

# Training Guide

## I - Set up Ruby on Rails Development Environment

Ubuntu 12.04 LTS

RVM - Ruby version manager

Ruby 2.0.0

Rails 4.0.0

Git

Heroku Toolbelt

Sublime Text 2

### ***Open Terminal to setup for Sh***

First change some settings in Gnome Terminal. Go to *Edit -> Profile Preferences -> Title and Command* and check the “Run Command as login shell” box.

### **1. Install Ruby on Rails**

Make sure all packages are up to date

```
sudo apt-get update
```

Install curl (if not have)

```
sudo apt-get install curl
```

Install rvm

```
\curl -L https://get.rvm.io | bash -s stable  
source ~/.rvm/scripts/rvm
```

Install RVM dependencies

```
rvm requirements
```

Now, install Ruby 2.0.0

```
rvm install 2.0.0
```

Let RVM use Ruby 2.0.0 by default

```
rvm use 2.0.0 --default
```

It's time to install Rails 4.0.0 without ri and rdoc (which you actually never use)

```
gem install rails --version 4.0.0 --no-ri --no-rdoc
```

Running the following command to make sure Ruby on Rails has been installed successfully

```
rails new test
```

```
cd test
```

```
rails s
```

Open browser, visit: <http://localhost:3000>, if a web page is open then you are success.

Fix a problem that makes "rails not found" error:

<https://rvm.io/integration/gnome-terminal>

## 2. Install Git

Create an account on Github @ <https://github.com/>

Install Git

```
sudo apt-get install git-core
```

Git basic config

```
git config --global user.name "Your Name"
```

```
git config --global user.email newuser@example.com
```

Your email should be the same as your Github account. You may set it differently, but it will introduce some additional work.

To avoid authentication each time you pull from or push to Github, add these lines to .netrc file:

```
machine github.com
  login yourusername
  password yourpassword
```

## 3. Install Heroku Toolbelt

Create an account on Heroku @ <https://www.heroku.com/>

Install Heroku Toolbelt follow the instruction at <https://toolbelt.heroku.com/>

After successfully install Heroku Toolbelt, run the following command:

```
heroku login
```

## 4. Install Sublime Text 2

```
sudo add-apt-repository ppa:webupd8team/sublime-text-2
```

```
sudo apt-get update
```

```
sudo apt-get install sublime-text
```

You can launch Sublime Text 2 from Terminal by typing `subl`

## **II - Learn Ruby on Rails**

### **1. Ruby on Rails tutorials**

Ruby on Rails Tutorial: <http://ruby.railstutorial.org/>

Good References:

<http://guides.rubyonrails.org/>

<http://api.rubyonrails.org/>

<http://google.com>

### **2. Rails & RESTful**

Basic about RESTful Web services

<http://www.ibm.com/developerworks/vn/library/ws-restful/>

References

Representational State Transfer (REST):

[http://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm)

RESTful Web Services book

<http://shop.oreilly.com/product/9780596529260.do>

### **le3. Working with form**

## **III - Learn Git**

### **1.**

Pro Git book: <http://git-scm.com/book>

### **2. Framgia pull-req procedure (True way)**

1. Before working on a new task

```
git checkout develop
git pull framgia develop
```
2. Create a new branch for the task

```
git checkout -b task-name
```
3. When ready for commit

```
git add .
git commit -m "descriptive message about this commit"
```
4. Checkout to the develop branch to pull newest code

```
git checkout
git pull framgia develop
```

5. Return to the working branch and rebase  
`git checkout task-name`  
`git rebase develop`
6. If there are conflicts, go to conflict files and fix them, after finish, continue rebase  
`git add .`  
`git rebase --continue`
7. Push the commit to Github account  
`git push origin task-name`

## IV - MySQL

Learning MySQL: <http://student.agh.edu.pl/~gegotek/Learning%20MySQL%20V413HAV.pdf>

Introduction to SQL: <http://cs.lmu.edu/~ray/notes/introsql/>

No SQL: <http://cs.lmu.edu/~ray/notes/nosql/>

Install: <https://help.ubuntu.com/12.04/serverguide/mysql.html>

1. `sudo apt-get install mysql-server`

## V - Coding Convention - Best Practices

Basic rules that you should follow in first place

- 80 chars rule (except view files)
- do not use `()` as possible

Links to Framgia Coding standard:

Ruby

<https://github.com/framgia/coding-standards/blob/master/vn/ruby/standard.md>

Ruby on Rails

<https://github.com/framgia/coding-standards/blob/master/vn/rails/standard.md>

## VI - Others

Install Memcached

`sudo apt-get install memcached`

Install [ImageMagick](#)

`sudo apt-get install imagemagick`

Install [Redis](#)

<http://redis.io/download>

Devise gem

<http://devise.plataformatec.com.br/>

Simple example

[https://github.com/jayshepherd/devise\\_example](https://github.com/jayshepherd/devise_example)

Install Node.js

<https://github.com/joyent/node/wiki/Installing-Node.js-via-package-manager#ubuntu-mint>

How to Setup a Production Server for Rails 4

<http://robmcclarty.com/blog/how-to-setup-a-production-server-for-rails-4>

Install mysql2 gem

```
sudo apt-get install libmysqlclient-dev
```