

# README

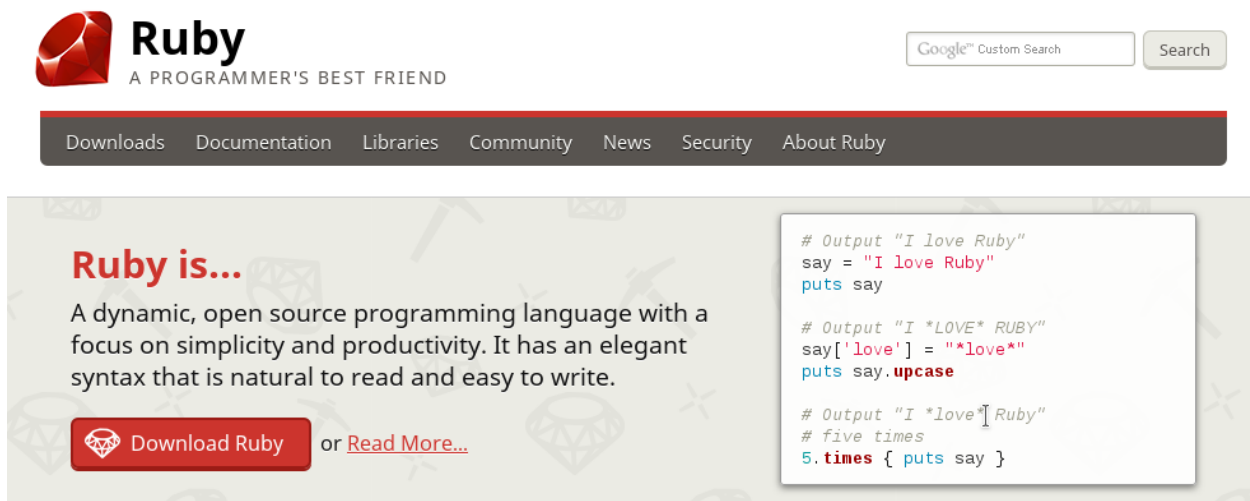
This documentation describes how to set up the application, its components, how to deploy it, and some internal information.

## 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.

```
Gemfile x
1  # frozen_string_literal: true
2  ruby '2.3.1'
3  source 'https://rubygems.org'
4
```

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](#).




**Ruby**  
A PROGRAMMER'S BEST FRIEND

Google™ Custom Search Search

Downloads Documentation Libraries Community News Security About Ruby

**Ruby is...**  
A dynamic, open source programming language with a focus on simplicity and productivity. It has an elegant syntax that is natural to read and easy to write.

 Download Ruby or [Read More...](#)

```
# Output "I love Ruby"
say = "I love Ruby"
puts say

# Output "I *LOVE* RUBY"
say['love'] = "*love*"
puts say.upcase

# Output "I *love* Ruby"
# five times
5.times { puts say }
```

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

```
Gemfile x
1  # frozen_string_literal: true
2  ruby '2.3.1'
3  source 'https://rubygems.org'
4
5  # Bundle edge Rails instead: gem 'rails', github: 'rails/rails'
6  gem 'rails', '~> 5.0.0', '>= 5.0.0.1'
```

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



[Blog](#) [Guides](#) [API](#) [Ask for help](#) [Contribute on GitHub](#)

## Imagine what you could build if you learned Ruby on Rails...

Learning to build a modern web application is daunting. Ruby on Rails makes it much easier and more fun. It includes **everything you need** to build fantastic applications, and **you can learn it** with the support of **our large, friendly community**.



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

```
gem install bundler
```

```
[master]$ gem install bundler
```

```
bundle install
```

```
[master]$ bundle install
```

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

```
rails db:setup
```

```
[master]$ rails db:setup
```

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

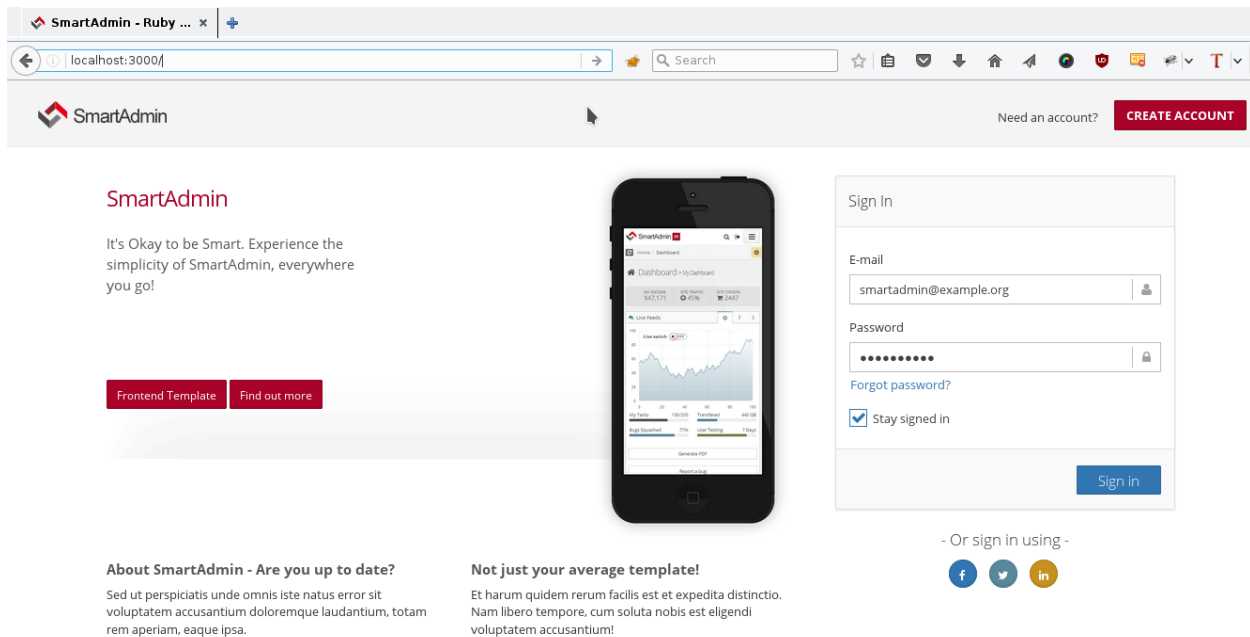
```
rails s
```

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

Point your browser to <http://localhost:3000/> and enter these credentials:

E-mail: smartadmin@example.org

Password: smartadmin



SmartAdmin

It's Okay to be Smart. Experience the simplicity of SmartAdmin, everywhere you go!

Frontend Template Find out more

Sign In

E-mail  
smartadmin@example.org

Password  
••••••••

[Forgot password?](#)

☒ Stay signed in

Sign in

- Or sign in using -

f t in

About SmartAdmin - Are you up to date?  
Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa.

Not just your average template!  
Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi voluptatem accusantium!

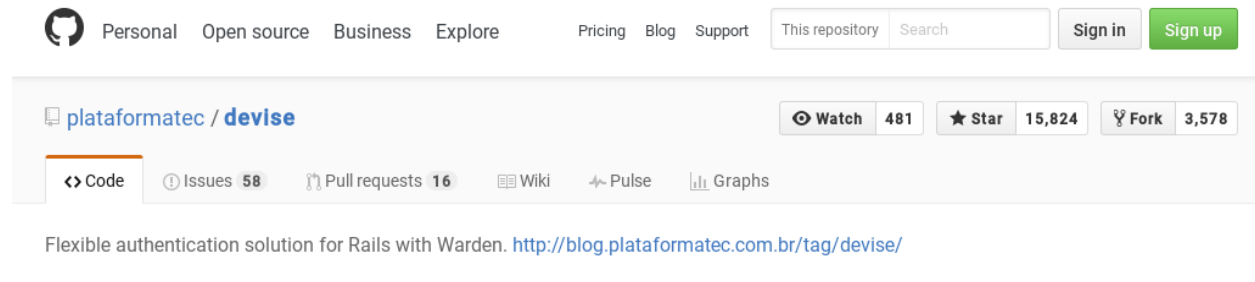
## Components

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:

```
Gemfile
1  ruby '2.3.1'
2  source 'https://rubygems.org'
3
4  # Bundle edge Rails instead: gem 'rails', github: 'rails/rails'
5  gem 'rails', '~> 5.0.0', '>= 5.0.0.1'
6
7  gem 'ckeditor'
8  gem 'devise'
9  gem 'gettext'
10
```

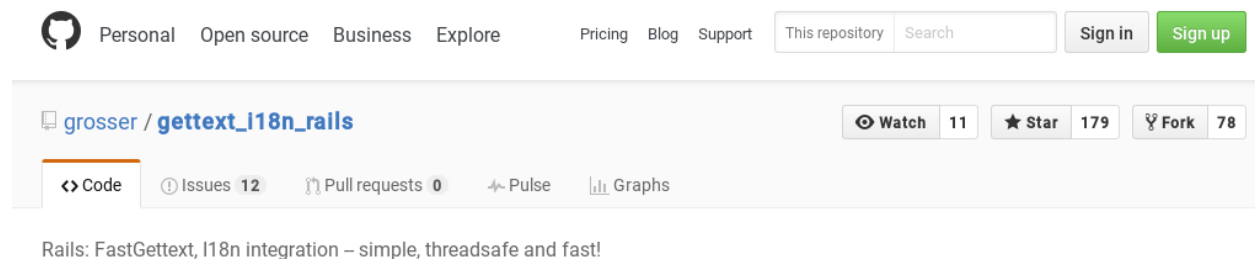
## Devise

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 serve as an example for devise users.



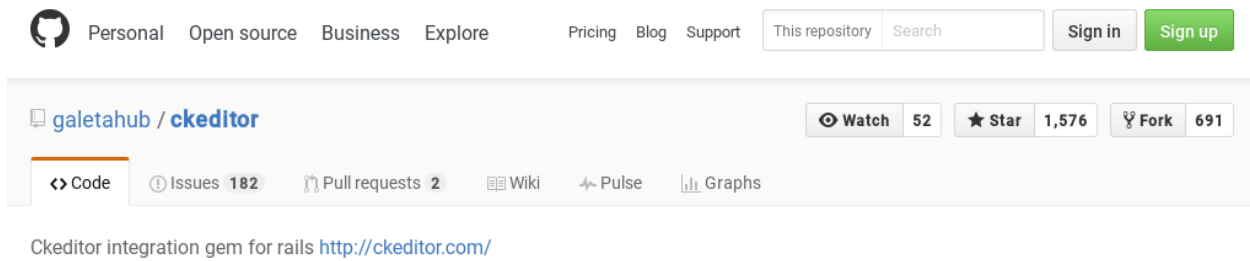
## Gettext

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
```

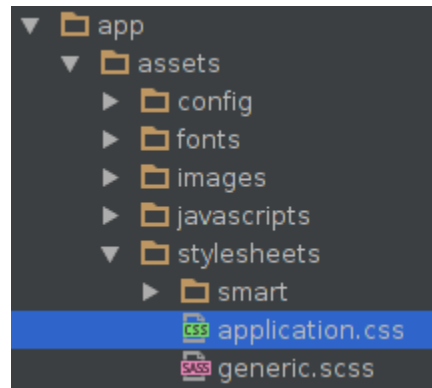
And set up the database

```
heroku run rake db:setup
```

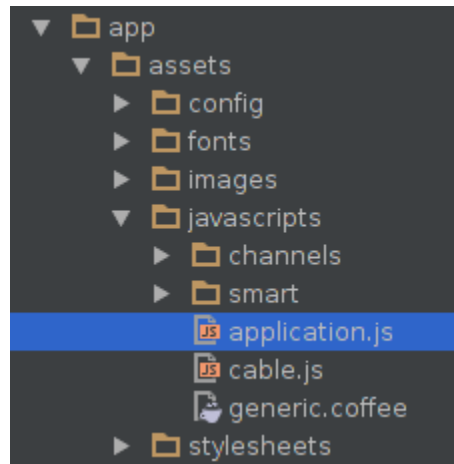
```
[master]$ heroku run rake db:setup
```

## 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.



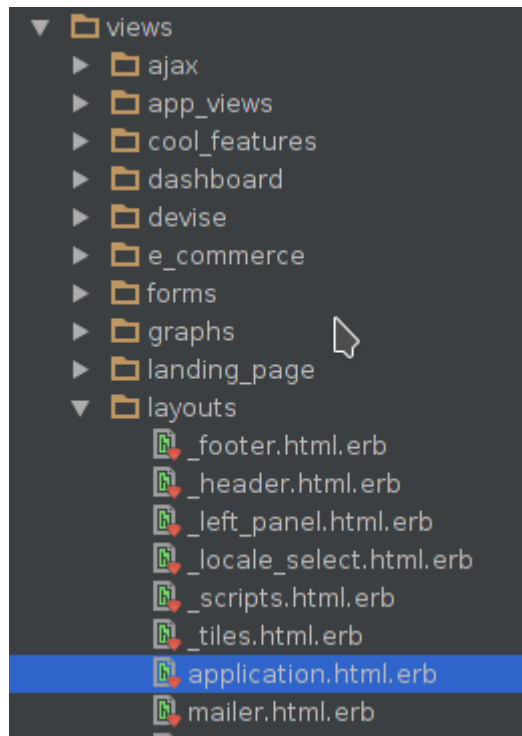
Each page can accept a `content_for :head` block

```
1 | <% content_for :head do %>
```

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.

```
141 | <% content_for :scripts do %>
```

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



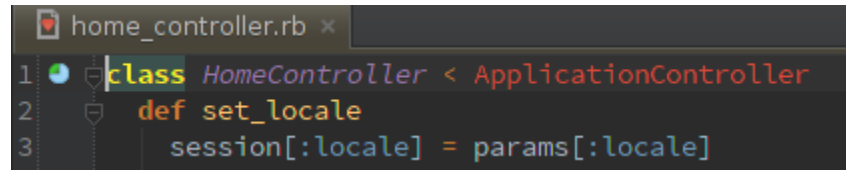
Left side menu is dynamically generated. You can configure it by going to `ApplicationHelper#left_menu_content`

```
application_helper.rb x
124     output.html_safe
125   end
126
127   def left_menu_content
128     [
129     {
```

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

```
application_helper.rb x
13
14   def locale_list
15     [
16     {
```

The locale is set as a session, variable, named locale, which is set in HomeController#set\_locale.

A screenshot of a code editor window titled 'home\_controller.rb'. The code defines a class 'HomeController' that inherits from 'ApplicationController'. Inside the class, there is a method 'set\_locale' which sets the session variable 'locale' to the value of 'params[:locale]'.

```
1 class HomeController < ApplicationController
2   def set_locale
3     session[:locale] = params[:locale]
```