

Instalación Redmine-Emergya

Documentación preparada para:



Empresa: Emergya Consultoría

The logo for Emergya Consultoría, featuring a stylized orange 'E' followed by the word 'EMERGYYA' in a purple, sans-serif font.

Autor / Procedencia.	Antonio Fernández. Mánager de Negocio.
Equipo de Trabajo	Área de Desarrollo de Negocio.
Versión	V1.0
Fecha:	09/03/12



ISO 9001:2000



ISO 14001:2000



CMMI – Nivel 2



1. ÍNDICE DE CONTENIDO

- 1 Requerimientos..... 3**
 - 1.1 Sistema operativo..... 3
 - 1.2 Ruby, Ruby on Rails y rack..... 3
 - 1.3 Compatibilidad..... 4
 - 1.4 Base de Datos..... 4
 - 1.5 Componentes opcionales..... 5
- 2 Versión de Redmine..... 6**
- 3 Procedimiento para la instalación..... 7**
 - 3.1 Obtención del código fuente..... 7
 - 3.2 Creación de base de datos..... 7
 - 3.3 Configuración opciones de base de datos..... 7
 - 3.4 Generar un almacén de la sesión..... 8
 - 3.5 Creación estructura base de datos..... 8
 - 3.6 Introducción datos configuración por defecto en base de datos..... 9
 - 3.7 Configuración de los permisos..... 9
 - 3.8 Utilización cuenta predeterminada del administrador para el inicio de sesión..... 10
- 4 Configuración..... 11**
 - 4.1 Configuración del servidor de correo SMTP..... 11
 - 4.2 Configuración de SCM..... 11
 - 4.3 Configuración de almacenamiento de datos adjuntos..... 11
 - 4.4 Configuración del registro..... 11
 - 4.5 Backups..... 12
- 5 Actualización a una nueva versión..... 13**



1 REQUERIMIENTOS

1.1 Sistema operativo

Redmine debería funcionar en la mayoría de los sistemas operativos Unix, Linux, Mac, servidor de Mac y Windows, siempre y cuando Ruby esté disponible en esa plataforma. Ver HowTos específica de instalación aquí.

1.2 Ruby, Ruby on Rails y rack

Las versiones requeridas de Ruby y Ruby on Rails para una determinada versión de Redmine son:

Versión de RedMine	Versiones de Ruby Soportadas	Versión requerida de Rails	Versiones soportadas de Rack	Versiones soportadas de Rubygems
current trunk	ruby 1.8.7, 1.9.2, 1.9.3	Rails 2.3.14	Rack 1.1.x	RubyGems <= 1.8
1.3.x	ruby 1.8.6, 1.8.7	Rails 2.3.14	Rack 1.1.x	RubyGems <= 1.7
1.2.x	ruby 1.8.6, 1.8.7	Rails 2.3.11	Rack 1.1.x	RubyGems <= 1.6
1.1.x	ruby 1.8.6, 1.8.7	Rails 2.3.5	Rack 1.0.1	RubyGems <= 1.4
1.0.x	ruby 1.8.6, 1.8.7	Rails 2.3.5	Rack 1.0.1	RubyGems <= 1.4
0.9.x	ruby 1.8.6, 1.8.7	Rails 2.3.5	Rack 1.0.1	RubyGems <= 1.4
0.8.x	ruby 1.8.6, 1.8.7	Rails 2.1.2		

Las versiones oficiales incluyen la versión de Rails apropiada en su directorio correspondiente por lo que sólo necesita instalar las versiones correctas de Ruby + RubyGems + Rake + Rack.

Si no, si obtiene la fuente desde el repositorio de Redmine, usted tendrá que instalar la versión de Rails requerida en el equipo (con `gem install rails - version = <version requerida>`).

1.3 Compatibilidad

- Ruby 1.9 no es compatible aún. Se requiere usar Ruby 1.8.x como se indicó anteriormente.
- Se requiere Rake 0.8.7 (versión 0.9.x no es compatible con Rails aún)
- Es necesario Rack 1.1.x , 1.1.0 tiene un error haciendo uso de comillas simples). La migración de base de datos fallaría con otra versión.
- Mongrel 1.1.5 necesita un parche para funcionar bien con Rails 2.3.11. En el caso de actualización, otro problema puede aparecer por algún tiempo después de la migración (# 7857).
- I18n 0.4.2 se requiere para Redmine >= 1.0.5
- Rails 2.3.11 hasta 2.3.14 sufren de un grave problema con el envío de correos electrónicos a varios destinatarios (cuando estos superan la cifra de 8). Ver <http://www.redmine.org/issues/8751> para más detalles y solución.

1.4 Base de Datos

- MySQL 5.0 o superior (recomendado).

Hay que asegurarse de instalar los enlaces de C para Ruby que mejoran el rendimiento. Se puede obtener mediante la ejecución de "gem install mysql". Si se tiene problemas para instalar la gema mysql ir a <http://wiki.rubyonrails.org/database-support/mysql>

- PostgreSQL 8

Hay que asegurarse de que el DateStyle de su base de datos de se establece en la norma ISO (ajuste de PostgreSQL por defecto).

- Se puede configurar para que se establezca ejecutando: ALTER DATABASE "redmine_db" SET dateStyle = "ISO, MDY";
- Algunos errores en PostgreSQL 8.4.0 y 8.4.1 afectan al comportamiento de Redmine (<http://www.redmine.org/issues/4259>, <http://www.redmine.org/issues/4314>), se solucionan en PostgreSQL 8.4.2.
- Hay que instalar el adaptador de PostgreSQL para Ruby. Se puede conseguir mediante la ejecución (uno de los siguientes):

gem install postgres-pr (nativo de la aplicación de Ruby no activamente mantenida por su autor, presenta problemas con 9.x)
 gem install pg (funciona con conexiones SSL, puede necesitar bibliotecas de desarrollo y paquetes para su sistema operativo antes de que se compile)



Ubuntu OS: bibliotecas de desarrollo se puede instalar con (o similar): sudo aptitude install libdbd-pg-ruby1.8

- SQLite 3.

1.5 Componentes opcionales

- Binarios del SCM (por ejemplo, svn), para la navegación de repositorios (debe estar disponible en su PATH).

Ver <http://www.redmine.org/projects/redmine/wiki/RedmineRepositories> para la compatibilidad y los requisitos del SCM.

- RMagick (<http://rmagick.rubyforge.org/>) para permitir la exportación de Gantt a imagen .jpeg)
- Ruby OpenID Library (<http://www.janrain.com/openid-enabled>) (para habilitar el soporte de OpenID) [sólo en Redmine trunk / 0.9-dev] Requerida Versión 2 o superior.

2 VERSIÓN DE REDMINE

Se recomienda a la mayoría de los usuarios instalar las versiones puntuales de Redmine. Redmine actualmente publica una nueva versión cada 6 meses, y estas publicaciones son consideradas muy usables y estables. No se recomienda instalar Redmine del tronco, a menos que esté muy familiarizado con Ruby on Rails y se encuentre siempre al día con los cambios – El Tronco se rompe a menudo.

3 PROCEDIMIENTO PARA LA INSTALACIÓN

3.1 Obtención del código fuente

Obtener el código fuente de Redmine ya sea a través de la descarga de una versión empaquetada o revisando el repositorio de código donde se encuentre. Por ejemplo en:

http://rubyforge.org/frs/?group_id=1850

3.2 Creación de base de datos

Crear una base de datos vacía y nombre de usuario de Redmine por ejemplo.

Para MySQL:

```
create database redmine character set utf8;
create user 'redmine'@'localhost' identified by 'my_password';
grant all privileges on redmine.* to 'redmine'@'localhost';
```

Para las versiones de MySQL anteriores a 5.0.2 - omita el paso de la "creación de usuario y en su lugar:

```
grant all privileges on redmine.* to 'redmine'@'localhost' identified by 'my_password';
```

Para PostgreSQL:

```
CREATE ROLE redmine LOGIN ENCRYPTED PASSWORD 'my_password' NOINHERIT VALID
UNTIL 'infinity';

CREATE DATABASE redmine WITH ENCODING='UTF8' OWNER=redmine;
```

3.3 Configuración opciones de base de datos

Copia el archivo config/ database.yml.example a config / database.yml y edita este archivo con el fin de configurar las opciones de la base de datos para el entorno de producción

Ejemplo para una base de datos MySQL

```
production:
  adapter: mysql
  database: redmine
```

```
host: localhost
username: redmine
password: my_password
```

Si el servidor no se está ejecutando en el puerto estándar (3306), utiliza esta configuración en su lugar:

```
production:
  adapter: mysql
  database: redmine
  host: localhost
  port: 3307
  username: redmine
  password: my_password
```

Ejemplo para una base de datos PostgreSQL (puerto por defecto)

```
production:
  adapter: postgresql
  database: <your_database_name>
  host: <postgres_host>
  username: <postgres_user>
  password: <postgres_user_password>
  encoding: utf8
  schema_search_path: <database_schema> (default - public)
```

3.4 Generar un almacén de la sesión

```
rake generate_session_store
```

3.5 Creación estructura base de datos

Crear la estructura de la base de datos, ejecutando el comando siguiente en el directorio raíz de la aplicación:

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


Se crearán tablas y una cuenta de administrador.

Si sale este error:

```
Rake aborted!
no such file to load -- net/https
```

es necesario instalar libopenssl-ruby1.8, en Ubuntu, de la siguiente forma:

```
apt-get install libopenssl-ruby1.8
```

3.6 Introducción datos configuración por defecto en base de datos

Introduzca los datos de configuración por defecto en la base de datos, ejecutando el comando siguiente:

```
RAILS_ENV=production rake redmine:load_default_data
```

Este paso es opcional pero altamente recomendado, ya que puede definir su propia configuración a partir de cero. Se cargará por defecto roles de usuarios, registros, estados de flujos de trabajo y etiquetas.

3.7 Configuración de los permisos

Nota: Los usuarios de Windows tienen que saltarse esta sección.

El usuario que ejecuta Redmine debe tener permiso de escritura en los siguientes subdirectorios: files, log,tmp (crear el último si no está presente).

Suponiendo que se ejecuta Redmine con un usuario Redmine:

```
mkdir tmp public/plugin_assets
sudo chown -R redmine:redmine files log tmp public/plugin_assets
sudo chmod -R 755 files log tmp public/plugin_assets
```

1. Comprobación de la instalación ejecutando el servidor web WEBrick :

```
ruby script/server webrick -e production
```

Una vez que ha lanzado WEBrick, dirija su navegador a <http://localhost:3000/>. Ahora debería ver la página de bienvenida de la aplicación.



Nota: WEBrick no es adecuado para el uso normal, utilice únicamente WEBrick para probar que la instalación ha llegado correctamente hasta este punto. No se recomienda el uso de webrick para otra cosa que el desarrollo. Utilice una de las muchas guías existentes para configurar Redmine utilizar cualquiera de los pasajeros (aka mod_rails aka) o Mongrel para servir su Redmine

3.8 Utilización cuenta predeterminada del administrador para el inicio de sesión

- login: admin
- password: admin

Usted puede ir a “Admin & Settings” para modificar la configuración de la aplicación.

4 CONFIGURACIÓN

Desde la versión 1.2.0, la configuración de Redmine se definen en un archivo llamado `config / configuration.yml`.

Si se necesita, para reemplazar la configuración predeterminada de la aplicación, basta con copiar `config / configuration.yml.example` a `config / configuration.yml`, a continuación editar el nuevo archivo, el archivo está bien comentado por sí mismo, por lo que se debe echar una ojeada al mismo.

Esta configuración se puede definir por entorno de Rails (producción / desarrollo / prueba).

Importante: no se olvide de reiniciar la aplicación después de cualquier cambio.

4.1 Configuración del servidor de correo SMTP

4.2 Configuración de SCM

4.3 Configuración de almacenamiento de datos adjuntos

4.4 Configuración del registro

Redmine se encuentra por defecto a un nivel de registro de: información, escribiendo en el registro de subdirectorio..... Dependiendo de la utilización del sitio, esto puede ser una gran cantidad de datos y para evitar que el contenido del archivo de registro siga creciendo sin límite, se considera rotar esos datos, ya sea a través de una utilidad de sistema como `logrotate` o mediante el fichero `config / additional_environment.rb`.

Para utilizar este último método, copia `config / additional_environment.rb.example` a `config / additional_environment.rb` y agregue las líneas siguientes.

```
#Logger.new(PATH, NUM_FILES_TO_ROTATE, FILE_SIZE)
config.logger = Logger.new(config.log_path, 2, 1000000)
config.logger.level = Logger::INFO
```

4.5 Backups

Copias de seguridad de Redmine deben incluir:

- datos (almacenado en su base de datos Redmine)
- archivos adjuntos (almacenado en el directorio de archivos de su instalación Redmine)

Aquí hay un simple script de shell que se puede utilizar para copias de seguridad diarias (suponiendo que usted está utilizando una base de datos mysql):

```
# Database
/usr/bin/mysqldump -u <username> -p<password> <redmine_database> |
gzip > /path/to/backup/db/redmine_`date +%y_%m_%d`.gz

# Attachments
rsync -a /path/to/redmine/files /path/to/backup/files
```



5 ACTUALIZACIÓN A UNA NUEVA VERSIÓN



Nuevo Tajamar, 555 – Piso 6 – Las Condes
Santiago de Chile
Tel. +562 427391700
www.emergya.cl