[Lesson 8.2: Collaboration Using GitHub](#h.u9vfvj4s3efd)

[Gem of the Day](#h.pwty1tkukjvb)

[Rails Composer Walk-Through](#h.sp1e4irjvnkr)

[Recipe Project](#h.daoa4txmjy02)

[Roles and Plan](#h.qj2ya8gymnzj)

[Action](#h.dmtblcgbw7st)

[Homework](#h.up1100wpuwwj)

[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 8.2: Collaboration Using GitHub

## Gem of the Day

ActiveAdmin <https://github.com/activeadmin/activeadmin>

<http://activeadmin.info/>

### Rails Composer Walk-Through

Rails Composer is a great tool that will build a lot of your basic app requirements all at once. Let’s walk through creating a fake app so that you understand the different options.

From the readme, we know to run this command (changing the app name to what you want your app to be named):

rails new myapp -m https://raw.github.com/RailsApps/rails-composer/master/composer.rb

Windows users will probably have an SSL error - here is the fix:

<https://gist.github.com/fnichol/867550>

Here are the options I generally choose for an app like this - feel free to experiment. However, in the beginning, I recommend you use less of the functionality until you get comfortable working with each of these tools.

* Build a starter application? 3) Custom application (experimental)
* Web server for development? 4) Puma
* Web server for production? 1) Same as development
* Database used in development? 1) SQLite
* Template engine? 1) ERB
* Test framework? 1) None (use default minitest)
* Front-end framework? 2) Bootstrap 3.3
* Add support for sending email? 1) None
* Authentication? 2) Devise
* Devise modules? 1) Devise with default modules
* Authorization? 2) Simple role-based
* Admin interface for database? 1) None (add ActiveAdmin later)
* Use a form builder gem? 2) SimpleForm
* Prepare for deployment 1) none or 2) Heroku
* Add gem and file for environment variables? 3) Add application.yml with Figaro

Notice that in both secrets.yml and application.yml, you have some default roles set up. These provide the admin login details you would need to log in as an admin - your database is currently seeded with this fake user (eventually you want to change this otherwise anyone can guess).

Go ahead and run your server and take a look at the app.

## Recipe Project

We are going to collaborate on a recipes project, which can be found at: <https://github.com/siakaramalegos/recipes-collaboration>

### Roles and Plan

Currently, it only has a basic scaffold of Recipes with title, ingredients, and steps. We want to make it fancier though, so here are some different features we can add - everyone should pick an item from this list or think of a new idea to add:

* add users using Devise, and add a user reference to the recipes table
* add styling (maybe using starter generators) and navbar
* add home and about pages with content, change root to home page
* modify index page to use panels
* convert to ajax (at least for the delete action on index)
* add seeds and maybe a populator task
* add pagination and/or friendly IDs

Once everyone has an item, think through whether anything depends on anything else, and if so, plan the steps accordingly.

### Action

Now that we have a plan, let’s start working! These are the general steps to contributing to an open-source project on GitHub. You can officially add collaborators, but we will do this project using pull requests.

1. Go to <https://github.com/siakaramalegos/recipes-collaboration> and click on “Fork”. This will make a copy of the repo in your own account, but it will still be linked to the original author.
2. Once the repo is copied into your account, make sure you are inside the version in your account, and copy the clone URL. Then go to your command line in your TTS folder and type “git clone URL”, but of course replace URL with that actual url.
3. Now, cd into that new app folder.
4. The first thing we will need to do is generally check out the gem file and modify. I’ve already conveniently includes gems for both Windows and Mac, so you should only need to bundle next. Before doing, this, verify which version of Rails you are running (especially for the Windows users).
5. Before we can take a peek at our app, we also need to do a rake db:migrate. This is because the development database is always local to a system and ignored by git.
6. Now run your rails server. If you get a permission denied, try doing bundle exec rails s instead.
7. Verify that the basic app is working as expected.

Now that we successfully pulled down a copy to our systems, we can start working on our assigned features. The generally accepted steps or norms are as follows:

1. Create a feature branch - use a branch name that is descriptive of the change you are going to make, like “starter-generators”. Type git checkout -b newbranchname to both create the branch and check it out.
2. Make all your changes in that branch. Use good git commit practices - one commit per logical change, and type good commit messages.  
     
   If applicable, also add tests. In real life, developers will expect you to write tests for whatever features you are adding. They will also expect that all tests will be passing before you do a pull request.
3. Once you are satisfied with your new feature, push that feature branch up to GitHub using git push origin newbranchname.
4. Navigate to your repo on GitHub, and switch to your feature branch. You should see a link that automatically pops up to do a Pull Request. Click on that. Make sure that your pull request selects my repo as the base fork, and then your repo and feature branch as the head fork. Submit the pull request with a descriptive comment of the changes you are making. Be nice!

Now, you wait until the base fork owner has a chance to review your changes and either accept or deny them. Conversations happen in this space, and some requests are rejected because you need to add something else. Don’t be bummed if this happens! It’s all a part of the process. Just fix your code and re-submit the pull request.

## Homework

Work on your personal apps!