

راهنمای نصب و راه اندازی

سامانه‌ی مدیریت پروژه‌ی Redmine

نام نگارنده	تاریخ نگارش	تغییرات	نسخه
امیر خلیلی	۹۹/۱۲/۱۲	• تکمیل اولین نسخه‌ی پیش‌نویس	۰,۱,۰
امیر خلیلی	۰۰/۱۰/۲۲	• به روزرسانی دریافت RVM • به روزرسانی نصب Passenger • انتشار نسخه‌ی اولیه	۱,۰,۰
امیر خلیلی	۰۰/۱۰/۲۶	• اصلاحات مربوط به بخش نصب Bundler • افزودن بخش «ورود به سامانه»	۱,۰,۱

فهرست

۳	قواعد نگارش
۴	مقدمه
۵	نصب و پیکربندی پیش‌نیازها
۵	نصب سیستم عامل
۵	نصب و پیکربندی پایگاه داده
۸	نصب Ruby
۹	نصب Phusion Passenger
۹	پیکربندی Nginx
۱۲	ساخت حساب Redmine
۱۳	نصب بسته‌های پیش‌نیاز
۱۳	نصب Redmine
۱۳	دریافت فایل‌های نصب
۱۴	پیکربندی دسترسی به MySQL
۱۵	نصب Gem
۱۶	دستورات Rake
۱۷	نصب افزونه‌ها
۱۸	افزونه‌ی چاپک
۱۸	افزونه‌ی More previews
۱۸	ورود به سامانه

قواعد نگارش

در این مستند قالب نگارش برخی اصطلاحات و نکات فنی که نیازمند توضیحات بیشتری بوده‌اند به صورت متمایز از قالب متن عادی مشخص شده‌اند. در ادامه انواع قالب‌های استفاده شده و مورد استفاده‌ی آن‌ها را مشاهده می‌کنید:

الف. قالب اصطلاحات ذکر شده در واژه‌نامه: اصطلاحاتی که در متن به صورت توپُر^۱ نوشته شده‌اند، در بخش واژه‌نامه دارای توضیحات کامل و احياناً ترجمه‌ی لاتین می‌باشند. قالب این اصطلاحات به صورت زیر می‌باشد:

- **به این صورت**

ب. قالب بخش‌های قابل توجه: در برخی از موارد کاربرد این مستند، لازم بوده تا کاربر توجه ویژه‌ای روی جزئیات روند اجرا داشته باشد. قالب این موارد به صورت زیر می‌باشد:

- به این صورت

پ. قالب بخش نکته: مواردی که خارج از مراحل انجام سناریوها بوده ولی ممکن است با خطای کاربر، روند استفاده از نرم‌افزار را با مشکل مواجه کند، به صورت «نکته» و با قالب متفاوت از متن، مشخص شده‌اند. قالب نگارش نکته‌ها به صورت زیر می‌باشد:

نکته: به این صورت

-

ت. قالب رشته‌های قابل جایگزینی: در متن مستند عباراتی وجود دارند که در هنگام استفاده‌ی هر کاربر ممکن است با عبارت نوشته شده در مستند مغایر باشد. به طور معمول این عبارات مربوط به پیکربندی‌های محلی هر رایانه می‌باشد (مانند نشانی IP). این عبارات که کاربر می‌بایستی با توجه به پیکربندی اختصاصی خود جایگزین کند با قالب زیر از متن معمولی مشخص شده‌اند:

- **به این صورت**

^۱ Bold

^۲ String

مقدمه

در این مستند نحوه‌ی نصب و راه‌اندازی سامانه‌ی مدیریت پروژه‌ی Redmine در سیستم عامل Ubuntu **16.04** شرح داده می‌شود.

نصب و پیکربندی پیش‌نیازها

نکته: در تمامی مراحل نصب سامانه در صورتی که دستورات را از روی این مستند رونوشت و جایگذاری می‌کنید، از جایگزینی مناسب کاراکترها (مانند «» و ...) اطمینان حاصل نمایید. در ترمینال Ubuntu در مواردی لازم است کاراکترهای دستورات به صورت دستی تایپ شود.

نکته: در طول مستند هر دستوری که با کاراکترهای # (برای دسترسی Root) و \$ (برای دسترسی عادی کاربران)، شروع شده است لازم است در خط فرمان سیستم عامل Ubuntu اجرا شود.

نصب سیستم عامل

مراحل نصب سیستم عامل Ubuntu به صورت عادی انجام می‌شود. تنها نکته‌ی مورد نیاز ایجاد یک پارتیشن جدا برای استفاده‌ی Redmine و پایگاه داده‌ی متناظر می‌باشد که در برخی اوقات ممکن است بیشتر از 500 GB فضا نیاز داشته باشد. در این مستند به طور مثال پارتیشن زیر را:

```
/data
```

با حجم ۱ ترابایت برای نصب Redmine به پارتیشن‌های سیستم عامل اضافه می‌کنیم.

نصب و پیکربندی پایگاه داده

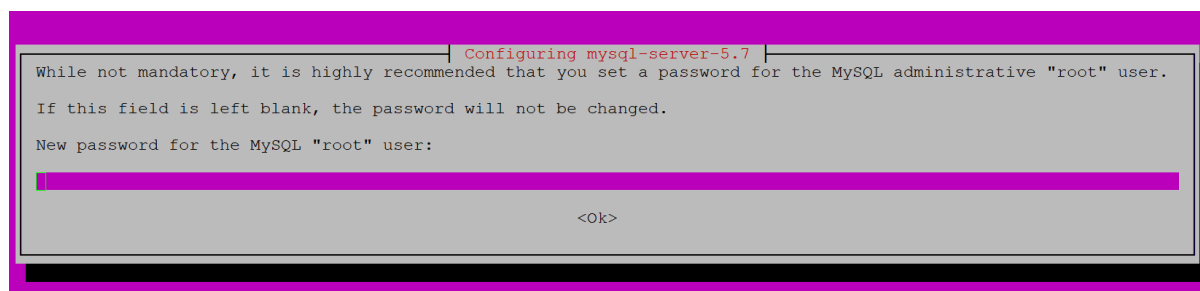
برای راه‌اندازی پایگاه داده‌ی مورد نیاز سامانه‌ی Redmine امکان استفاده از PostgreSQL و MySQL وجود دارد. در این مستند نحوه‌ی نصب و پیکربندی MySQL شرح داده می‌شود.

نکته: تمامی مراحل نصب و پیکربندی در ترمینال Ubuntu اجرا می‌شود.

الف. در ترمینال خط فرمان Ubuntu دستور زیر را اجرا می‌کنیم.

```
# apt-get update
# apt-get install -y mysql-server libmysqlclient-dev
```

به این ترتیب فایل‌های مورد نیاز نصب پایگاه داده، از اینترنت دریافت و نصب می‌شوند. در حین مراحل نصب گذرواژه‌ی کاربر مدیر پایگاه داده از شما پرسیده می‌شود. در این کادر گذرواژه‌ی دلخواه را وارد نمایید (شکل ۱).



شکل ۱ – انتخاب گذرواژه‌ی کاربر مدیر

در کادر بعدی مجدداً از شما درخواست می‌شود که گذرواژه را تکرار نمایید. پس از ورود مجدد عبارت قبلی مراحل نصب را پیگیری نمایید.

ب. پایگاه داده‌ی مورد نیاز را با استفاده از دستورات زیر ایجاد و پیکربندی می‌کنیم.

- ابتدا به کنسول پایگاه داده ورود می‌کنیم

```
# mysql -p -u root
```

پس از ورود دستور بالا لازم است گذرواژه‌ای که در مرحله‌ی قبل تعیین کردیم را وارد کنیم. به این ترتیب کنسول پیکربندی پایگاه داده‌ی MySQL باز می‌شود.

- در گام بعدی پایگاه داده را با اطلاعات دسترسی زیر ایجاد می‌کنیم:

```
CREATE DATABASE redmine CHARACTER SET utf8;  
CREATE USER 'redmine'@'localhost' IDENTIFIED BY 'yourpassword';  
GRANT ALL PRIVILEGES ON redmine.* TO 'redmine'@'localhost';
```

توجه داشته باشید که عبارت **yourpassword** را با گذرواژه‌ی دلخواه خود جایگزین نمایید.

پ. در گام بعدی لازم است فضای ذخیره‌سازی پایگاه داده را به پارتیشن مورد نظر انتقال دهیم.

- به این ترتیب ابتدا سرویس پایگاه داده را متوقف می‌کنیم.

```
# systemctl stop mysql
```

- پرونده‌ی پیکربندی را در مسیر زیر ویرایش می‌کنیم

```
/etc/mysql/mysql.conf.d/mysqld.cnf
```

- در این پرونده مسیر **datadir** را از مسیر **/var/lib/mysql** به **/data/mysql** تغییر می‌دهیم

```
datadir = /data/mysql
```

- پرونده‌های ایجاد شده‌ی پیش‌فرض پایگاه داده را به مسیر ایجاد شده انتقال می‌دهیم

```
# mv /var/lib/mysql/ /data/mysql/
```

- AppArmor را به روزرسانی می‌کنیم.

```
# vim /etc/apparmor.d/usr.sbin.mysqld
```

- خط‌های زیر را در بخش «Allow system resource access» اضافه نمایید:

```
/sys/devices/system/node/ r,  
/sys/devices/system/node/** r,  
/proc/** r,
```

- سپس در بخش «Allow data dir access» مسیر جدید را جایگزین مسیر قبلی کنید. بدین منظور با استفاده از کاراکتر # خط‌های قبلی را کامنت می‌کنیم و مسیر جدید را در ادامه‌ی خط‌های قدیمی اضافه می‌کنیم.

```
# Allow data dir access  
# /var/lib/mysql/ r,  
# /var/lib/mysql/** rwk,  
/data/mysql/ r,  
/data/mysql/** rwk,
```

- AppArmor را Reload می‌کنیم

```
# systemctl reload apparmor
```

- نهایتاً سرویس پایگاه داده را مجدداً راه‌اندازی می‌کنیم.

```
# service mysql start
```

برای اینکه از صحت تغییرات اعمال شده اطمینان حاصل کنیم می‌توانیم دستور زیر را یک بار در ترمینال اجرا کنیم. در صورتی که امکان ورود به خط فرمان پایگاه داده را داشتید، همه چیز درست انجام شده است:

```
# mysql -p -u redmine
```


نصب Ruby

برای نصب نسخه‌ی به روز Ruby از RVM استفاده می‌کنیم.

الف. ابتدا دستورات زیر را برای دریافت و نصب RVM اجرا می‌کنیم:

```
# apt-add-repository -y ppa:rael-gc/rvm
# apt-get update
# apt-get install rvm
```

ب. یک بار «logout» نمایید و بار دیگر با دسترسی Root به ترمینال «login» نمایید.

```
# logout
```

پ. دستورات زیر را برای نصب Ruby در خط فرمان اجرا می‌کنیم:

با توجه به ایرادات پیش آمده در اعتبار Certification های rvm لازم است دستور قدیمی `rvm get stable` با دستورات جدید زیر جایگزین شود. توجه داشته باشید در ابتدای خط اولین دستور کاراکتر «\» باید درج شود.

```
# command curl -sSL https://rvm.io/pkuczynski.asc | gpg --import -
# \curl -sSL https://get.rvm.io -k | bash -s stable
# apt-get install ruby
# rvm reload
# rvm requirements
# rvm install 2.5.0
# rvm use 2.5.0 --default
```

در صورتی که کار را درست انجام داده باشید با استفاده از دستور زیر باید بتوانید نسخه‌ی Ruby نصب شده را مشاهده نمایید که در خط بعدی می‌توانید آن را مشاهده نمایید.

```
# ruby -v
ruby 2.5.0p0 (2017-12-25 revision 61468) [x86_64-linux]
```

پس از نصب لازم است به ازای هر کاربر یک بار دستور زیر را انجام دهید و کاربران را در گروه rvm قرار دهید:

```
# usermod -a -G rvm username
```

به طور مثال:

```
# usermod -a -G rvm root
```

نصب Phusion Passenger

الف. برای نصب Phusion به صورت زیر عمل می‌کنیم:

در نسخه‌ی های اخیر تا تاریخ به‌روزرسانی این مستند Certification های نصب بسته‌های Passenger نیز غیر معتبر شده‌اند به این دلیل لازم است در گام اول تنظیماتی در مخازن قابل اعتماد Apt ایجاد کنیم. این بخش به ابتدای دستورات لازم افزوده شده است.

ابتدا یک پرونده در مسیر زیر ایجاد می‌کنیم:

```
/etc/apt/apt.conf.d/trusted-apt
```

و تنظیمات زیر را در آن جایگذاری می‌کنیم:

```
// Do not verify peer certificate
Acquire::https::Verify-Peer "false";
// Do not verify that certificate name matches server name
Acquire::https::Verify-Host "false";
```

سپس دستورات زیر را اجرا می‌کنیم:

```
# apt-get install -y dirmngr gnupg
# apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 561F9B9CAC40B2F7
# apt-get install -y apt-transport-https ca-certificates
```

ب. سپس لازم است اطلاعات دسترسی به مخزن سفارشی Phusion را به لیست مخازن Ubuntu اضافه کنید:

```
# sh -c 'echo deb https://oss-binaries.phusionpassenger.com/apt/passenger xenial main >
/etc/apt/sources.list.d/passenger.list'
```

پ. در نهایت پس از انجام یک به‌روزرسانی لیست بسته‌ها افزونه‌ی nginx و passenger را نصب می‌کنیم:

```
# apt-get update
# apt-get install -y nginx-extras passenger
```

پیکربندی Nginx

الف. پرونده‌ی پیکربندی Nginx را از مسیر زیر باز می‌کنیم:

```
/etc/nginx/nginx.conf
```

ب. سپس خط زیر را از حالت توضیح (کامنت) خارج می‌کنیم. برای این کار کافیهست عبارت # جلوی خط را پاک کنیم:

```
include /etc/nginx/passenger.conf
```

پ. فایل تغییر داده شده را ذخیره می‌کنیم و سپس یک بار سرورِس nginx را راه اندازی مجدد می‌کنیم:

```
# systemctl restart nginx
```

ت. برای اطمینان از صحت اعمال تنظیمات، دستورات زیر را انجام می‌دهیم:

```
# /usr/bin/passenger-config validate-install
```

همانطور که در شکل ۲ قابل مشاهده است Phusion passenger را انتخاب می‌کنیم:

```
root@redmine-deployment:~# vi /etc/nginx/nginx.conf
root@redmine-deployment:~# systemctl restart nginx
root@redmine-deployment:~# /usr/bin/passenger-config validate-install
What would you like to validate?
Use <space> to select.
If the menu doesn't display correctly, press '!'

☐ ☐ Passenger itself
☐ Apache
```

شکل ۲ – اعتبارسنجی Passenger

در صورت تایید عمل کرد، خروجی شبیه به شکل ۳ را در ترمینال مشاهده خواهید نمود:

```
root@redmine-deployment:~# /usr/bin/passenger-config validate-install
What would you like to validate?
Use <space> to select.
If the menu doesn't display correctly, press '!'

☐ ☐ Passenger itself
☐ Apache

-----

* Checking whether this Passenger install is in PATH... ✓
* Checking whether there are no other Passenger installations... ✓

Everything looks good. :-)
```

شکل ۳ – نتیجه‌ی اعتبارسنجی Phusion

سپس دستور زیر را برای اطمینان از بارگذاری Phusion در RAM اجرا می‌کنیم:

```
# /usr/sbin/passenger-memory-stats
```

در صورتی که Phusion به درستی در حافظه بارگذاری شده باشد خروجی مانند شکل ۴ را مشاهده خواهید نمود:

```

root@redmine-deployment:~# /usr/sbin/passenger-memory-stats
Version: 6.0.7
Date : 2021-03-02 04:32:39 -0500
----- Apache processes -----
*** WARNING: The Apache executable cannot be found.
Please set the APXS2 environment variable to your 'apxs2' executable's filename, or set the
or 'apache2' executable's filename.

----- Nginx processes -----
PID      PPID      VMSize    Private   Name
-----
29332     1         175.0 MB  0.7 MB    nginx: master process /usr/sbin/nginx -g daemon on; master
29335     29332    175.0 MB  0.7 MB    nginx: worker process
29336     29332    175.0 MB  0.7 MB    nginx: worker process
29339     29332    175.0 MB  0.7 MB    nginx: worker process
29340     29332    175.0 MB  0.7 MB    nginx: worker process
### Processes: 5
### Total private dirty RSS: 3.46 MB

----- Passenger processes -----
PID      VMSize     Private   Name
-----
29313     379.3 MB   2.1 MB    Passenger watchdog
29316     1273.5 MB  3.5 MB    Passenger core
### Processes: 2
### Total private dirty RSS: 5.64 MB

```

شکل ۴ - بررسی بارگذاری Phusion در RAM

ت. در این مرحله پیکربندی nginx را در مسیر مناسب رونوشت می‌کنیم:

```
# cp /etc/nginx/sites-available/default /etc/nginx/sites-available/default.orig
```

ث. فایل پیکربندی را از مسیر زیر باز می‌کنیم:

```
/etc/nginx/sites-available/default
```

در این فایل مسیر ریشه‌ی پرونده‌های Redmine را جایگذاری می‌کنیم و برخی تنظیمات دیگر را نیز اعمال می‌کنیم (شکل ۵):

```
# Default server configuration
#
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/html;
```

شکل ۵ - پرونده‌ی پیکربندی nginx

```
root /data/redmine/redmine/public/;
passenger_enabled on;
client_max_body_size 10m;
```

برای جلوگیری از پیغام‌های نامناسب 404 بخش زیر را (شکل ۶) به حالت توضیحات تغییر می‌دهیم (کامنت می‌کنیم):

```
server_name _;

#location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    #try_files $uri $uri/ =404;
#}
```

شکل ۶ - کامنت کردن بخش location

ساخت حساب Redmine

با استفاده از دستور زیر حساب کاربری Redmine را ایجاد می‌کنیم:

```
# adduser --system --shell /bin/bash --gecos 'Redmine Administrator' --group --disabled-password --
home /data/redmine redmine; sudo usermod -a -G rvm redmine
```

برای ادامه‌ی کار، به طور موقت دسترسی کامل Root را به این کاربر (کاربر Redmine) اعطا می‌کنیم. بدین منظور دستور زیر را اجرا می‌کنیم:

```
# visudo
```

به این ترتیب پرونده‌ی پیکربندی کاربران باز می‌شود. در این پرونده عبارت زیر را اضافه می‌کنیم.

```
redmine    ALL=(ALL)    NOPASSWD:ALL
```

پرونده را ذخیره می‌کنیم و می‌بندیم.

نصب بسته‌های پیش‌نیاز

با استفاده از دستورات زیر بسته‌های پیش‌نیاز را نصب می‌کنیم:

```
# apt-get install -y build-essential imagemagick libmagickwand-dev
```

نصب Redmine

دریافت فایل‌های نصب

در این مرحله پرونده‌های Redmine فارسی را از مخزن محلی Git دریافت و در مسیر متناسب قرار می‌دهیم. بدین منظور مراحل زیر را طی می‌کنیم:

الف. از کاربر Root به کاربر Redmine سوئیچ می‌کنیم:

```
# su - redmine
```

در صورتی که در ترمینال دستور pwd را اجرا نمایید می‌بایستی مسیر زیر نمایش داده شود:

```
$ redmine@redmine-deployment:~$ pwd  
/data/redmine
```

ب. فایل‌های نصب را از مخزن کد محلی دریافت می‌کنیم. به طور مثال دستور زیر را اجرا می‌کنیم:

```
$ git clone https://github.com/dearssickness/redmine-fa.git redmine
```

توجه داشته باشید که در نشانی دسترسی به مخزن Git (که در بالا با رنگ قرمز مشخص شده است)، مسیر محلی دسترسی به کد خود را جایگزین نمایید.

پیگر بندی دسترسی به MySQL

الف. در این بخش اطلاعات دسترسی به پایگاه داده را برای Redmine مشخص می‌کنیم:

نکته: کپی این پرونده در نسخه‌ی موجود در Github از پیش اعمال شده است.

```
$ cd redmine
$ cp -pR config/database.yml.example config/database.yml
```

ب. پرونده‌ی رو نوشت شده را باز می‌کنیم و اطلاعات دسترسی به پایگاه داده‌ها را به روز می‌کنیم. در صورتی که با استفاده از این مستند تا این قسمت پیش رفته باشید تنها کافیسست نام کاربری و گذرواژه‌ی دسترسی به پایگاه داده را به‌روز نمایید.

```
$ vim config/database.yml
```

در این پرونده بخش «production» را پیدا می‌کنیم و از زیر مجموعه‌ی آن مقادیر «username» و «password» را با مقادیر با ارزش جایگزین می‌کنیم (شکل ۷):

```
# Default setup is given for MySQL 5.7.7 or later.
# Examples for PostgreSQL, SQLite3 and SQL Server can be found at the end.
# Line indentation must be 2 spaces (no tabs).

production:
  adapter: mysql2
  database: redmine
  host: localhost
  username: redmine
  password: "a"
  # Use "utf8" instead of "utfmb4" for MySQL prior to 5.7.7
  encoding: utf8mb4

development:
  adapter: mysql2
  database: redmine_development
  host: localhost
  username: root
  password: ""
  # Use "utf8" instead of "utfmb4" for MySQL prior to 5.7.7
  encoding: utf8mb4

# Warning: The database defined as "test" will be erased and
# re-generated from your development database when you run "rake".
# Do not set this db to the same as development or production.
test:
  adapter: mysql2
  database: redmine_test
  host: localhost
  username: root
  password: ""
  # Use "utf8" instead of "utfmb4" for MySQL prior to 5.7.7
  encoding: utf8mb4
```

شکل ۷ – نمونه‌ی پیکربندی پایگاه داده

در مثال بالا نام کاربری «redmine» و گذرواژه «a» قرار داده شده است.

نصب Gem

در این بخش بسته‌های Gem پروژه را نصب می‌کنیم.

الف. با در نظر گرفتن مسیر جاری بخش قبلی

```
/data/redmine/redmine
```

دستورات زیر را انجام می‌دهیم:

```
$ gem install bundler -v '2.2.27'
$ bundle install --without development test
```

نکته: توجه داشته باشید نسخه‌ی درج شده در دستور بالا که به صورت **2.2.27** مشخص شده است، باید با نسخه‌ی درج شده در بخش «BUNDLED WITH» (آخرین خط) فایل Gemfile.lock یکی باشد.

پس از اجرای دستور بالا نصب کننده‌ی bundle برای دریافت بسته‌های مورد نیاز به شبکه‌ی اینترنت وصل خواهد شد. در صورت موفقیت پیغام متناظر مشابه پیغام زیر را مشاهده خواهید نمود:

```
Bundle complete! 35 Gemfile dependencies, 63 gems now installed.  
Gems in the groups development and test were not installed.  
Use `bundle info [gemname]` to see where a bundled gem is installed.
```

دستورات Rake

الف. پرونده‌ی زیر را باز می‌کنیم:

```
/data/redmine/redmine/config/application.rb
```

در خط اول این فایل وابستگی زیر را اضافه می‌کنیم و سپس فایل را ذخیره می‌کنیم و می‌بندیم (شکل ۸):

نکته: این عبارت در نسخه‌ی موجود در Github از پیش اعمال شده‌است.

```
require 'uri'
```

```
require 'uri'  
# frozen_string_literal: true  
  
require File.expand_path('../boot', __FILE__)  
  
require 'rails'  
# Pick the frameworks you want:  
require 'active_model/railtie'  
require 'active_job/railtie'  
require 'active_record/railtie'  
require 'active_storage/engine'  
require 'action_controller/railtie'  
require 'action_mailer/railtie'  
require 'action_view/railtie'  
require 'action_cable/engine'  
# require 'sprockets/railtie'  
require 'rails/test_unit/railtie'
```

شکل ۸ – افزودن وابستگی

نکته: لطفا توجه داشته باشید که عبارت بالا را در فایل مورد نظر تایپ کنید و از رونوشت کردن آن از روی فایل Word اجتناب نمایید.

ب. در پوشه‌ی جاری (/data/redmine/redmine) دستورات زیر را اجرا می‌کنیم:

```
$ bundle exec rake generate_secret_token
```

پ. در صورتی که پیغام خطایی مشاهده نکردید، دستورات زیر را برای ایجاد و پیکربندی جداول پایگاه داده اجرا نمایید:

```
$ RAILS_ENV=production bundle exec rake db:migrate
```

ت. سپس دستور زیر را اجرا نمایید:

```
$ RAILS_ENV=production bundle exec rake redmine:load_default_data
```

به این ترتیب از شما درخواست می‌شود برای بارگذاری تنظیمات پیش‌فرض، زبان مورد نظر را انتخاب نمایید (شکل ۹):

```
(see full list of languages by running 'rake redmine:load_default_data --help')
redmine@redmine-deployment:~/redmine$ RAILS_ENV=production bundle exec rake redmine:load_default_data

Select language: ar, az, bg, bs, ca, cs, da, de, el, en, en-GB, es, es-PA, et, eu, fa, fi, fr, gl, he, hr, hu, id, it, ja, jp, ko, lt, lv, mk, mn, nl, no, pl, pt-BR, pt-br, ro, ru, sk, sl, sq, sr, sr-YU, sv, th, tr, uk, vi, zh, zh-TW [en] fa
```

شکل ۹ – انتخاب زبان تنظیمات پیش‌فرض

به این ترتیب با نمایش شکل ۱۰ موفقیت بارگذاری تنظیمات پیش‌فرض اعلام می‌شود:

```
n] fa
=====
Default configuration data loaded.
```

شکل ۱۰ – موفقیت بارگذاری تنظیمات پیش‌فرض

نصب افزونه‌ها

در حال حاضر دو افزونه‌ی Redmine Agile و Redmine more previews فارسی‌سازی شده‌اند. برای نصب و استفاده از این افزونه‌ها لازم است گام‌های زیر را انجام دهیم.

نکته: پس از نصب و پیکربندی افزونه‌ها لازم است تنظیمات متناظر با هر یک از افزونه‌ها از بخش «راه‌بری» سامانه انجام شود. در صورت لزوم می‌بایستی اجازه‌های دسترسی به روز شوند و افزونه‌ها برای پروژه‌های مورد نظر فعال شوند.

افزونه‌ی چابک

ترمینال را با مسیر جاری زیر باز می‌کنیم:

```
/data/redmine/redmine
```

سپس دستور زیر را در آن اجرا می‌کنیم:

```
$ bundle exec rake redmine:plugins NAME=redmine_agile RAILS_ENV=production
```

افزونه‌ی More previews

برای نصب این افزونه نیازی به پیکربندی‌های پایگاه داده وجود ندارد و تنها لازم است بسته‌های پیش‌نیاز استفاده از آن از مخزن Ubuntu دریافت و نصب شود. بدین منظور دستور زیر را در ترمینال اجرا می‌کنیم:

```
# apt-get install libreoffice pandoc
```

نکته: دستور بالا نیاز به دسترسی Root برای نصب دارد.

ورود به سامانه

پس از انجام مراحل نصب با ورود نشانی IP ماشین، می‌توانید به سامانه ورود کنید. نکاتی که پس از بالا آمدن ردمین ممکن است مفید باشند را در ادامه مشاهده می‌نمایید:

- حساب کاربری و گذرواژه پیش‌فرض admin/admin می‌باشد.
- پس از ورود لازم است این گذرواژه را با عبارت ۸ کاراکتری جایگزین نمایید.
- با توجه به تغییرات فراوان ایجاد شده در سامانه (دسته گل‌های بنده) امکان استفاده‌ی کامل از زبان انگلیسی وجود ندارد. با اولین ورود از منوی بالا بخش «Administration» ← «settings» ← صفحه‌ی «Display» مقدار «Default language» را به فارسی تغییر دهید. سپس تیک مربوط به گزینه‌های
 - «Force default language for anonymous users» و
 - «Force default language for logged-in users» را روشن نمایید و گزینه‌ی «Save» را از پایین صفحه انتخاب نمایید.
- لطفاً قالب تاریخ و زمان را تغییر ندهید چون بعضی از کاراکترهای جدا کننده‌ی زمان D-: Hard code شده‌اند.