waspyfaeleith / e21_projects.md Secret

Last active 2 days ago • Report gist

Project Briefs (and other information)

e21_projects.md

Project - Web Programming with Ruby

Over the next week you will build an app to consolidate and expand on everything you have learned:

- Object oriented programming with Ruby
- Test Driven Development
- Web Programming (REST, MVC)
- Interacting with a PostgreSQL database (CRUD)

Rules

The project must be built using only:

- HTML / CSS
- Ruby
- Sinatra
- PostgreSQL and the PG gem

It must **NOT** use:

- Any Object Relational Mapper (e.g. ActiveRecord)
- JavaScript. At all. Don't even think about it.
- Any pre-built CSS libraries, such as Bootstrap.
- Authentication. Assume that the user already has secure access to the app.

All of these will be covered later in the course. Do not make any attempt to use them this week, even if you've used them before.

Make lots of little (but sensible!) git commits. The number of commits in your final submission is one of the criteria your work will be judged on.

Briefs

These briefs detail what is expected of you, including an MVP. Note that the MVP is the **Minimum** Viable Product. You are strongly encouraged to go beyond these specifications, and some possible extensions are provided.

Talk through your ideas with an instructor during the planning process, and before you begin coding.

Animal Shelter

The Scottish Animal Shelter accepts orphaned or stray animals and takes care of them until they can be adopted by a new owner. The shelter has a list of potential new owners for the animals. Animals may take a while to be trained up and made healthy before being available for adoption.

They are looking for a management system to keep track of their animals and owners.

MVP:

- A list of all their animals and their admission date
- Mark an animal as being adoptable/not adoptable
- Assign an animal to a new owner

- List all the owners and their adopted animals

Possible Extensions:

- CRUD actions for animals/owners
- Have separate pages for animals ready for adoption and ones still in training/vet care
- Search for animals by breed/type
- Any other ideas you might come up with

Shop Inventory

Build an app which allows a shopkeeper to track their shop's inventory. This is not an app which the customer will see, it is an admin/management app for the shop workers.

MVP

- The inventory should track individual products, including a name, description, stock quantity, buying cost, and selling price.
- The inventory should track manufacturers, including a name and any other appropriate details.
- The shop can sell anything you like, but you should be able to create, edit and delete manufacturers and products separately.
    - This might mean that it makes more sense for a car shop to track makes and models of cars. Or a bookstore might sell books by author, or by publisher, and not by manufacturer. You are free to name classes and tables as appropriate to your project.
- Show an inventory page, listing all the details for all the products in stock in a single view.
- As well as showing stock quantity as a number, the app should visually highlight "low stock" and "out of stock" items to the user.

Inspired by

eBay, Amazon (back end only), Magento

Possible Extensions

- Calculate the markup on items in the store, and display it in the inventory
- Filter the inventory list by manufacturer. For example, provide an option to view all books in stock by a certain author.
- Categorise your items. Books might be categorised by genre (crime, horror, romance...) and cars might be categorised by type (SUV, coupé, hatchback...). Provide an option to filter the inventory list by these categories.

Travel Bucket List

Build an app to track someone's travel adventures.

MVP:

- The app should allow the user to track countries and cities they want to visit and those they have visited.
- The user should be able to create, edit and delete countries
- Each country should have one or more cities to visit
- The user should be able to create, edit and delete cities
- The app should allow the user to mark destinations as visited or still to see

Possible Extensions:

- Have separate pages for destinations visited and those still to visit
- Add sights to the destination cities
- Search for destination by continent, city or country
- Any other ideas you might come up with

Sports Scoring app

Build an app that let's a sportsfan keep track of their favourite sports league.

MVP:

- The app should allow the user create, edit and delete teams to the league
- The user should be able to create, edit and delete games
- There should be a way to display all the games for a team and all the teams involved in a game
- The app should display if a game was won or lost

Inspired by

Sky Sports, BBC Sports

Possible Extensions:

- Create a league table to see who is on top of the league
  - You might not want/need to create a table in the database for this, but just use Ruby logic in the models to calculate each teams points etc, based on the results of games
- Add players to teams and let them have game stats (points won, fouls, etc.)
- Anything else you can think of

Submission

Submit a link to your Github repo through the Homework Submission form as soon as the repo has been created.

Your code will be reviewed with an instructor on Thursday afternoon.

Presentation

Presentations will be held on Friday. You will have 10 minutes to present your work, with 5 minutes for questions.

We expect you to:

- Show your sketches, notes, diagrams, wireframes and user planning
- Demonstrate your app's functionality
- Talk about some code that you are particularly proud of
- Highlight your main learning points

Do not:

- Talk through every single line of code
- Go over the time limit

Asking for help

The instructional team will be on hand for technical assistance, but you are strongly encouraged to take this opportunity to practise self-sufficiency, as will be expected of you as a professional software developer.

Before asking an instructor for help this week, you should be ready to answer the following questions:

- What are you trying to achieve?
- What do you think the problem is?
- What have you tried so far?
- What did Google say about your error message?

Attendance

The cohort will meet at the usual time every morning. Your attendance is recorded every day as normal.

PDA Reminder

Remember to gather evidence for your PDA this week. This should only take 5 minutes:

- Go to your PDA Checklist
- Submit your PDA evidence (screenshots, etc.) to your own PDA repo