CIS-196 - Final Project

Milestone 1 Due: Sunday, April 19th
Milestone 2 Due: Wednesday, April 29th, 11:59pm (last day of class)
Demo Due: Sunday, May 10th

I. Requirements

Your final project needs to demonstrate sufficient technical complexity. Here are a few requirements:

Model

1. Your models should have (at least one) `has_many` to `has_many` relationship.
2. Your project must have (at least one) `has_many` to `belongs_to` relationship, and the two models involved must have a nested resources structure (in routes)
3. You must query a third-party API to present external data relevant to (some of) your models.

Controller

1. There should be an idea of a "user", so account/session management (logging in + password management) should be implemented.
2. You should have a file uploading feature implemented using ActiveStorage.
3. You must use a third-party gem to add some additional features to your app.

View

1. You should use Bootstrap or other CSS library gems to stylize your project.
2. The project must be aesthetically "ready-for-production".

Deployment

1. Your project should be deployed to Heroku/AWS/DigitalOcean.
2. You should maintain a *private* GitHub/Gitlab repository.

Please stray away from making a personal website as they often do not have the complexity we are looking for. We also implemented a personal blog earlier in the semester and we want to avoid repetition.

Additionally, I would strongly discourage building on top of HW3.

II. Tips

It's hard to provide tips since everyone is doing something completely different, but here are some general things that should help:

- Start early and keep up with the milestones.
- **Set up Git for your project ASAP, commit early and often!!**
    - If you make a change that breaks something that was previously working, you'll be able to go back to a previous commit and not lose a lot of code.
- Research for ruby gems that can help you with your project!
    Take a look at [the awesome ruby repo](#) to see a curated list of useful gems.
- It'll be more difficult to help on Piazza, so take advantage of Office Hours.
    - When you post on Piazza, please be specific about how to reproduce your bug.
- Use [the pry gem](#) to debug your controllers!
- Use `rails c` command to debug your models!
    - Because if you come to us for debugging, this is exactly what we would do.

III. Deadlines

Milestone 1

For this milestone, you need to submit a 1-2 page "Project Proposal" by **Sunday, April 19th** according to the Project Proposal Template

- summarize your idea
- include a database relation diagram like that of HW3's in your project proposal
- Discuss step by step how you are going to implement this project

Milestone 2

1. Start by creating a new Rails project.
2. Create a private GitLab repository on your personal account.
3. Follow the instructions on the GitLab repo to make your first commit and push it up to GitLab.
    - Come to Office Hours or consult Google for help on this.
4. **Post privately to your GitLab repository.**
5. Think about the database tables in your project proposal and determine which generator makes the most sense for each one.
6. Generate all of the tables with the appropriate generator (this would be a great place to make another commit).
7. Set up associations and add validations to the models.
8. Test your models in the Rails Console (or write your own tests) and commit again when everything is working as expected.
    - Note that you are not restricted to the database that we discussed in your proposal. You can make additional changes as you develop the application, but this should give you a good place to start.
9. Spend some time thinking about all of the routes and views that you will need for your application. Especially consider these in relation to the routes that you've scaffolded already.
10. Start implementing the controllers, routes, and views in a flow that makes sense to you.
    - Regardless of how you go about this, you should **commit early and often**.

By the end of milestone 2 you should have a working prototype of your project, although the view files of the project may not be very sophisticated.

Project Demos

Project demos will take place during the final days. You can schedule a time to meet with Zhilei before **May 10th** via YouCanBookMe that he will send out later.

During the demo time, you will walk Zhilei through all of the features of your project and he will ask questions. Your demo must be done on your Heroku site (not locally), but otherwise the flow of your demo is completely up to you!

V. Grading

Grading Breakdown:

- 40% - Implementation
- 30% - Scope/Difficulty
- 20% - Technical Demo
- 10% - Best Practices & Style

Extra Credits

1. Implement a self-referring many-to-many association.
    - Most intuitively it could be "users have many users"
    - Twitter-like followers/followings
    - Facebook-like friendships
2. Social impact: build a project that may be useful for Philadelphia / the world's fight against the COVID-19 pandemic!