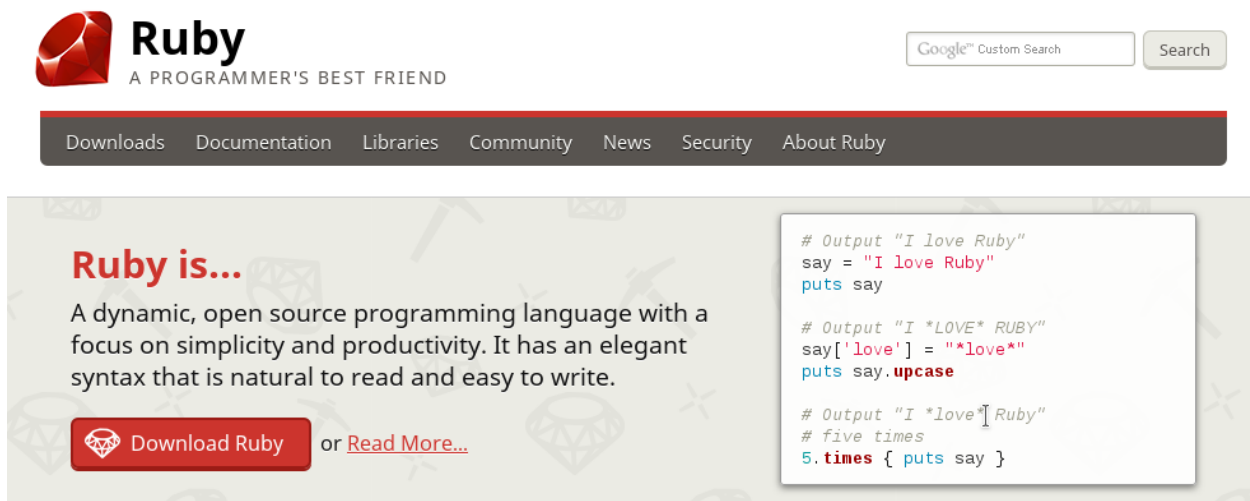# README

This documentation describes how to set up the application, its components, how to deploy it, and some internal information.

Set up

This application needs ruby version 2.3.1. The version is set at the top of the Gemfile, in a way similar to Heroku.

Most ruby version managers and Heroku recognize this syntax and will select the right version, or ask to install it. To find more about ruby, go here.

This application is also based on the latest stable Rails version, which is, at this moment, 5.0.1.

To find more about Rails, go [here](#).

First, you have to install all the dependencies. To do so, run:
```
gem install bundler
```



```
bundle install
```


After that, you have to set up the database. This can be done by running:
```
rails db:setup
```


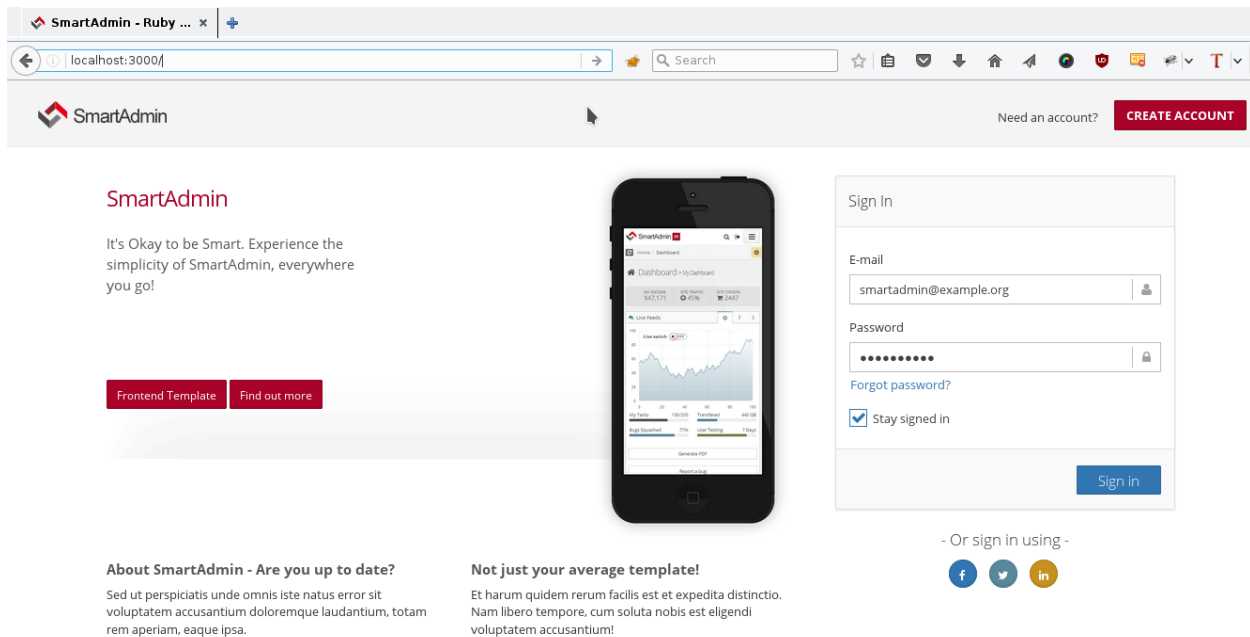After that, your application is ready to go. To run it, you have to execute:
```
rails s
```

```
[master]$ rails s
```

Point your browser to http://localhost:3000/ and enter these credentials:
```
E-mail: smartadmin@example.org
Password: smartadmin
```


Components

As it is said in the set up section, this application is based on ruby 2.3.1 and Rails 5.0.1. This application also uses the following gems:

Devise

This application is bundled with the devise gem for authentication. This gem has been included as it is one of the most popular choices for authentication. Devise's log in page has been re-skinned to match SmartAdmin own log in page, and to sirve as an example for devise users.
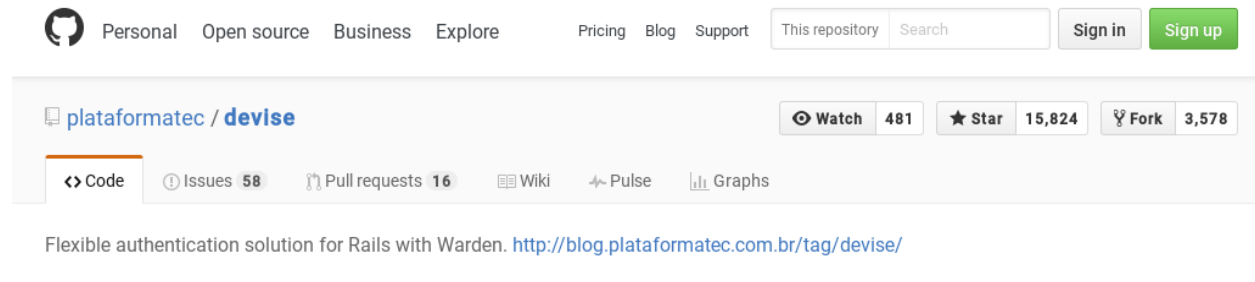
Gettext

This application is translated using gettext. To do so, it includes the gettext_i18n_rails gem. You can switch the language using the top-right selector, once you log in. This gem as been included as PO files are quite popular among translators.
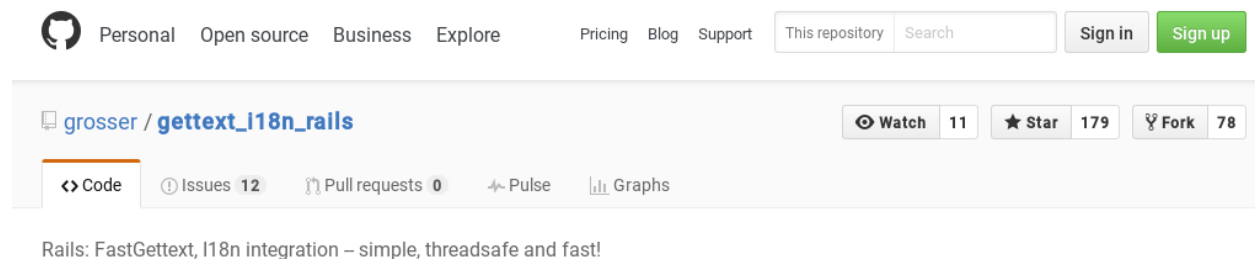
CKEditor

This application also includes a CKEditor integration. It uses the most recent version of CKEditor, which at this time is 4.5.11.
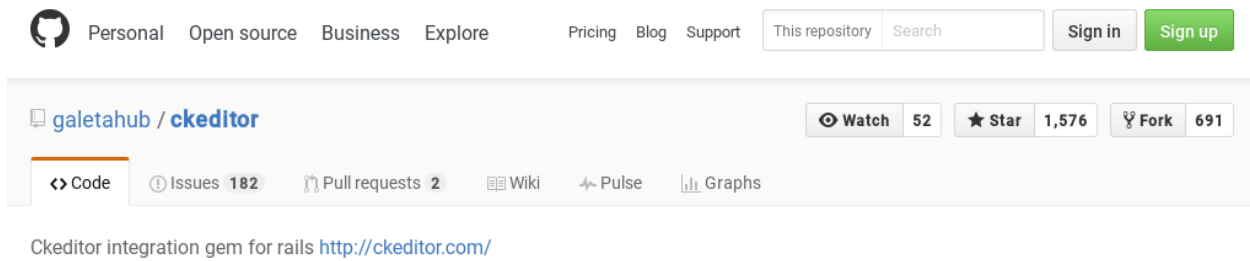
Deployment instructions

This application is Heroku-ready. To deploy it to heroku, you have to first set up an application
on Heroku, and add Heroku as a remote with this:
`heroku git:remote -a your-app-name`

```
[master]$ heroku git:remote -a your-app-name
```

After that, you only have to push it to Heroku:
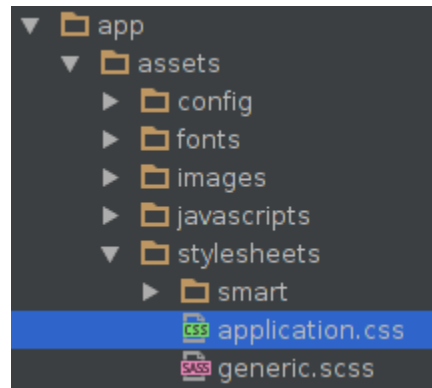`git push heroku master`

```
[master]$ git push heroku master
```

And set up the database
`heroku run rake db:setup`

```
[master]$ heroku run rake db:setup
```

Internal information

All the needed stylesheets are loaded, as standard, from
`assets/stylesheets/application.css`.

Almost all the javascripts are loaded, as standard, from
`assets/javascripts/application.js,` except for some that have to be loaded
directly from the template, as they require some special attributes.
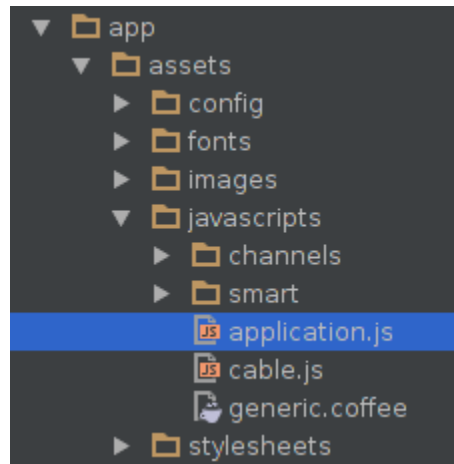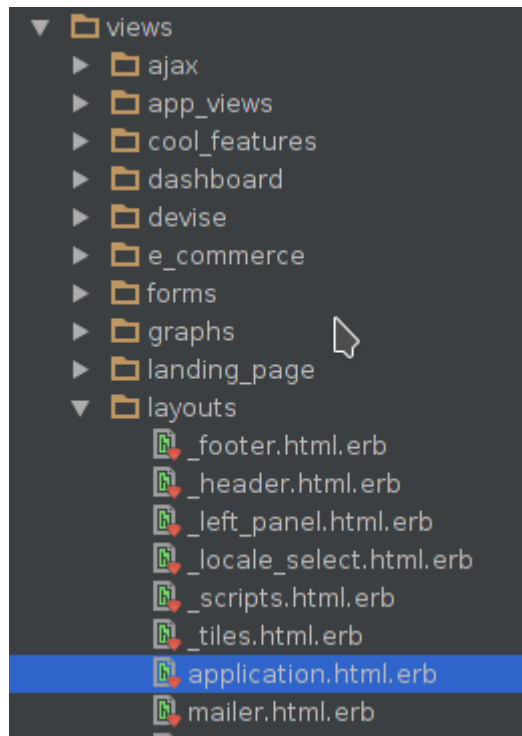

Each page can accept a `content_for :head` block

which you can use to insert styles and other types of content into the header HTML section. Also,
each page can accept a `content_for :scripts` block where you can set page specific
javascript.

If none is set, the contents of `app/views/layouts/_scripts.html.erb` are used as
default.

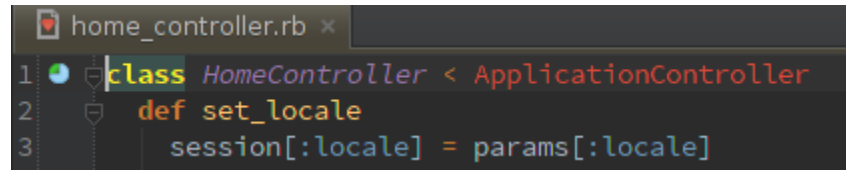Left side menu is dynamically generated. You can configure it by going to
`ApplicationHelper#left_menu_content`

The list of locales shown in the top-right selector can be also be configured by going to
`ApplicationHelper#locale_list`.

The locale is set as a session, variable, named locale, which is set in
`HomeController#set_locale`.

```ruby
class HomeController < ApplicationController
  def set_locale
    session[:locale] = params[:locale]
```