Quick notes for today’s class

We installed two gems today, devise and activeadmin. Devise is used for user authentication (i.e. signing in and signing out). ActiveAdmin is used to create an admin portal.

Steps for adding devise to our musicdb

1. Make sure you commit everything and merge into master.
2. create a new branch off of master
   1. git checkout -b devise
3. search for gem devise, and read the github documentation.
4. The documentation tells us:
   1. put into gemfile:
      1. gem ‘devise’
   2. run bundle install
   3. run the command:
      1. rails generate devise User
      2. this will create a new user resource much like a scaffold would, but there are many more files associated with this resource
   4. Go to config/environments/development.rb and include this line:
      1. config.action\_mailer.default\_url\_options = { host: 'localhost', port: 3000 }
   5. make sure you have a root route
   6. run rake db:migrate
   7. visit /users/sign\_up
   8. you should now be able to sign up for an account!
   9. Now the problem is we can sign out… so… let’s add this in our application.html.erb, above the yield
      1. <strong><p class="sign\_up">
      2. <% if user\_signed\_in? %>
      3. Logged in as <strong><%= current\_user.email %></strong>.
      4. <%= link\_to 'Edit profile', edit\_user\_registration\_path, :class => 'navbar-link' %> |
      5. <%= link\_to "Logout", destroy\_user\_session\_path, method: :delete, :class => 'navbar-link' %>
      6. <% else %>
      7. <%= link\_to "Sign Up", new\_user\_registration\_path, :class => 'navbar-link' %> |
      8. <%= link\_to "Login", new\_user\_session\_path, :class => 'navbar-link' %>
      9. <% end %>
      10. </p></strong>
   10. As of right now, even if you are not signed in, you can still see everything… let’s change this by adding this helper method into each of our controllers (except for application\_controller, and static\_pages\_controller)
       1. before\_action :authenticate\_user!
       2. this will prevent anyone from using the important parts of our application!
   11. If this is working nicely for you make sure to commit this
   12. then switch branches back to master
       1. git checkout master
   13. then merge in this devise branch
       1. git merge devise

Now lets add in active admin!

1. search for the gem activeadmin, and read it’s github documentation
2. The documentation tells us to install the gem in our gemfile
   1. gem 'activeadmin', github: 'activeadmin'
   2. bundle install
3. run this command in terminal
   1. rails g active\_admin:install
4. rake db:migrate
5. check it out in the server
   1. go to /admin/login
   2. use the default email: [admin@example.com](mailto:admin@example.com)
   3. use the default password: password
6. if this looks terrible, with no styling you will need to generate assets for activeadmin, if it looks great then you do not need to do this
   1. rails g active\_admin:assets
   2. we will then have to physically move the css file that was just created, from our stylesheets directory into vender/assets/stylesheets directory
      1. this is so it does not interfere with other stylings
7. We will also have to create a file in the config/initializers folder
   1. create a file called kaminari.rb and insert this code
   2. Kaminari.configure do |config|  
       config.page\_method\_name = :per\_page\_kaminari  
      end
   3. kill your server and restart it
8. last but not least we need to create models for our resources
   1. so run these commands
   2. rails generate active\_admin:resource User
   3. rails generate active\_admin:resource Genre
   4. rails generate active\_admin:resource Artist
   5. rails generate active\_admin:resource Album
   6. rails generate active\_admin:resource Song
9. Now for each of these models we have to permit our attributes to be saved to the database via the browser, so for each model include this code. We also have to make it work nicely with friendly\_id.
   1. app/admin/User.rb
      1. permit\_params :email
   2. app/admin/Genre.rb
      1. permit\_params :name
      2. controller do
      3. defaults finder: :find\_by\_slug
      4. end
   3. app/admin/Artist.rb
      1. permit\_params :name, :bio, :genre\_id
      2. controller do
      3. defaults finder: :find\_by\_slug
      4. end
   4. app/admin/Album.rb
      1. permit\_params :title, :cover, :genre\_id, :artist\_id
      2. controller do
      3. defaults finder: :find\_by\_slug
      4. end
   5. app/admin/song.rb
      1. permit\_params :name, :genre\_id, :artist\_id, :album\_id
      2. controller do
      3. defaults finder: :find\_by\_slug
      4. end
10. You should now be good to go!