[Lesson 8.3: User Roles, OmniAuth, and ActiveAdmin](#h.skne8etstdwz)

[Gem of the Day](#h.tjqswavd9eee)

[Adding an Admin Role to Doggy Daycare](#h.brkjt98m3l7x)

[Add Admin to User Model](#h.fbb2y17th7cv)

[Set Yourself as Admin](#h.avpodutdd0hg)

[Restricting Methods to Admins](#h.njhp6g1kn3uk)

[OmniAuth for Facebook (Optional)](#h.qr4fepoon12p)

[ActiveAdmin (Optional)](#h.wbvw9j1ngbj2)

[Lesson 8.4: Mailers](#h.bjnz9kyrh5hq)

[Gem of the Day](#h.dtqebew68fyi)

# Lesson 8.3: User Roles, OmniAuth, and ActiveAdmin

## Gem of the Day

CanCanCan <https://github.com/CanCanCommunity/cancancan>

OmniAuth <https://github.com/intridea/omniauth>

## Adding an Admin Role to Doggy Daycare

You can do more complex user management with the CanCanCan gem, but today we are just going to implement a basic admin role so that we can better understand how that gem makes a more complex arrangement easier to manage.

### Add Admin to User Model

Basically, we want to only allow admins to add/edit/destroy products, dogs, and owners.

1. First, we need to add an admin column to the user table:  
   rails g migration add\_admin\_to\_user admin:boolean
2. Edit your migration to make admin default to false:  
   add\_column :users, :admin, :boolean, :default => false
3. Now run rake db:migrate.
4. Finally, commit your changes.

### Set Yourself as Admin

How would we set the admin status of a user? We can manually go into rails console. What about in production? We can either use Figaro to set an environment variable and seed our user database with ourselves, OR we can manually go into rails console on Heroku with

heroku run rails console

So how do we do that in console?

1. ap User.all (awesome print if you have it, then see which user you are)
2. Assuming you’re the first user, user=User.first
3. user.admin = true
4. user.save
5. ap User.first (to double-check)

### Restricting Methods to Admins

Since we added a column with the property admin set to a boolean, we can call the following method to determine whether a user is an admin or not:

current\_user.admin?

We can also create a method similar to the one Devise uses for authenticate\_user.

First, in our **Dogs controller**, let’s add an authenticate admin user method that we will write in the next step:

class DogsController < ApplicationController

before\_action :set\_dog, only: [:show, :edit, :update, :destroy]

before\_action :authenticate\_user!, except: [:index, :show]

before\_action :authenticate\_admin\_user!, except: [:index, :show]

Now, let’s create that action in the **Application Controller** as well as what to do if it errors out.

class ApplicationController < ActionController::Base

# Prevent CSRF attacks by raising an exception.

# For APIs, you may want to use :null\_session instead.

protect\_from\_forgery with: :exception

# restrict access to admin module for non-admin users

def authenticate\_admin\_user!

raise SecurityError unless current\_user.try(:admin?)

end

rescue\_from SecurityError do |exception|

redirect\_to root\_url, alert: 'You do not have access to that feature.'

end

end

Finally, let’s add alerts to our application.html.erb and customize the format so it’s red:

<% if notice.present? %>

<div class="alert alert-dismissable alert-success">

<button type="button" class="close" data-dismiss="alert" aria-hidden="true">&times;</button>

<%= notice %>

</div>

<% elsif alert.present? %>

<div class="alert alert-dismissable alert-danger">

<button type="button" class="close" data-dismiss="alert" aria-hidden="true">&times;</button>

<%= alert %>

</div>

<% end %>

Now test it out! Commit your changes.

Classroom challenge: Add admin authentication to Owners (optional), Products, Categories, and Orders, though be careful which methods you add it to - you may kill some of your current functionality.

## OmniAuth for Facebook (Optional)

See if you can follow along with this lesson to add sign-up/login ability with Facebook:

<http://techtalentsouth.slides.com/techtalentsouth/atl-facebook-omniauth-186?token=qG3hLhh3>

## ActiveAdmin (Optional)

See if you can follow along with the second part of this method to add ActiveAdmin to your project - you will need to think through this and change things that do not make sense for your app:

<https://docs.google.com/document/d/1t1BODGNSiGHMQ5B41ZKCM--59BUXJ1IjNmA398ZdViM/edit?usp=sharing>

# Lesson 8.4: Mailers

## Gem of the Day

Letter Opener <https://github.com/ryanb/letter_opener>

**Vote**

Confirmable and Letter opener or OmniAuth (facebook)?

<https://github.com/plataformatec/devise/wiki/How-To:-Add-:confirmable-to-Users>