# Springboard

**Software Engineering Career Track**
**Capstone Project Two: Overview**

# Project Steps

We have broken down the Capstone Project into easy-to-follow steps. Each step of the capstone contains a link with instructions for that step. You may notice this second Capstone follows a similar pattern to your first Capstone, however, there are key differences. Here's a quick overview of what you'll do for each step of your capstone project:

1. Step One: Initial Project Ideas
   a. You'll pick up to 3 project ideas to propose to your mentor and the Springboard community. You'll also explore some potential APIs.
2. Step Two: Project Proposal
   a. For this step, you'll write a proposal for the site you want to build. This will help your mentor better understand your chosen capstone project idea.
3. Step Three: Source Your Data
   a. After your mentor approves of your capstone project proposal, you'll figure out the database design of your application and where your data will come from. You may choose to use an existing API or create your own.
4. Step Four: Coding User Flows
   a. Once you've figured out what you're building, you'll write the code to implement it. It's important to think about what you want a user's

experience to be like as they navigate your site.

5. Step Five: Polishing Your Application
    a. Once you have the core functionality implemented, you'll focus on additional UI enhancements and styling for your application. This is wher you will implement your bells and whistles.
6. Step Six: Documenting and Submission
    a. You've done a lot of work so now it's time to show your mentor your progress! Create a README in markdown, make sure your GitHub is organized, and submit your finalized project.

---

# Overview

For your second Capstone Project, you'll build a more complex database-driven website. Most students will choose to develop this app in React and Node, however, Flask/Python is also an option if you tackle a difficult idea. This website will be powered either off of an external API or an API that you build yourself. Your finished capstone will be an integral part of your portfolio; it will demonstrate to potential employers everything you've learned from this course.

We want you to work on a challenging project that will incorporate all of the full-stack skills you've been developing. The goal of this project isn't to create something that's never been done before but should be more ambitious than your last capstone. You could potentially create a website similar to one that already exists, but this time, perhaps add a feature that you wish the website had. We do encourage you to be creative when building your site. You're free to choose any API you'd like to use or build your own. We encourage you to tap into your imagination throughout the project.

---

# Examples

You already know about the wealth of APIs available online. Perhaps on this capstone, you can work on one of your ideas that was a bit too complicated for the last project. We also encourage you to create your own API if you cannot find one with the data you are looking for. You can do this through web scraping, importing a CSV, or loading your own data into the API.

Let's give you an example of what a site could look like. Say you want to make a website or mobile app that was like Facebook for dogs - something that would allow pet owners to connect with other pets in their neighborhood. First, you could load information into the application about various breeds of dogs, which would populate drop down lists and allow users to sort for the kind of dog they would like to sit. This will help users build the profile for their animal. You could add forms with various information about the pets. You could allow them to upload pictures (dog owners love nothing more than to take pictures of their animals). Most importantly, you could allow the pets to connect with other pets through a graph.

Now let's talk about bells and whistles. What if a user of your Dogbook was leaving town and wanted to find users in their neighborhood to watch their dog for the weekend. You could implement a geographical filtering and simple messaging or request system in order to help Spot find the best pet sitter. And since no one wants their dog watched by some kind of monster, you could implement reviews to see if people recommend this sitter.

There are a million different features you could add! Verified users, so celebrities could show off their dogs. Hafthor Bjornsson, the actor who plays the Mountain on Game of Thrones, has an adorable pomeranian and people demand picture proof! You could implement an adoption system so people can give shelter pets a good home. Of course, adding in all of these features would be beyond the scope of this project, but you should expect this app to have more functionality than the last Capstone.

# Guidelines

1. You can use any technology we've taught you in the course, and there's nothing stopping you from using outside libraries are services. That being said, we recommend you use React, and Node.js for this Capstone. If you completed the optional Redux unit, we recommend you use Redux as well. You can use Flask/Python but will be expected to make a much more fully featured application than last time.
2. Every step of the project has submissions. This will alert your mentor to evaluate your work. Pay attention to the instructions so you submit the right thing. You will submit the link to your GitHub repo several times, this is for your mentor's convenience. Your URL on GitHub is static and will not change.

3.  The first two steps require mentor approval to proceed, but after that, you are free to continue working on the project after you submit your work. For instance, you don't need your mentor to approve your database schema before you start working on your site. Likewise, you don't need your mentor to approve the first iteration of your site before you start polishing it.

4.  If you get stuck, there is a wealth of resources at your disposal. The course contains all of the material you will need to complete this project, but a well-phrased Google search might yield you an immediate solution to your problem. Don't forget that your Slack community, TAs, and your mentor there to help you out.

5.  Make sure you use a **free API** or **create your own API** and **deploy your project on Heroku**, so everyone can see your work!