 9: Logotipo de Mysql

13.4. Licencia

La licencia de software bajo la cual el software de la fundación Apache es distribuido es una parte distintiva de la historia de Apache HTTP Server y de la comunidad de código abierto. La Licencia Apache permite la distribución de derivados de código abierto y cerrado a partir de su código fuente original.

14. MySQL.

MySQL[21] es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

14.1. Desarrollo del proyecto

El desarrollo de MySQL se fundamenta en el trabajo de los desarrolladores contratados por la empresa MySQL AB quienes se encargan de dar soporte a los socios comerciales y usuarios de la comunidad MySQL y dar solución a los problemas encontrados por los usuarios.

14.2. MySQL en cifras

Según las cifras del fabricante, existirían más de seis millones de copias de MySQL funcionando en la actualidad, lo que supera la base instalada de cualquier otra herramienta de bases de datos. El tráfico del sitio web de MySQL AB superó en 2004 al del sitio de IBM.

15. Redmine.

Redmine [22][23] es una herramienta para la gestión de proyectos que incluye un sistema de seguimiento de incidentes con seguimiento de errores. Otras herramientas que incluye son calendario de actividades, diagramas de Gantt para la representación visual de la línea del tiempo de los proyectos, wiki, foro, visor del repositorio de control de versiones, RSS, control de flujo de trabajo basado en roles, integración con correo electrónico, etcétera.

Está escrito usando el framework Ruby on Rails. Es software libre y de código abierto, disponible bajo la Licencia Pública General de GNU v2.



Figura 10: Logotipo Redmine.

15.1. Características

Soporta múltiples proyectos. Roles flexibles basados en control de acceso. Sistema de seguimiento de errores flexible. Diagramas de Gantt y calendario. Administración de noticias, documentos y archivos. Fuentes web y notificaciones por correo electrónico. Integración SCM (Subversion, CVS, Git, Mercurial, Bazaar y Darcs). Soporta diferentes bases de datos (MySQL, PostgreSQL y SQLite).

Instalación: http://www.redmine.org/projects/redmine/wiki/Como_instalar_Redmine

Documentación: <http://www.redmine.org/projects/redmine/wiki/EsGuide>

Las últimas versiones estables de Redmine son las siguientes:

2.2.0 (2012-12-18)

2.1.5 (2012-12-18)

1.4.5 (2012-11-17)

Parte III

Seguridad

Básicamente la seguridad la voy a enfocar sobre dos pilares:

1. Uno se basará en la base de datos MySQL, la propia DDBB está preparada para ello. Decir también que es una seguridad basada en las comunicaciones. Consistirá en que no voy a permitir los accesos a la DDBB desde fuera del servidor a excepción de las IP's que tenga dadas de alta en la propia DDBB.
2. El otro pilar reside a través de iptables. No voy a permitir ninguna conexión a la que no le dé explícitamente permiso. Esto se hace cerrando todos los puertos y poniendo las reglas de iptables a rechazar todo por defecto. Con ello permito solamente la conexión a las IP's fijas que yo determino.

A la hora de corregir el ejercicio esto va a ser un problema porque al desconocer las IP's en donde se corrija el trabajo no puedo dar de alta IP's específicas con la comunicación abierta, y si dejo la comunicación abierta totalmente carece de todo sentido la seguridad por su inexistencia. El estado final por defecto será que todo el personal incluido en la LAN se pueda conectar vía HTTP, pero no mediante ssh, ni tampoco a la MySQL. Ni que decir tiene que toda conexión externa a la LAN no podrá conectarse.

15.2. Acceso restringido a Mysql

Para hacer 'login' en la MySQL y que el acceso sea seguro lo primero que hay que hacer es ponerle un 'passwd' al usuario 'root' que por defecto en la instalación en CentOS no viene. Para ello, después de haber instalado todos los paquetes necesarios para la MySQL se lanza el siguiente comando para empezar a utilizarla.

```
'/usr/bin/mysql_secure_installation'
```

Este comando realiza una serie de acciones que paso a enumerar.

1. Por defecto la passwd de root está en blanco, esta primera acción te permite crear una passwd.
2. Por defecto también una base de datos MySQL tiene un usuario 'anonymous' que permite a cualquiera consultar la base sin que se haya dado de alta en ella. Esta acción te permite borrarlo.
3. La siguiente acción te pregunta si la base de datos MySQL permitirá o no que el usuario 'root' se pueda conectar de forma remota en vez de hacerlo en localhost. Por seguridad es recomendable contestar a esta pregunta que no se permita la conexión de forma remota al usuario 'root'.
4. Por defecto existe una base de datos llamada 'test', que como su nombre indica se utiliza para hacer pruebas. Esta parte del comando permite borrarla, y así se debe hacer.

Acto seguido se crean los usuarios. Siempre deben tener 'passwd'. En nuestro caso crearemos el usuario 'redmine'.

```
mysql> use mysql
grant all on *.* to redmine@localhost identified by 'redmine';
100% grant all on *.* to redmine@172.16.69.28 identified by 'redmine';
100% grant all on *.* to redmine@192.168.1.34 identified by 'redmine';
100% update user set Select_priv = 'Y' where user = 'redmine' and Host = 'localhost';
update user set Select_priv = 'Y' where user = 'redmine' and Host = '172.16.69.28';
update user set Select_priv = 'Y' where user = 'redmine' and Host = '192.168.1.34';
flush tables;
```

```

11 mysql> create database redmine character set utf8;
13 Query OK, 1 row affected (0.00 sec)

15 mysql> show databases;
17 +-----+
17 | Database |
17 +-----+
19 | information_schema |
19 | mysql |
21 | redmine |
21 +-----+
23 3 rows in set (0.00 sec)

```

Con esta serie de comandos hemos creado un usuario llamado 'redmine', al cual le hemos dado permisos para utilizar las tablas de la base de datos, que además para acciones administrativas le vamos a permitir que se conecte en local, y desde dos IP's remotas, y grabamos todo y lo activamos con 'flush tables' sin tener la obligación de reiniciar la Base de datos.

Hemos conseguido con la configuración anterior de la MySQL que no se pueda acceder desde cualquier lugar de la red sino exclusivamente desde los lugares IP estipulados. Por ejemplo se puede dar de alta en MySQL una IP específica de la LAN a la que pertenece que será la única que podrá realizar acciones de administración, cualquier otro lugar no lo tendrá habilitado y no lo podrá realizar. Es una característica interna de MySQL que utilizo en esta instalación.

Para acabar configuraremos el acceso a la base de datos del RedMine en su fichero de configuración.

```

1 cd /opt/redmine_2.0.3/config
2 vim database.yml
3
4 production:
5 adapter: mysql2
6 database: redmine
7 host: localhost
8 username: redmine
9 password: "redmine"
10 encoding: utf8

```

A continuación podemos ver el acceso que permite esta Base de datos MySQL. En un principio se puede observar que las únicas conexiones permitidas son a los usuarios 'root' y 'redmine'.

```

mysql> select Host, User, Password, Select_priv from user;

```

Host	User	Password	Select_priv
localhost	root	*6FB42516CD14796D672425B675EFEDF668BF5C5F	Y
127.0.0.1	root	*6FB42516CD14796D672425B675EFEDF668BF5C5F	Y
127.0.0.1	redmine	*275589819C2DE801B002DFED28FBC0736109C358	Y
localhost	redmine	*275589819C2DE801B002DFED28FBC0736109C358	Y
tfc_redmine	redmine	*275589819C2DE801B002DFED28FBC0736109C358	Y
tfc_redmine	root	*6FB42516CD14796D672425B675EFEDF668BF5C5F	Y
192.168.1.34	redmine	*275589819C2DE801B002DFED28FBC0736109C358	Y
192.168.1.34	root	*6FB42516CD14796D672425B675EFEDF668BF5C5F	Y
172.16.69.28	root	*6FB42516CD14796D672425B675EFEDF668BF5C5F	Y
172.16.1.0	root	*6FB42516CD14796D672425B675EFEDF668BF5C5F	Y
172.16.69.28	redmine	*275589819C2DE801B002DFED28FBC0736109C358	Y
172.16.1.0	redmine	*275589819C2DE801B002DFED28FBC0736109C358	Y

```

18 12 rows in set (0.00 sec)

```

Las conexiones de root pueden realizarse desde el ordenador propio (localhost, 127.0.0.1) y desde la IP 172.16.69.28 de la LAN a la que pertenece el servidor. Al mismo tiempo ocurre casi lo mismo con

el usuario “redmine”, que puede conectarse desde localhost y desde dos únicas IP’s, la 172.16.69.28 y la 192.168.1.34. La 28 puede ser perfectamente la dirección del administrador, y la que queda, que es la 34, puede ser la del operador de informática que se dedica a realizar trabajos rutinarios de reservas y procesos Batch. Un nivel más refinado de seguridad nos lo pueden dar los puertos que posee la MySQL, por defecto escucha en el puerto “3306” y el resto de puertos en las conexiones de red de la máquina donde reside la Base de datos también pueden cambiarse. Esto se controla mediante reglas con IPTables que vamos a ver a continuación.

En este servidor solamente puede acceder a la base de datos el usuario root, desde localhost preferiblemente, y el usuario redmine, nadie más. Así de esta manera sin consumir recursos valiosos ni teniendo que adquirir tecnologías sofisticadas y caras podemos conseguir aumentar significativamente el nivel de seguridad de nuestra máquina.

16. IPTables.

Netfilter/iptables[24] es un framework disponible en el núcleo Linux que permite interceptar y manipular paquetes de red. Dicho framework permite realizar el manejo de paquetes en diferentes estados del procesamiento. Netfilter es también el nombre que recibe el proyecto que se encarga de ofrecer herramientas libres para cortafuegos basados en Linux.

El componente más popular construido sobre Netfilter es iptables, una herramienta de cortafuegos que permite no solamente filtrar paquetes, sino también realizar traducción de direcciones de red (NAT) para IPv4 o mantener registros de log. El proyecto Netfilter no sólo ofrece componentes disponibles como módulos del núcleo sino que también ofrece herramientas de espacio de usuario y librerías.

“iptables” es el nombre de la herramienta de espacio de usuario mediante la cual el administrador puede definir políticas de filtrado del tráfico que circula por la red. El nombre iptables se utiliza frecuentemente de forma errónea para referirse a toda la infraestructura ofrecida por el proyecto Netfilter. Sin embargo, el proyecto ofrece otros subsistemas independientes de iptables tales como el connection tracking system o sistema de seguimiento de conexiones, que permite encolar paquetes para que sean tratados desde espacio de usuario. iptables es un software disponible en prácticamente todas las distribuciones de Linux actuales.

Ahora crearemos las reglas de las IPTables que van a restringir bajo nuestras órdenes las comunicaciones en la red. Las IPTables están levantadas pero no funcionan porque hay que configurarlas. Con ‘iptables -L’ se puede ver las normas que hay activadas, con ‘iptables -F’ se pueden quitar las normas hasta el siguiente arranque.

La página web oficial de CentOS para las IPTables es:

IPTables en Cent OS <http://wiki.centos.org/es/HowTos/Network/IPTables?highlight=%28IPTables%29>

Donde se puede ver como configurar las IPTables y además en Español, es una página que no tiene desperdicio.

El fichero de configuración es el siguiente.

```
1 more /etc/sysconfig/iptables
```

y su contenido después de configurarlo es el siguiente.

```
1 # Firewall configuration written by system-config-firewall
2 # Manual customization of this file is not recommended.
3 *filter
4 INPUT DROP [0:0]
5 FORWARD DROP [0:0]
6 OUTPUT ACCEPT [0:0]
7
8 # -----
9 # Servicio Ping a esta máquina.
10 # -----
11 -A INPUT -i eth1 -p icmp -j ACCEPT
```

```

13 # -----
14 # No perder las conexiones ya establecidas.
15 # -----
16 -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
17
18 # -----
19 # Conexión ssh desde la red de la oficina y al Gateway
20 # -----
21 -A INPUT -s 172.16.1.0/24 -i eth1 -p tcp -m tcp --dport 22 -j ACCEPT
22 -A INPUT -s 192.168.1.0/24 -i eth1 -m tcp -p tcp --dport 22 -j ACCEPT
23
24 # -----
25 # Conexión web. Abrimos el puerto 80 del Apache.
26 # -----
27 -A INPUT -i eth1 -p tcp -m tcp --dport 80 -j ACCEPT
28
29 # -----
30 # Conexión externa a la DDBB MySQL para administración.
31 # -----
32 -A INPUT -i eth1 -p tcp -m tcp -s 172.16.1.0/24 --dport 3306 -j ACCEPT
33 -A INPUT -i eth1 -m tcp -p tcp --dport 3306 -s 192.168.1.0/24 -j ACCEPT
34 -A INPUT -i lo -j ACCEPT
35
36 COMMIT

```

Cuyo resultado en memoria después de cada arranque es el siguiente.

```

[root@rysredmine203 ~]# iptables -L -vn
2 Chain INPUT (policy DROP 3592 packets, 351K bytes)
3 pkts bytes target prot opt in out source destination
4 4 240 ACCEPT icmp -- eth1 * 0.0.0.0/0 0.0.0.0/0
6 167 13155 ACCEPT all -- * * 0.0.0.0/0 0.0.0.0/0 state RELATED,ESTABLISHED
8 0 0 ACCEPT tcp -- eth1 * 172.16.1.0/24 0.0.0.0/0 tcp dpt:22
9 0 0 ACCEPT tcp -- eth1 * 192.168.1.0/24 0.0.0.0/0 tcp dpt:22
10 5 300 ACCEPT tcp -- eth1 * 0.0.0.0/0 0.0.0.0/0 tcp dpt:80
11 0 0 ACCEPT tcp -- eth1 * 172.16.1.0/24 0.0.0.0/0 tcp dpt:3306
12 0 0 ACCEPT tcp -- eth1 * 192.168.1.0/24 0.0.0.0/0 tcp dpt:3306
13 0 0 ACCEPT all -- lo * 0.0.0.0/0 0.0.0.0/0
14
15 Chain FORWARD (policy DROP 0 packets, 0 bytes)
16 pkts bytes target prot opt in out source destination
17
18 Chain OUTPUT (policy ACCEPT 159 packets, 30278 bytes)
19 pkts bytes target prot opt in out source destination

```

17. LiveUSB, Características

Una distribución live o LiveCD[25] o LiveDVD, más genéricamente LiveDistro, traducido en ocasiones como CD vivo o CD autónomo, es un sistema operativo almacenado en un medio extraíble, tradicionalmente un CD o un DVD (de ahí sus nombres), que puede ejecutarse desde éste sin necesidad de instalarlo en el disco duro de una computadora, para lo cual usa la memoria RAM como disco duro virtual y el propio medio como sistema de archivos.

Como la creación y utilización de un liveCD es muy problemática me he decidido por realizar mi TFC sobre un liveUSB. Esto es debido a que el liveUSB tiene mas ventajas que el liveCD de siempre. Paso a enumerar las más importantes.

1. En Internet hay mucha mas información para poder crear un liveUSB, e incluso hay “tools” que facilitan el trabajo para llevarlo a cabo.



Figura 11: Fedora, Live-usb creator

2. También hay otra ventaja y es la persistencia. En un LiveUSB se pueden guardar modificaciones que se mantendrán operativas incluso después de haber apagado la máquina y las encontraremos activas en el siguiente encendido.

Para crear un Live USB es necesario: Bajar de Internet una imagen ISO y grabarse en este caso en una memoria USB. Acto seguido configurar la ROM de la computadora para que arranque desde la unidad “usb”, reiniciando luego con el mismo usb montado, con lo que el LiveUSB se iniciará automáticamente.

17.1. Grabación “liveUSB”

Como mi idea desde un principio era instalar un CentOS, me he decidido por esta herramienta “liveusb-creator”[26]. Su uso es muy sencillo. Se baja la aplicación desde la red, el último desarrollo que sea estable es la mejor idea, se instala, dando igual que sea desde un Windows o un Linux, y finalmente se arranca.

17.2. Pre-requisitos

Para poder grabar el LiveUSB necesitaré los elementos de la siguiente lista:

- Yo lo he realizado sobre un ordenador con un Windows 7. Se puede realizar también desde un Linux.
- $2\text{GB} \leq \text{USB Flash drive}$. Yo lo he grabado en una memoria de 4 Gigas de capacidad.
- CentOS 6.3 LiveCD.iso
- liveusb-creator-x-x.zip

La aplicación para grabar la imagen es muy simple, sólo tiene una pantalla, en ella cargaremos la imagen que queremos grabar. La aplicación por defecto apunta a imágenes Fedora y podemos elegir la que mas nos guste, pero como lo que quiero es una CentOS lo que tengo que hacer es:

1. Bajar lo primero la imagen a mi disco duro local.
2. Lo siguiente es elegir la unidad USB que deseamos que albergue nuestro sistema,

3. Acto seguido solo nos queda para terminar el poner la cantidad de memoria que queremos de persistencia. Como tengo una memoria grande, y voy a instalar abundantes paquetes de software he elegido una cantidad de 1 Giga.

Le damos a grabar, aparecerá una barra de progreso. Cuando acaba con la grabación de todas nuestras elecciones anteriores instala un bootloader para que pueda ser autoarrancable al encender el ordenador. Eso es todo, acto seguido se re-arranca el ordenador.

Por supuesto que en la BIOS del PC lo primero que hay que habilitar es que el ordenador busque un disco arrancable en una unidad USB.

Parte IV

Utilización de RedMine

18. Guía de instalación

18.1. Descargar RedMine

Accede a tu cuenta de alojamiento vía “*ssh*” y ejecuta los siguientes comandos en la consola para descargar y descomprimir RedMine:

```
1 cd
2 wget http://rubyforge.org/frs/download.php/27473/redmine-2.1.0.tar.gz
3 tar -xvzf redmine-0.6.0.tar.gz
4 rm -f redmine-2.1.0.tar.gz
```

18.2. Configuración

1. Crear una Base de datos MySQL
2. Crea una base de datos. Por ejemplo, asignarle el nombre blog, de forma que su nombre real completo sea usuario_blog.
3. Crea un usuario MySQL. Por ejemplo, asignarle el nombre redmine y una contraseña, de forma que su nombre completo sea usuario_redmine.
4. Por último, otorga permisos ALL al usuario MySQL que has creado (p.e.: usuario_redmine) sobre la base de datos que has creado (p.e.: usuario_blog).

18.3. Configuración de RedMine

Accede a tu cuenta vía “*ssh*” y ejecuta los siguientes comandos para crear el archivo de configuración de base de datos:

```
1 cd
2 cd redmine-0.6.0/config
3 cp -a database.example.yml database.yml
```

Ahora debes definir los valores de conexión a MySQL en el archivo “database.yml” que has creado, para ello edita el

archivo: nano database.yml

Y define lo siguiente:

```
1 Producción :
2 Adaptar: MySQL
3 Base de datos: usuario_blog
4 Nombre de usuario: usuario_redmine
5 Contraseña: JsuKop)2s.Kjhs28s
6 Servidor: local
7 host
```

Edita también el archivo de configuración environment.rb:

```
1 nano environment.rb
```

Y asegurarse de configurar el entorno en producción definiendo la siguiente línea:

```
1 ENV[RAILS_ENV] = producción
```

18.4. Crear la estructura de base de datos:

- Por último, debes crear la estructura y contenidos predeterminado de la base de datos, para ello ejecuta lo siguiente:

```
1 cd
  cd redmine -0.6.0
3 Rake db:bootstrap RAILS_ENV=production
```

18.5. BitNami, instalador todo en uno.

Bitnami[27] es una empresa privada residente en la nube que se especializa en facilitar el desarrollo de aplicaciones web. Muchas empresas confían en su experiencia para la instalación de un RedMine propio corporativo. Su producto se llama “Redmine Stack”, es un instalador multiplataforma con posibilidad de utilización en castellano, que simplifica la puesta en marcha de Redmine y sus dependencias. Puede ser ejecutado usando un instalador nativo, como máquina virtual, en la cloud, o como módulo sobre una instalación previa de un Stack de infraestructura.

Sus características principales son:

- Soporte de múltiples proyectos
- Control de acceso flexible basado en roles
- Sistema de gestión de casos flexible
- Diagrama de Gantt y calendario
- Gestión de noticias, documentos y ficheros
- Notificaciones por feeds y email
- Wiki por proyecto
- Foros por proyecto
- Funcionalidad simple de registro de tiempo
- Campos personalizables para los casos, proyectos y usuarios
- Integración con sistemas de control de versiones (SVN, CVS, Mercurial y Darcs)
- Soporte para autenticación LDAP
- Soporte para auto-registro
- Soporte multilinguaje
- Soporte para múltiples base de datos

Los instaladores nativos de BitNami, “Redmine Stack”, han sido empaquetados usando el instalador multiplataforma de BitRock[28].

19. Actualizar una instalación existente

19.1. Actualización

La documentación siguiente se basa en una actualización a Redmine 2.x

Step 1 - Check requirements El primer paso para ver si se puede actualizar es, comprobar los requerimientos necesarios para la versión que se desea instalar, es decir, comprobar si podemos instalar el paquete de software.

Step 2 - Backup Se recomienda que se hagan copias de seguridad de la Base de Datos y de los ficheros. La mayoría de actualizaciones son seguras, pero es mejor asegurarse el llevar a buen término la actualización, nunca está de más.

Todos los ficheros se almacenan en el directorio “files”. Realizar un BackUp de este directorio es tan sencillo como hacer una copia de su contenido en otro lugar.

MySQL El comando “mysqldump” se puede utilizar para salvaguardar el contenido de la base de datos MySQL. Un ejemplo puede ser este:

```
1 /usr/bin/mysqldump -u <username> -p<password> <redmine_database> | gzip > /path/to/backup/db/redmine_`date +%v_%n_%d`.gz
```

SQLite database La bases de datos SQLite están autocontenidas en un único fichero, de esta manera se puede guardar una copia de seguridad simplemente copiando el fichero a otro lugar.

PostgreSQL El comando ph_dump de Postgres es similar al mysqldump de MySQL, comando más conocido. Un ejemplo puede ser este:

```
1 /usr/bin/pg_dump -U <username> -Fc --file=redmine.sqlc <redmine_database>
```

Step 3 - Comencemos con la actualización Comencemos con la actualización. Este proceso es diferente dependiendo de como se haya bajado el paquete Redmine. Existen dos únicas opciones:

- Opción 1 - Bajar el paquete en formato (tar.gz or zip file)
 1. Descomprimir el archivo en un nuevo directorio.
 2. Copiar la configuración de la Base de datos que ya poseemos “config/database.yml” en el nuevo directorio de configuración. Si tenemos una Redmine >= 1.4 con MySQL y una versión de ruby superior a la 1.9, tendremos que cambiar el driver de la Base de datos MySQL de “mysql” a “mysql2”, sino no funcionará.
 3. Copiar el fichero de configuración “config/configuration.yml” en el nuevo directorio donde se almacenan las diversas configuraciones.

O si estamos actualizando desde una versión anterior a la 1.2.0, hay que copiar la configuración del mail desde el fichero “config/email” en el nuevo fichero de configuración situado en “config/configuration.yml”.
 4. Copiar los ficheros del directorio de contenidos en el nuevo directorio de la nueva instalación.
 5. Copiar también las carpetas de los plugins instalados del directorio “vendor/plugins”, en la nueva instalación que tiene un directorio propio en la carpeta raíz del Redmine.

6. Generar un fichero nuevo “*secret_token.rb*”

Antes de la versión 2.0.0, se necesitaba un fichero “*session_store.rb*” que se generaba en el directorio de las configuraciones del Redmine, para trabajar. A partir de la versión mencionada, el fichero no existe en la instalación, en su lugar, lanzar el siguiente comando.

```
1 rake generate_secret_token
```

Esto nos va a generar un fichero (*config/initializers/secret_token.rb*) con un número aleatorio secreto para utilizarlo en las sesiones seguras de datos. Es una especie de clave de seguridad secreta. En el momento en que esté operativo RedMine, si se cambia este número dejaría todo de funcionar. Se supone que nadie tiene que acceder a este fichero, ni siquiera “*root*”, ni “*admin*”. No se debe tocar bajo ningún concepto.

7. Comprobar por algún tema que tuvieras instalado en la versión antigua en el directorio “*public/themes*” para pasarlos a la versión nueva. También habrá que comprobar si todavía son compatibles.

Y sobre todo ALGO MUY IMPORTANTE: No escribir el fichero de configuración “*config/settings.yml*” con el fichero antiguo, son totalmente incompatibles, la instalación de RedMine no funcionará.

■ Opción 2 - Bajar la versión desde algún SVN con un Checkout

1. Entrar en el directorio raíz de Redmine y lanzar el siguiente comando.

```
1 svn update
```

2. Si estás actualizando Redmine de una versión anterior a la 0.8.7, debes generar un número secreto para almacenar las cookies.

Step 4 - Update the database Este paso es el único que puede cambiar los contenidos de la base de datos.

Nos vamos al directorio raíz de la nueva instalación de Redmine y lanzamos el siguiente comando de “*rake*”.

```
1 rake db:migrate RAILS_ENV=production
```

Si también hemos instalado plugins, estamos obligados a lanzar el siguiente comando, que complementa al anterior en la carga de información en la base de datos MySQL.

```
1 rake redmine:plugins:migrate RAILS_ENV=production
```

Step 5 - Limpieza general de la instalación

1. Para limpiar la instalación deberíamos borrar la cache y las sesiones que se han quedado abiertas.

```
1 rake tmp:cache:clear
   rake tmp:sessions:clear
```

2. Reiniciar el servidor, (e.g. puma, thin, passenger), en mi caso va a ser Apache que además tendrá cargados unos módulos de passenger.

3. Para acabar deberemos ir a "Admin -> Roles & permissions" para dar de alta check/set permissions para las nuevas funcionalidades.

Cuestiones comunes.

Error en los roles de los miembros, “member_roles” file. Si has tenido una migración que te falló en el pasado, entonces podrías tener una tabla “member_roles” y/o “group_users” creada. La migración de la Base de datos a una versión superior nos dará entonces problemas. La solución es sencilla, se entra en MySQL a través de su cliente y se le cambia los nombres a las tablas mencionadas con el siguiente comando.

```
mysql> rename table member_roles to member_roles_saved
mysql> rename table groups_users to groups_users_saved
```

Error sobre “undefined method ‘add_frozen_gem_path’” Si ves este error, debes comprobar si existe el directorio “vendor/rails” y si existe se debe borrar o cambiar de nombre. Podría ser por ser una versión antigua de la versión RoR.

20. Guía del administrador

20.1. Configuración de la aplicación.

20.1.1. Mostrar

Tema Esta opción le permite elegir un tema personalizado. Redmine se entrega con dos temas adicionales, además del tema por defecto:

alternate, que principalmente ofrece coloración cuestión lista basada en temas prioritarios.

clásico, que se deriva de la Redmine 0.5.1 diseño y trae una clásica apariencia.

20.1.2. Autenticación

Se requiere autenticación Si esta opción está activada, no hay ninguna página de la aplicación accesible a usuarios anónimos. Los usuarios deben firmar para acceder a la aplicación. Valor predeterminado: No

Autologin Esta opción permite a los usuarios utilizar la función de acceso automático. Valor predeterminado: Deshabilitado

20.1.3. Proyectos

Los nuevos proyectos son públicos por defecto Es el estado por defecto de los proyectos de nueva creación. El proyecto todavía se puede hacer que no sea público, mientras se crea un nuevo proyecto o incluso después de la creación del proyecto.

Generar identificadores secuenciales de proyecto Este ajuste le permitirá proponer a Redmine identificadores secuenciales del proyecto para usted. Esto todavía se puede cambiar manualmente sólo al crear el proyecto, no después.

Rol para un usuario que no es administrador que ha creado un proyecto Define qué papel se le da de forma predeterminada para un usuario que no es administrador y que crea un proyecto (esto sólo se aplica cuando se ha configurado permisos Redmine de tal manera que los usuarios que no sean administradores poseen en realidad el privilegio de poder crear proyectos).

20.1.4. Seguimiento de asuntos

Permita diversas relaciones entre proyectos

Si se establece en true, las relaciones entre los problemas de diferentes proyectos pueden ser creados.

20.1.5. Repositorios

Recopilación automática: Si esta opción está activada, la aplicación recupera automáticamente las nuevas revisiones cuando un usuario consulta el repositorio.

Puede desactivar esta opción y automatizar la llamada al repositorio `# fetch_changesets` utilizando un cron para recuperar las revisiones regularmente para todos los depósitos en el fondo.

20.2. Roles y permisos

Los Roles permiten definir los permisos que tienen los miembros de un proyecto. Cada miembro de un proyecto tiene un rol o múltiplos(s) de rol en el proyecto. Un usuario puede tener diferentes funciones para diferentes proyectos.

Puede crear funciones nuevas o editar las existentes. Puede eliminar una función sólo si nadie tiene ese papel en un proyecto.

20.2.1. Roles propiedades

- Nombre: Visualiza el nombre del rol.
- Las cuestiones pueden ser asignados a este rol: si no está activado, los problemas no se pueden asignar a los miembros de este rol en un proyecto.
- Visibilidad de los problemas: suponemos que este rol incluye el permiso necesario y se puede ver, las siguientes reglas se aplican a los problemas de los proyectos:
 - Todos los problemas: el usuario puede ver todos los temas. Este es el valor predeterminado.
 - Todos los problemas que no están marcados como privados: el usuario puede ver todas las cuestiones que no estén marcados como privados.
 - Temas creados por o asignados al usuario: el usuario sólo puede ver los problemas creados o asignados a él/ella.

20.2.2. Funciones del sistema

Hay dos funciones del sistema en Redmine: "No ser miembro" y "Anonymous". Se utilizan para uso interno del Redmine y por lo tanto no se puede eliminar.

No soy miembro: esta función le permite definir los permisos que tiene un usuario registrado en proyectos que él no es un miembro.

Anónimo: esta función le permite definir los permisos que tienen los usuarios anónimos en los proyectos.

Tenga en cuenta que estas dos funciones sólo se aplican a los proyectos públicos ya que los usuarios anónimos y los usuarios que no son miembros de un proyecto privado, ni siquiera puede verlo.

21. Guía del usuario

21.1. Cuentas de usuario

21.1.1. Mi cuenta

Esta pantalla le permite editar los ajustes de su cuenta: apellido, nombre, dirección de correo electrónico, idioma.

The screenshot shows a 'My account' page with three main sections:

- Information:** Includes input fields for 'Firstname *', 'Lastname *', and 'Email *', a 'Language' dropdown menu set to 'English', and a 'Save' button.
- Email notifications:** Features a dropdown menu with the selected option 'Only for things I watch or I'm involved in' and a checkbox labeled 'I don't want to be notified of changes that I make myself' which is currently unchecked.
- Preferences:** Includes a checked checkbox for 'Hide my email address', a 'Time zone' dropdown menu set to '(GMT-05:00) Eastern Time', and a 'Display comments' dropdown menu set to 'In chronological order'.

On the right side of the page, there is a 'Change password' link, a 'My account' header, and user details: 'Login: [redacted]', 'Created: [redacted] 12:49', and 'RSS access key created 6 months ago (Reset)'. A yellow 'Donate' button is also visible.

Figura 12: Mi cuenta.

Notificaciones por correo Si se ha marcado Notificaciones por correo, recibirá notificaciones sobre cualquier evento que se produzca en los proyectos a los que pertenezca (problema añadido o editado, nuevo documento, ...).

Si no se ha marcado, sólo recibirá notificaciones sobre los problemas en los que esté involucrado (autor o asignado) y sobre las cosas que vigile (por ejemplo problemas, foros).

Contraseña Para cambiar su contraseña, teclee su contraseña antigua, y la nueva dos veces. La longitud de la contraseña debe ser de entre 4 y 12 caracteres.

Si su cuenta usa un medio de autenticación externo (LDAP), no puede cambiar su contraseña en Redmine.

Mi página Esta página le permite mostrar varios bloques de información sobre sus proyectos. Por omisión, sólo están habilitados los siguientes dos bloques:

- Peticiones asignadas a mí
 - proporciona una lista de las peticiones de todos los proyectos asignadas al usuario actual, mostrando el identificador, proyecto, rastreador y asunto de la petición, junto con su estado actual entre corchetes
- Peticiones enviadas
 - proporciona una lista de las peticiones de todos los proyectos enviadas por el usuario actual, mostrando el identificador, proyecto, rastreador y asunto de la petición, junto con su estado actual entre corchetes

Cada usuario puede personalizar su propia «Mi página» pulsando en «Personalizar esta página». Entonces el usuario puede elegir cuáles de los bloques de información disponibles se mostrarán o no. Los bloques adicionales además de los dos predeterminados son:

- Calendario
 - proporciona un vistazo del calendario semanal para todos los proyectos
- Documentos
 - proporciona un vistazo de los últimos documentos de todos los proyectos
- Últimas noticias
 - proporciona un vistazo de las últimas noticias de todos los proyectos

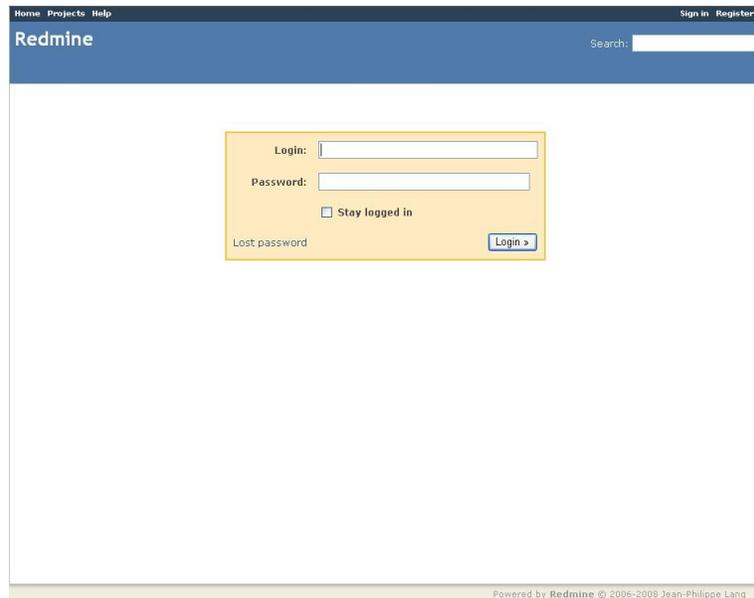


Figura 13: Mi login.

- Tiempo empleado
 - proporciona un vistazo de las entradas de tiempo empleado del usuario actual durante los últimos siete días en todos los proyectos
- Peticiones vigiladas
 - proporciona una lista de todas las peticiones de todos los proyectos vigiladas por el usuario actual, mostrando el identificador, proyecto, rastreador y asunto de la petición, junto con su estado actual entre corchetes

La disposición (dónde se colocan los bloques) de «Mi página» también se puede reordenar y modificar arrastrando y soltando.

Contraseña perdida Si olvida su contraseña, hay un procedimiento que le permite elegir una nueva. En la pantalla de inicio, pulse en Contraseña perdida. Introduzca su dirección de correo electrónico y envíe el formulario. Entonces se le enviará un mensaje, que contendrá un enlace que le permitirá cambiar su contraseña.

Observe que el administrador puede inhabilitar esta funcionalidad. Si su cuenta usa un medio de autenticación externo (LDAP), este procedimiento no está disponible.

Registro Al registrarse, puede obtener una cuenta sin la intervención del administrador. En la pantalla de inicio, pulse Registrarse. Rellene el formulario y envíelo. Se le enviará un mensaje. Para activar la cuenta, use el enlace que hallará en dicho mensaje.

Observe que el administrador puede inhabilitar esta funcionalidad.

21.2. Ingreso

La página “Ingresar” se usa para ingresar en el proyecto para el que haya sido activado. El enlace “Contraseña perdida” sólo se muestra si el administrador lo ha habilitado.

Figura 14: Mi registro.

21.3. Registro

La página de registro se usa para registrar un nuevo usuario. Un usuario nuevo tiene que dar la información sobre su Nombre de usuario, Contraseña, Nombre, Apellido y dirección de Correo electrónico y el lenguaje que desea usar en la interfaz de usuario.

Proceso de registro Basándose en los Ajustes de Redmine, el usuario tendrá que activar su cuenta por correo electrónico, esperar a que un administrador valide su cuenta o ver su cuenta activada automáticamente.

Activación de la cuenta por correo electrónico Tras proporcionar la información necesaria en la página de registro, el usuario recibirá un correo electrónico en la dirección proporcionada en la página de registro. Al pulsar en el enlace de activación facilitado en el mensaje, el propio usuario habilita su cuenta.

Este método se usa con frecuencia para evitar el spam.

Activación manual de la cuenta Tras proporcionar la información necesaria en la página de registro, el usuario debe esperar a la aprobación de un administrador. El administrador, al ingresar en el panel de administración, habilitará la cuenta del usuario. Cuando la cuenta del usuario haya sido habilitada, se le permitirá ingresar.

Activación automática de la cuenta Tras proporcionar la información necesaria en la página de registro, se le permite al usuario ingresar en cualquier momento, sin requerirse ningún paso adicional.

21.4. Búsquedas en Redmine

Parte de cualquier proceso de gestión de proyectos se emplea en buscar entre la información existente. Redmine cuenta para ello con una función de búsqueda.



Figura 15: Búsqueda.

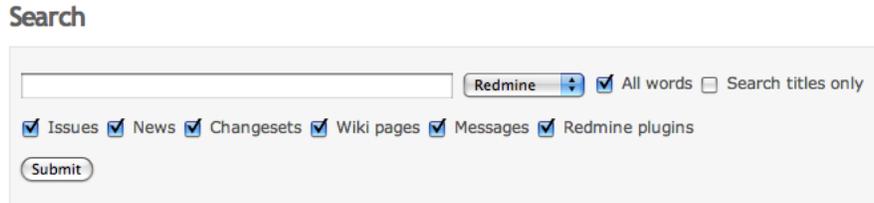


Figura 16: Búsqueda avanzada.

21.4.1. Caja de búsquedas

En la parte superior derecha de Redmine hay una sencilla caja de búsqueda. Teclee cualquier cosa y pulse Retorno para buscar.

Si busca `#{IDPetición}` o `{IDPetición}` irá directamente a la petición. Los parámetros de búsqueda se pueden encerrar entre comillas dobles.

Dependiendo de los ajustes de los permisos y la membresía en proyectos, un menú desplegable puede mostrar el proyecto actual. La búsqueda simple busca el texto completo en el proyecto mostrado actualmente en ese selector.

Al pulsar la palabra “*Buscar*” se le llevará inmediatamente a la búsqueda avanzada y la página de resultados. Observe que el selector de proyectos permanece, y que hay cajas de selección para limitar la búsqueda a tipos de entrada específicos. Las cajas de selección se muestran sólo si se ha habilitado el módulo a ese nivel del proyecto.

21.4.2. Configuración de la búsqueda

Buscar sólo en los títulos Desde la caja de búsqueda avanzada, se puede habilitar “*Buscar sólo en los títulos*” para limitar los resultados.

Campos de búsqueda personalizados No olvide establecer la caja de selección “*Buscable*” de los campos personalizados a Verdadero para buscar también en esos campos.

21.4.3. Resultados

Ésta es una página de resultados de muestra de este sitio sobre "petición de pruebas".

- Muchos enlaces para ir a lo que se encontró.
- Realce de texto único por los términos de búsqueda.
- Capacidad para saltar a grupos de resultados (por ejemplo, entradas de Noticias).
- Iconos al principio para categorizar el tipo de resultado (por ejemplo control de cambios, petición, comentario).

21.4.4. Controlador de la búsqueda

Compruebe siempre la fuente lo primero:

source:/trunk/app/controllers/search_controller.rb

Un vistazo más avanzado al controlador explica parte del comportamiento anterior:

Results (1453) Issues (735) Messages (638) Wiki pages (55) Changesets (23) News (2)

[Help: RE: Stages - what to do next?](#)
Yes they are, you can edit what statuses a ce ... ply what changes to the status, see [[RedmineIssueTrackingSetup#Workflow]]. You could for examp ... "verified" to "in progress" or "resolved", a tester can set it from "resolved" to "tested", and the manager can set it from "tested" to "committed" or "closed". If you don ... , you'll probably need a custom field for it.
 2010-11-07 16:14

[Open discussion: RE: How to contribute](#)
Eric Davis wrote: > Anton Nepomnyaschih wrot ... Any code contributions should be sent to the issue tracker, either as an attachment to an existing issue or a new "Patch" issue. Right now, we really need some people to help triage the issues list. Like review patches to see if they apply to the latest trunk, if they have tests, etc. > > Eric If the wiki is a probl ... are similar but not the same. So I'd add an issue relation between all three. That would help ... cted, don't spam. See [[ForumCodeOfConduct]]
 2010-11-05 20:35

[Revision 4367: Allow key authentication when deleting issues \(with tests\) #6447](#)
 2010-11-05 18:49

[Revision 4366: Allow key authentication when updating issues \(with tests\) #6447](#)
 2010-11-05 18:49

[Revision 4365: Allow key authentication when creating issues \(with tests\) #6447](#)
 2010-11-05 17:29

Figura 17: Resultados.

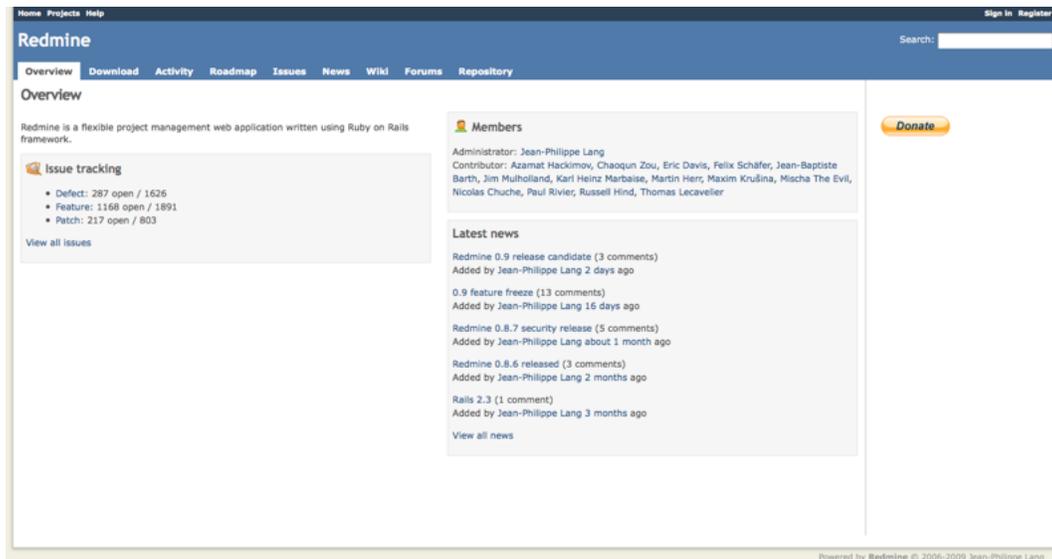


Figura 18: Redmine Project Overview.

- Sólo busca los primeros 5 elementos de una búsqueda
- Los elementos deben tener una longitud mínima de 2 caracteres
- Los resultados se devuelven en conjuntos paginados de 10

21.5. Vista del proyecto

La vista del proyecto le facilitará una perspectiva general de todo el proyecto. Arriba a la izquierda, en el área “*Peticiones*” obtendrá una visión global de cuántas tareas hay abiertas y cerradas para cada rastreador indicado para el proyecto. En el área “*Miembros*” puede ver quienes son los miembros del proyecto en cuestión y en particular quién es el administrador del proyecto. En el área Últimas noticias puede ver cuáles son las últimas noticias para este proyecto concreto.

En el lado superior derecho puede ver “*Ingresar*” y “*Registrarse*”. El enlace “*Ingresar*” permite ingresar a quien tenga una cuenta para este proyecto. El enlace “*Registrarse*” permite que los usuarios no registrados se registren por sí mismos, permitiéndoles ser parte de cualquier proyecto contenido en esta instalación de Redmine.

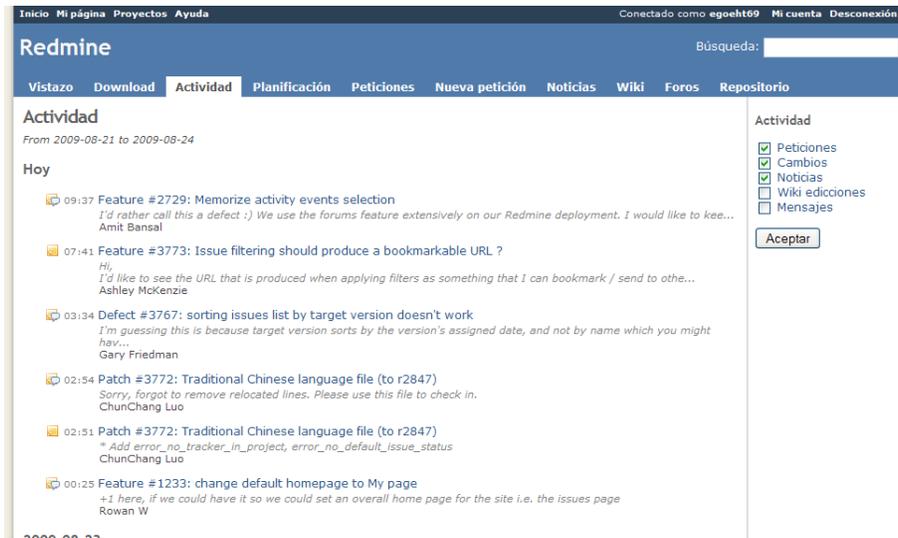


Figura 19: Actividad del proyecto presente.

Observe que el administrador puede inhabilitar el enlace “*Registrarse*”.

21.6. Actividad del proyecto

La página Actividad le dará un registro histórico de al menos todas las actividades que se han producido en el proyecto indicado.

Esto puede incluir:

- Se ha creado una nueva petición (funcionalidad, defecto...).
- Se ha cambiado el estado de la petición.
- Se ha cerrado la petición en cuestión.
- Se ha añadido una nota a la petición.
- Esto significa que se ha enviado un registro de cambios al repositorio.
- Se ha añadido una noticia al área de Noticias.
- Se ha añadido un mensaje al área del foro.
- Se ha añadido una respuesta a un mensaje del foro.
- Se ha editado o creado una página del wiki.
- Se ha añadido un nuevo fichero al área de ficheros o a un documento.
- Se ha añadido un nuevo documento. Se han registrado algunas horas.
- Se ha añadido un nuevo proyecto.

Usando las cajas de selección de la barra lateral de la derecha, puede elegir qué elementos se pueden mostrar.

La actividad también está disponible como una fuente RSS Atom.

En la página de “*Proyectos*” está disponible una vista de actividad multiproyecto, que lista la actividad de todos los proyectos visibles.

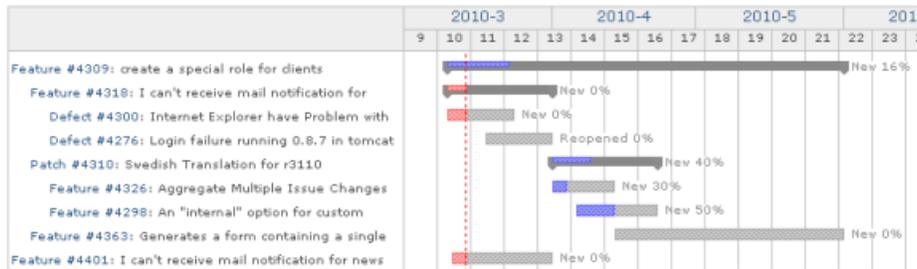


Figura 20: Diagrama de Gantt.

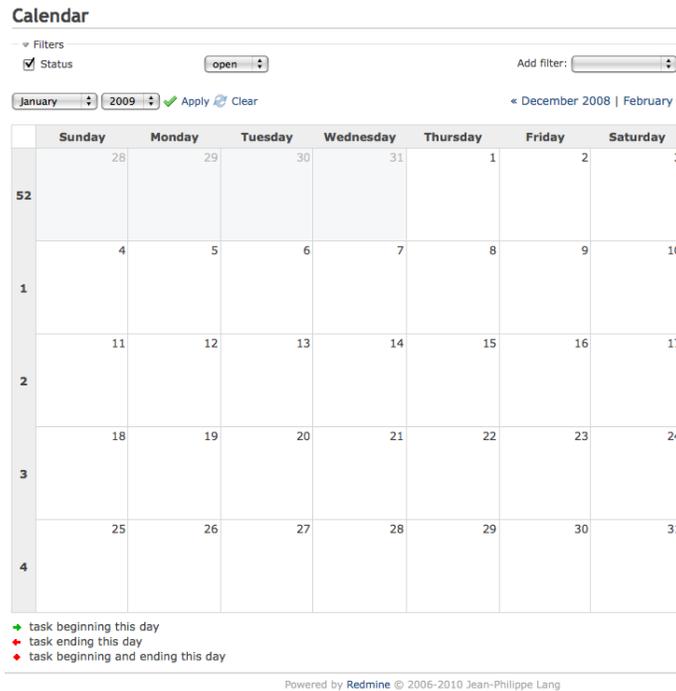


Figura 21: Calendario.

21.7. Diagramas de Gantt

El diagrama de Gantt nos muestra las peticiones que se han introducido con una fecha de comienzo y una estimada de terminación, o bien las mismas están asignadas o son dependientes de otras que si tienen control por fecha.

21.8. Calendario

El calendario nos muestra el proyecto presente en una visión mensual. Esta vista nos muestra todas las peticiones que tienen la misma fecha de comienzo y lo muestra por pantalla.

21.8.1. Filtros

Como en la lista de peticiones “Issues-lists” y en el diagrama de Gantt “Gant chart”, aquí también podemos definir filtros cuyo resultado nos aparecerá en el calendario. Estos filtros son similares a los que tenemos en cualquier lista de peticiones “Issues-lists”.

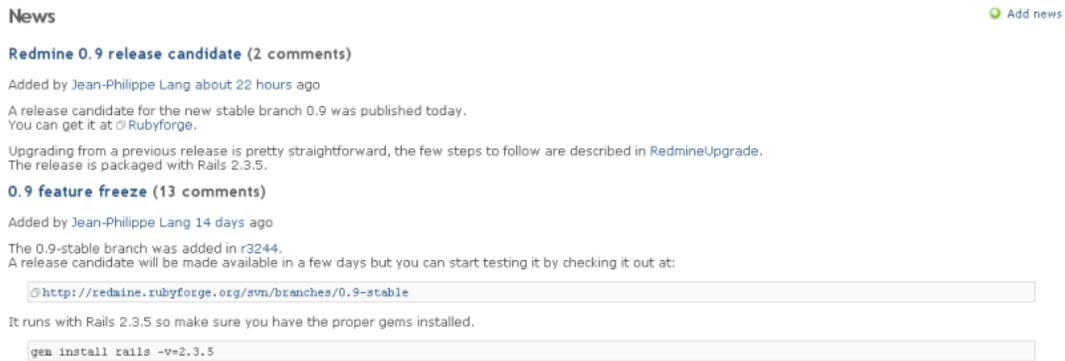


Figura 22: Noticias.

21.9. Noticias

En el área de noticias se puede publicar asuntos relacionados con el proyecto.

A la noticia se le puede dar un título, un sumario y una descripción detallada. El sumario se mostrará en la zona del proyecto donde la descripción aparecerá si se le da un click al título de la noticia que se desea consultar.

Si se posee permisos para crear noticias, entonces se puede añadir/editar/borrar una entrada. Esto se puede saber porque aparece una cruz (green plus sign) de color verde que significa que puedes añadir una nueva noticia. Si la presionas con un click aparecerá un formulario donde se puede completar el título, sumario y descripción de una nueva noticia.

21.10. Documentos Redmine

21.10.1. Introducción

Añadir documentos es un módulo que tiene Redmine de serie.

21.10.2. Añadir Documentos

En la pestaña “*Documentos*” se presiona el link “*Nuevo Documento*”. Por defecto existen dos tipos distintos de documentos.

1. Documentos propiedad de los usuarios.
2. Documentos Técnicos.

Estas categorías se pueden modificar y también se pueden añadir otras distintas por el Administrador de RedMine.

21.10.3. Visor de Documentos

Desde el momento en que se ha subido algún documento, se utiliza el título como el link que te permite después bajarlos, o incluso sirve también para añadir alguno más.

21.11. Ficheros en Redmine

21.11.1. Módulo de ficheros.

Cada proyecto puede tener el módulo de ficheros activado/desactivado vía a través del menú “*set-tings*”.

Una tabla que puede ser ordenada nos muestra los ficheros que han sido recogidos por Redmine. Puede haber ficheros idénticos que se diferencian con versiones distintas. Las versiones se representan gracias a un número hash de tipo MD5.

New document New document

Category: User documentation

Title *

Description

Text formatting: Help

Files: Choose File no file selected Optional description

Add another file (Maximum size: 15.6 MB)

Create Cancel

Figura 23: Documento Nuevo.

Documents

User documentation

Developer provided user documentation

09/23/2010 05:15 pm

This is the developer provided user documentation.

It outlines

1. How to do things
2. Who to contact for help

Management Provided Documentation

09/23/2010 05:16 pm

Useless documentation

Technical documentation

Systems Documentation

09/23/2010 05:17 pm

How to recover from disaster.

Figura 24: Documentos.

Files + New file

File ▲	Date	Size	D/L	MDS	
sample.exe	09/23/2010 03:58 pm	116 Bytes	0	7b7c58caf906b329a4306090eea1b2da	🗑
sample.zip	09/23/2010 03:58 pm	623 KB	2	254194061f53693e9313e2a1e6807157	🗑

Figura 25: Ficheros almacenados.

Files Choose File no file selected Optional description

[Add another file](#) (Maximum size: 15.6 MB)

Figura 26: Fichero nuevo a subir.

Los nuevos ficheros se añaden a través de un botón que reside en la parte superior derecha con forma de cruz verde “plus”.

21.11.2. Almacenamiento de los ficheros.

Todos los ficheros se almacenan en el directorio

```
${REDMINE_HOME}/files
```

A los ficheros se le añade un ID único para su almacenamiento..

21.11.3. Formulario de nuevo fichero

Si se añade un nuevo fichero, se utiliza el formulario 26. Se pueden añadir múltiples ficheros de una sola vez utilizando el link de “Añadir otro fichero”.

21.12. Foros

Los foros permiten a los usuarios de un proyecto el comunicarse con cada uno de los componentes del mismo. El administrador define los “foros” que van a existir en la parte de configuración/administración del proyecto. La lista de “foros” en el proyecto nos permite ver todo lo que está acaeciendo en cada uno de ellos de forma resumida.

- el número total de temas.
- el número total de mensajes.
- un link que apunta al último mensaje del foro.

Forums

Forum	Topics	Messages	Last message
Open discussion General discussion	836	2930	Added by jacob briggs about 1 month ago LDAP auth modification to sync LDAP groups and Redmine gr...
Help Get public help	1669	5016	Added by J Kanoyer about 1 month ago Ability to set the assigned to field as required?
Plugins Announcements or discussions about Redmine plugins	231	1872	Added by Obivan Chinobi about 1 month ago RE: Using Amazon S3 for storage

Also available in: Atom

Figura 27: Foros de discusión.

Statistics | Revision:

Name	Size	Revision	Age	Author	Comment
branches		3296	about 3 hours	Jean-Philippe Lang	Merged r3295 from trunk.
0.6-stable		1233	about 1 year	Jean-Philippe Lang	Docs updated for 0.6.4 release.
0.7-stable		2203	about 1 year	Jean-Philippe Lang	Updates CHANGELOG and version for 0.7.4 release.
0.8-stable		3081	about 1 month	Jean-Philippe Lang	Backported r3080 from trunk (#4248).
0.9-stable		3296	about 3 hours	Jean-Philippe Lang	Merged r3295 from trunk.
nbc		2464	10 months	Nicolas Chuche	
swistak		2058	about 1 year	Liwiusz Ociepa	Fix access to Repository Parent Path (FORBIDDEN ...
work		2356	11 months	Jean-Philippe Lang	Branches cleaning.
plugins		2721	8 months	Jean-Philippe Lang	Fixes menu tab and settings partial name of sim...
sandbox		2868	3 months	Jean-Philippe Lang	Highlight new group users.
tags		3297	about 3 hours	Jean-Philippe Lang	tagged version 0.9.0 (release candidate)
trunk		3295	about 3 hours	Jean-Philippe Lang	CHANGELOG update.

Latest revisions

#	Date	Author	Comment
3297	2010-01-09 13:03	Jean-Philippe Lang	tagged version 0.9.0 (release candidate)
3296	2010-01-09 12:50	Jean-Philippe Lang	Merged r3295 from trunk.
3295	2010-01-09 12:49	Jean-Philippe Lang	CHANGELOG update.
3294	2010-01-09 12:47	Jean-Philippe Lang	Merged r3293 from trunk.
3293	2010-01-09 12:46	Jean-Philippe Lang	Updated doc.
3292	2010-01-09 11:41	Jean-Philippe Lang	Locales updated to trunk @ r3291.
3291	2010-01-09 11:36	Jean-Philippe Lang	Translation updates.
3290	2010-01-09 11:27	Jean-Philippe Lang	Merged r3289 from trunk.
3289	2010-01-09 11:27	Jean-Philippe Lang	Fixed: undefined method `stringify_keys!' error in Issue#attributes_with_tracker_first= with ruby1.9 (#4540).
3288	2010-01-09 11:06	Jean-Philippe Lang	Translation update.

[View differences](#)

[View all revisions](#)

Figura 28: Repositorio del proyecto.

21.12.1. Añadir un tema

Para añadir un tema en un foro determinado, se clickea en el nombre del mismo, después en “nuevo mensaje”. Entonces ahora se puede introducir un tema, un cuerpo y añadir ficheros al mensaje. Existen dos opciones distintas.

sticky: si está activada, el tema se mostrará en lo mas alto de la lista en negrita.

locked: si está activada los usuarios que la lean no podrán contestar al mensaje.

22. Repositorio

La pestaña del repositorio te permite navegar a través del repositorio del proyecto y comprobar las últimas actualizaciones. Se puede expandir un directorio dándole un click en el mismo. También se puede entrar en un directorio dándole un click encima de su nombre. Si el click se lo haces en un número determinado de revisión, lo que consigues es ver todos los detalles del *'commit'* específico. Si quieres entrar directamente en el directorio de una revisión cuyo número ya conoces puedes hacerlo tecleándolo en la parte superior-derecha y darle al *'Enter'*.

Parte V

Conclusiones.

23. Redmine, gestor de proyectos de código libre para nuestras empresas

Organizar la información que guardamos es uno de los caballos de batalla de la empresa moderna ya que a poco que nos empeñemos podemos sufrir el síndrome de Diógenes digital y acabar por ir añadiendo discos duros y discos duros aumentando la capacidad de almacenamiento de forma indefinida. Al final debemos buscar soluciones y una de las que podemos implementar para organizar mejor tanto la información como el trabajo es Redmine, gestor de proyectos de código libre para nuestras empresas.

Redmine es un gestor de proyectos moderno y vivo, en plena evolución. Su mayor ventaja que nos ofrece es su capacidad para la organización. La gran flexibilidad y facilidad para la personalización que posee Redmine es una de sus características más apreciadas. Se puede crear una lista particular de funcionalidades que deben ser integradas en el proyecto, permite llevar un control preciso de lo que ocurre, así como los errores que puedan ir apareciendo, al igual que una gestión óptima de los recursos. Los proyectos de Redmine pueden ser utilizados en empresas grandes puesto que soporta distintos perfiles de usuario, que pueden trabajar en diversos proyectos diferentes.

Redmine hace uso de Ruby on Rails. Como base de datos soporta tanto MySQL como PostgreSQL o SQLite. La principal ventaja que nos aporta como gestor de proyectos es poder tener toda la información asociada a un proyecto acotada dentro del mismo. Además nos permitirá el control de la ejecución del mismo, todo ello a través de una interfaz web que hace sencilla la gestión de los mismos.

Redmine es multiplataforma, está disponible para Linux, Windows y Mac.

24. Principales características de Redmine

Redmine soporta distintos tipos de proyectos. Los usuarios que acceden tienen distintas funciones según el rol asignado, ya sea como usuario, jefe de proyecto, administrador, etc. todo ello en función de un sistema de permisos. Cada proyecto puede llevar asociado si así lo deseamos documentos, archivos o noticias. Además se establece un sistema de notificaciones para los usuarios mediante correo electrónico ya sea porque le han asignado una tarea o porque una parte del proyecto ha cambiado o se ha actualizado.

A cada proyecto le podemos asociar un Wiki que nos permite generar contenidos rápidamente, y editarlos y gestionarlos de forma colaborativa. También tenemos la posibilidad de tener un foro por proyecto, lo que sin duda contribuye en gran medida a reducir el nivel de correos que nos cruzamos entre los participantes en el mismo, lo que sin duda contribuye a mejorar la productividad de cada uno de los miembros del proyecto.

Por último otra característica importante es el control de errores y cambios de versiones. Además tenemos la posibilidad de reflejar el desarrollo del proyecto a través de diagramas de gant, una herramienta muy visual que nos ayudará a controlar los progresos del mismo. Una herramienta ya clásica en todo gestor de proyectos que se precie.

25. Iniciando el trabajo con Redmine

Lo primero que debemos hacer cuando tenemos instalado Redmine es iniciar el trabajo de backoffice y dar de alta los usuarios, con sus distintos roles y permisos, jefes de proyecto, desarrolladores, etc. aunque también pueden darse de alta ellos mismos a través de la interfaz web, para que después el administrador les apruebe el acceso.

El siguiente paso es dar de alta un proyecto y asignarle un jefe. Una vez completada esta fase podemos comenzar a establecer los hitos o etapas del mismo. Si vamos añadiendo las previsiones de

Nuevo proyecto

Nombre *

Descripción Formato de texto

Identificador *
 Longitud entre 1 y 100 caracteres. Letras minúsculas (a-z), números y signos de puntuación permitidos.
 Una vez guardado, el identificador no puede modificarse.

Sitio web

Público

Módulos

<input checked="" type="checkbox"/> Peticiones	<input checked="" type="checkbox"/> Control de tiempo	<input checked="" type="checkbox"/> Noticias
<input checked="" type="checkbox"/> Documentos	<input checked="" type="checkbox"/> Ficheros	<input checked="" type="checkbox"/> Wiki
<input checked="" type="checkbox"/> Repositorio	<input checked="" type="checkbox"/> Foros	<input checked="" type="checkbox"/> Calendario
<input checked="" type="checkbox"/> Gantt		

Figura 31: Proyecto de RedMine.

tiempo de cada una de las tareas podemos obtener el gráfico de gant de para cada hito. Cada tarea se asignará a un miembro del equipo, que cuando se conecten a la página del proyecto podrán visualizar las que tienen asignadas. A medida que van marcando tareas como terminadas, en el gráfico del hito del proyecto se van rellenando de color verde, de manera que podemos controlar de esta manera de un golpe de vista como va el progreso.

Todo proyecto tiene una fase de ajustes, donde se descubren pequeños fallos que pueden corregirse, de manera que una vez aislados podemos asignarlos a un miembro del proyecto para que ejecute la tarea pertinente. Esta es una cuestión habitual en el mundo del desarrollo de software pero que no está exenta el resto de empresas, además que es habitual encontrar flecos por cerrar, aun después de haber cobrado y cerrado el proyecto.

26. Conclusiones

Redmine nos puede ayudar en gran medida a gestionar de manera ágil todos los proyectos de nuestra empresa, la documentación asociada, los archivos que anexamos y todo ello controlando en cada momento la ejecución del mismo. Su administración es sencilla y permite definir flujos de trabajo que evitarán que la gestión nos inunde la bandeja de entrada con un correo por cada pequeño detalle a discutir en el mismo.

Para probarlo tenemos la opción de descargarlo e instalarlo a través de Bitnami, que nos permitirá utilizarlo de forma fiable en local para después si lo consideramos adecuado llevarlo al servidor de manera que lo ponemos disponible para todos los usuarios de la empresa. Sin duda merece la pena hacer la prueba.

Parte VI

Anexos

A. Nomenclatura

Nomenclatura

ApplicationController	Clase especial del Framework Rails. Es la parte controladora del modelo MVC.
CSV	Los ficheros CSV (del inglés comma-separated values) son un tipo de documento en formato abierto sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas (o punto y coma en donde la coma es el separador decimal: España, Francia, Italia...) y las filas por saltos de línea. Los campos que contengan una coma, un salto de línea o una comilla doble deben ser encerrados entre comillas dobles.
DDBB	Data Base en plural; Bases de datos.
GIT	Git es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente.
GNU v2.	La Licencia Pública General de GNU o más conocida por su nombre en inglés GNU General Public License o simplemente sus siglas del inglés GNU GPL, es una licencia creada por la Free Software Foundation en 1989 (la primera versión, escrita por Richard Stallman), y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.
HTTP	Hypertext Transfer Protocol o HTTP (en español protocolo de transferencia de hipertexto) es el protocolo usado en cada transacción de la World Wide Web.
IMAP	Internet Message Access Protocol, o su acrónimo IMAP, es un protocolo de aplicación de acceso a mensajes electrónicos almacenados en un servidor. Mediante IMAP se puede tener acceso al correo electrónico desde cualquier equipo que tenga una conexión a Internet.
JSP	Java Server Page.
KVM	Kernel-based Virtual Machine o KVM, (en español, Máquina virtual basada en el núcleo) es una solución para implementar virtualización completa con Linux. Está formada por un módulo del núcleo (con el nombre kvm.ko) y herramientas en el espacio de usuario, siendo en su totalidad software libre. El componente KVM para el núcleo está incluido en Linux desde la versión 2.6.20.
MIME	Multipurpose Internet Mail Extensions o MIME (en español extensiones multi-propósito de correo de internet) son una serie de convenciones o especificaciones dirigidas al intercambio a través de Internet de todo tipo de archivos (texto, audio, vídeo, etc.) de forma transparente para el usuario.
MTA	Mail Transfer Agent. El típico servidor de correo electrónico que existe en los servidores.
MUA	Mail User Agent. La típica aplicación de correo electrónico.
MVC	Modelo, Vista, Controlador

MySQL	MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. ¹ MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.
NAT	NAT (Network Address Translation - Traducción de Dirección de Red) es un mecanismo utilizado por routers IP para intercambiar paquetes entre dos redes que asignan mutuamente direcciones incompatibles. Consiste en convertir, en tiempo real, las direcciones utilizadas en los paquetes transportados.
NTP	Network Time Protocol. Con este protocolo se puede poner en hora los relojes del servidor con un reloj atómico que lanza una señal de sincronización en Internet.
POP3	En informática se utiliza el Post Office Protocol (POP3, Protocolo de Oficina de Correo o Protocolo de Oficina Postal) en clientes locales de correo para obtener los mensajes de correo electrónico almacenados en un servidor remoto. Es un protocolo de nivel de aplicación en el Modelo OSI.
PostgreSQL	PostgreSQL es un SGBD relacional orientado a objetos y libre, publicado bajo la licencia BSD. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una empresa y/o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, altruista, libre y/o apoyados por organizaciones comerciales. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).
Rails	Ruby on Rails, también conocido como RoR o Rails es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC).
Redmine	Redmine es una herramienta para la gestión de proyectos que incluye un sistema de seguimiento de incidentes con seguimiento de errores entre otras muchas mas herramientas.
REST	La Transferencia de Estado Representacional (Representational State Transfer) o REST es una técnica de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web. El término se originó en el año 2000, en una tesis doctoral sobre la web escrita por Roy Fielding, uno de los principales autores de la especificación del protocolo HTTP y ha pasado a ser ampliamente utilizado por la comunidad de desarrollo.
RHEL	Red Hat Enterprise Linux también conocido por sus siglas RHEL es una distribución comercial de Linux desarrollada por Red Hat. Es la versión comercial basada en Fedora que a su vez está basada en el anterior Red Hat Linux.
RSS	RSS son las siglas de Really Simple Syndication, un formato XML para indicar o compartir contenido en la web. Se utiliza para difundir información actualizada frecuentemente a usuarios que se han suscrito a la fuente de contenidos. El formato permite distribuir contenidos sin necesidad de un navegador, utilizando un software diseñado para leer estos contenidos RSS (agregador).
Ruby	Ruby es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro Matz Matsumoto, quien comenzó a trabajar en Ruby en 1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en Python y Perl con características de programación orientada a objetos similares a Smalltalk.

SGBDR	Sistema Gestor de Base de Datos Relacional.
SMTP	El Simple Mail Transfer Protocol (SMTP) (Protocolo para la transferencia simple de correo electrónico), es un protocolo de la capa de aplicación. Protocolo de red basado en texto, utilizados para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos (PDA, teléfonos móviles, etc.). Está definido en el RFC 2821 y es un estándar oficial de Internet.
SQLite	SQLite es un sistema de gestión de bases de datos relacional compatible con ACID, contenida en una relativamente pequeña (275 kiB) biblioteca escrita en C. SQLite es un proyecto de dominio público creado por D. Richard Hipp.

B. Instrucciones de uso

Al arrancar el sistema operativo aparece en el Desktop un icono que al darle doble click nos arranca una página HTML. Esta página aparece en el navegador con instrucciones aclaratorias de uso. En este Anexo las repito para darle un mayor alcance.

B.1. Accesos

Los logines y password's para acceder al sistema son los siguientes.

Accesos	LOGIN	PASSWORD
Sistema Operativo: CentOS 6.3	root	'felisa'
	estudiante-uoc	'UOC' <i>En mayusculas.</i>
Base de Datos MySQL	root	'blanquita'
	redmine	'redmine'
Redmine	admin	'admin'

B.2. Redmine

El acceso a Redmine se puede hacer escribiendo en el navegador cualquiera de las siguientes direcciones url;

- http://localhost.tfc_redmine
- http://localhost
- http://127.0.0.1

O encontrándose en la misma LAN introduciendo la IP que se le ha asignado mediante el protocolo DHCP. Si se tuviera una IP fija se podría meter en las tablas del DNS para así de esta manera poder acceder al RedMine mediante un nombre canónico desde cualquier lugar de la red LAN.

B.3. Login

El login es automático. Me ha parecido una complicación innecesaria el poner un login de entrada con password. Esto es un trabajo de desarrollo y no es un trabajo sobre seguridad. Creo que de esta manera facilito su evaluación.

B.4. IPTables

El estado de las “*IPTables*” se puede contrastar con el comando:

```
1 iptables -L
```

Las “*IPTables*” se pueden deshabilitar momentáneamente en la presente sesión con el comando:

```
1 iptables -F
```

En el siguiente arranque volverán a estar activas otra vez.

C. Bibliografía de URL's

Referencias

- [1] M. USB, La página de wikipedia que explica el liveusb.
URL http://en.wikipedia.org/wiki/Live_USB 5
- [2] M. USB, Wikipedia, comparativa de liveusb's.
URL http://en.wikipedia.org/wiki/List_of_tools_to_create_Live_USB_systems/ 5
- [3] CentOS, Versiones y documentación de centos.
URL http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/index.html 5.2
- [4] CentOS, Descripción detallada de las nuevas funcionalidades que posee la nueva versión 6.3.
URL <http://www.centos.org/> 5.2
- [5] CentOS, Descripción detallada de las nuevas funcionalidades que posee la nueva versión 6.3.
URL <http://wiki.centos.org/> 5.2
- [6] CentOS, Descripción detallada de las nuevas funcionalidades que posee la nueva versión 6.3.
URL <http://lists.centos.org/> 5.2
- [7] CentOS, Foros de discusión de centos-6. lugar donde se habla sobre los temas que interesa a la comunidad centos.
URL <http://bugs.centos.org/> 5.2
- [8] CentOS, Foros de discusión de centos-6. lugar donde se habla sobre los temas que interesa a la comunidad centos.
URL <http://planet.centos.org/> 5.2
- [9] CentOS, Foros de discusión de centos-6. lugar donde se habla sobre los temas que interesa a la comunidad centos.
URL <http://mirrors.kernel.org/redhat/redhat/linux/enterprise/6Client/en/os/SRPMS/> 5.2
- [10] CentOS, Foros de discusión de centos-6. lugar donde se habla sobre los temas que interesa a la comunidad centos.
URL <http://mirrors.kernel.org/redhat/redhat/linux/enterprise/6Server/en/os/SRPMS/> 5.2
- [11] Ruby, La página de wikipedia donde explica el lenguajes ruby.
URL <http://es.wikipedia.org/w/index.php?title=Ruby&printable=yes> 6
- [12] Ruby, La página oficial de ruby.
URL <http://www.rubyonrails.org.es/> 7
- [13] Ruby, La página oficial de las gemas de ruby.
URL <http://rubygems.org/> 8
- [14] Ruby, Utilidad para gestionar las dependencias de los distintos paquetes de ruby; "gemas".
URL <http://gembundler.com/v0.9/index.html> 9
- [15] Git, La página de wikipedia donde explica lo que es git.
URL <http://es.wikipedia.org/wiki/Git> 10
- [16] MUA, La página de wikipedia donde explica lo que es un "mail user agent".
URL http://es.wikipedia.org/wiki/Cliente_de_correo_electrónico 11.1
- [17] MTA, La página de wikipedia donde explica lo que es un "mail transfer agent".
URL http://es.wikipedia.org/wiki/Servidor_de_correo_electrónico 11.2

- [18] RFC2821, La página oficial de la ieee donde explica el protocolo smtp.
URL <http://tools.ietf.org/html/rfc2821> 11.2
- [19] NTP, La página oficial del "network time protocol".
URL <http://www.ntp.org/> 12
- [20] Apache, La página de wikipedia donde explica lo que es el servidor apache.
URL http://es.wikipedia.org/wiki/Servidor_HTTP_Apache 13
- [21] MySQL, La página web de mysql.
URL <http://es.wikipedia.org/wiki/MySQL> 14
- [22] Redmine, La página de wikipedia donde explica lo que es el redmine.
URL <http://es.wikipedia.org/wiki/Redmine> 15
- [23] Redmine, La página oficial de redmine.
URL <http://www.redmine.org/> 15
- [24] IPTables, La página de wikipedia donde explica lo son las reglas iptables.
URL <http://es.wikipedia.org/wiki/Netfilter/iptables> 16
- [25] LiveCD, La página de wikipedia donde explica el livecd.
URL <http://es.wikipedia.org/wiki/LiveCD> 17
- [26] M. U. creator, Página donde reside el liveusb creator que he utilizado.
URL <https://fedorahosted.org/liveusb-creator/> 17.1
- [27] Bitnami, Empresa privada que ofrece un instalador sencillo de redmine, pero de pago.
URL <http://bitnami.org/es/stack/redmine> 18.5
- [28] I. Bitnami, Instalador bitnami.
URL <http://installbuilder.bitrock.com/index.html> 18.5