

# Final Project For Stream Two

## PROJECT BRIEF

---

The project guidelines are as follows:

In this project you'll be building a data-driven frontend and backend website using the technologies that you have learned throughout Streams 1 and 2.

You can either choose to use the example brief below or you can use your own idea for the website.

1. Get Yourself Elected
2. You (the student) are running for office as the governor of the US state of Arkansas (this state was randomly chosen just for you :D). As part of your campaign you would like to tell your potential voters about how bad the funding for education is in your state and what you would do to change it.
3. In order to achieve this, you will use the school donations data to inform them of the lack of funding in the state. This information should be based on true data from the school donations database, but it doesn't have to include all of the data.
4. Choose the pieces of data that will help you tell your story in the most effective manner.
5. Make sure to include additional pages to describe your platform and promote your plans regarding education. Feel free to include (or invent) any additional data or detail that would support your campaign.
6. Create Your Own Project (Recommended)
7. If you choose to create your own project, the scope should be similar to that of the example brief above. If you want some ideas, please ask your mentor for advice and direction.

## PROJECT GUIDELINES

---

- The data that you choose to work with, should be stored in either an SQL database (preferably MySQL), or a noSQL database (preferably MongoDB)
- The project must use Flask to retrieve the data from the database and return it to the browser.

- The dashboard should include, at the minimum a line/bar graph and a pie graph. Add any additional graph types that may be relevant to your dataset.
- The front end must use D3.js to display the data and there must be some way of interacting and filtering the data in the charts using DC.js and crossfilter.js, or another equivalent.
- Use as much functionality as you deem necessary from the lessons
- Use Github for version control
- You should deploy the final working version of your code to Heroku (or an alternative hosting platform that you are familiar with)
- Make sure your site is as responsive as possible. Use appropriate testing sites to test your web sites in several different environments
- Write a README.md file for your project that explains what the project does and the need that it fulfills. It should also describe the functionality of the project, as well as the technologies used. Detail how the project was deployed and tested and if some of the work was based off other code, explain what was kept and/or how it was changed to fit your need. A project submitted without a README.md file will **FAIL**.

## ADDITIONAL ADVICE

---

- We advise that you create wireframes before embarking on full blown development.
- Incorporate as much as you have learned in our lessons.
- The site can also make use of CSS, Javascript and Python libraries/frameworks, just make sure you maintain a clear separation between the library code and your own code
- Please refer back to your mentor to get constant feedback as you progress
- Don't forget to use your mentor's help. Your mentor can act as:
  - Your client
  - Your technical consultant

**Good luck and enjoy!**