➜ Install heroku using its toolbelt

<https://toolbelt.heroku.com>

➜ Login to heroku through CLI console

$ heroku login

➜ Create an app on Heroku

$ heroku create

➜ Edit database.yml and add

development:

adapter: postgresql

encoding: unicode

database: railsforcharity\_development

host: 127.0.0.1

user: pkhadloya

password:

port: 5432

min\_messages: warning

production:

adapter: postgresql

encoding: unicode

database: railsforcharity\_production

host: 127.0.0.1

user: root

password:

port: 5432

min\_messages: warning

➜ Back it up

$ cp config/database.yml config/database.yml.example

➜ Add pg gem for postgresql database

gem ‘pg’

➜ Remove sqlite gem

➜ Uncomment unicorn gem

gem ‘unicorn’

➜ Create a config/unicorn.rb file and paste the folllowing

worker\_processes 3 # amount of unicorn workers to spin up

timeout 30 # restarts workers that hang for 30 seconds

preload\_app true # Enabling this preloads an application before forking worker processes.

before\_fork do |server, worker|

if defined?(ActiveRecord::Base)

ActiveRecord::Base.connection.disconnect!

Rails.logger.info('Disconnected from ActiveRecord')

end

sleep 1

end

after\_fork do |server, worker|

if defined?(ActiveRecord::Base)

ActiveRecord::Base.establish\_connection

Rails.logger.info('Connected to ActiveRecord')

end

end

➜ Create a Procfile and paste the following

web: bundle exec unicorn -p $PORT -c ./config/unicorn.rb

➜ Put the below line in application.rb

config.assets.initialize\_on\_precompile = false

Ref: <http://www.simonecarletti.com/blog/2012/02/heroku-and-rails-3-2-assetprecompile-error/>

➜ Deploy

$ git push heroku master

➜ Migrate database

$ heroku run rake db:migrate

➜ Open the site in a browser

$ heroku open

➜ If it fails to load, then check logs and troubleshoot

$ heroku logs

➜ Rename your app from CLI

$ heroku apps:rename newname --app oldname

**LOGIN to HEROKU CONTROL PANEL**

➜ Add your domain

➜ Add Zerigo DNS add on and configure

**New Relic Addon**

➜ Add gem 'newrelic\_rpm' to your Gemfile under staging & production

➜ bundle install

➜ Login to heroku control panel and add newrelic addon

➜ Once added, setup automatic pinging to your website so that it does not idle

→ Browse to

Menu > Availability Monitoring (under Settings)

→ Click “Turn on Availability Monitoring”

→ Enter the url to ping as <yourwebsite.com>

→ Select 1 minute for the interval

**~~Airbrake~~****~~Addon~~**

~~➜ Login to heroku control panel and add airbrake addon or~~

~~➜ heroku addons:add airbrake # This adds the the developer plan. # If you’d like another plan, specify that instead.~~

~~➜ Add gem 'airbrake' to your Gemfile~~

~~➜ bundle install~~

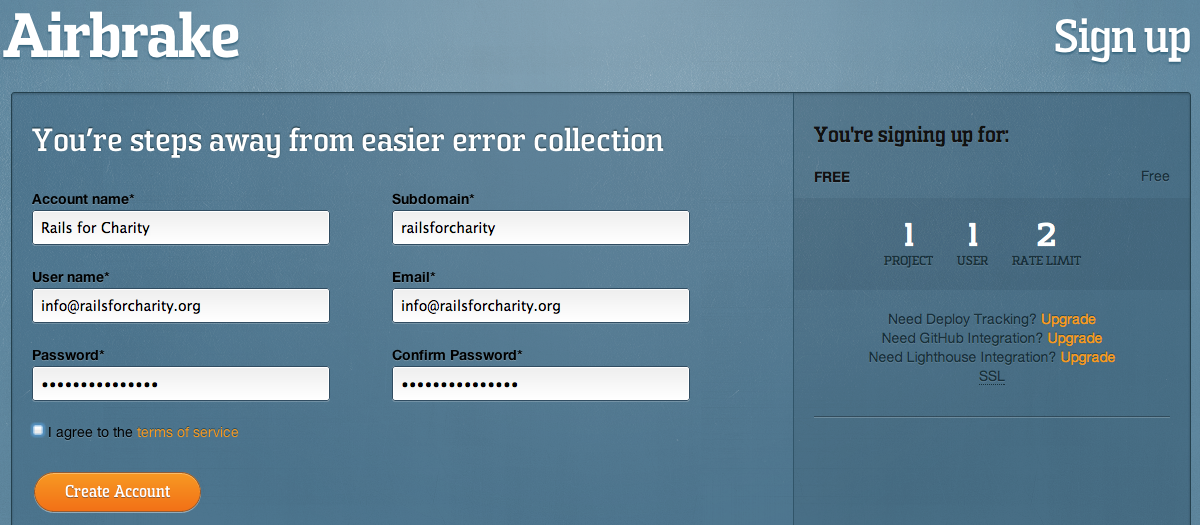
~~➜ rails g airbrake --heroku~~

~~➜ After adding the Airbrake add-on, you will need to install and configure the Airbrake notifier.~~

~~Your application connects to Airbrake with an API key. On Heroku, this is automatically provided to your application in ENV['AIRBRAKE\_API\_KEY'], so installation should be a snap!~~

**Airbrake**

1. Signup at airbrake.io for a free plan at <https://airbrake.io/account/new/Free>



2. Login and create a project

3. Put the airabrake gem in your Gemfile

4. Copy the key generation command from airbrake’s website and run

script/rails generate airbrake --api-key <your-api-key>

5. The generator will finish the installation and send a test error. Look for the exception to appear in your errors list on the airbrake website.

Also check https://github.com/errbit/errbit