CS50's Web Programming with Python and JavaScript

OpenCourseWare

Donate (https://community.alumni.harvard.edu/give/59206872) (https://community.alumni.harvard.edu/give/59206872)

Brian Yu (https://brianyu.me)
brian@cs.harvard.edu

David J. Malan (https://cs.harvard.edu/malan/)
malan@harvard.edu
(https://www.facebook.com/dmalan) (https://github.com/dmalan) (https://www.instagram.com/davidjmalan/) (https://www.linkedin.com/in/malan/) (https://www.quora.com/profile/David-J-Malan) (https://www.reddit.com/user/davidjmalan) (https://twitter.com/davidjmalan)

Capstone

Designing and implementing a web application of your own with Python and JavaScript.

Overview

The final project is your opportunity to design and implement a dynamic website of your own. So long as your final project draws upon this course's lessons, the nature of your website will be entirely up to you.

Requirements

In this project, you are asked to build a web application of your own. The nature of the application is up to you, subject to a few requirements:

- Your web application must be sufficiently distinct from the other projects in this course (and, in addition, may not be based on the old CS50W Pizza project (https://docs.cs50.net/web/2020/x/projects/3/project3.html)), and more complex than those.
- Your web application must utilize Django (including at least one model) on the back-end and JavaScript on the front-end.
- Your web application must be mobile-responsive.

> The most common cause for failure of the final project is not spending enough effort on this next instruction. Your `README.md` file should be minimally multiple paragraphs in length, and should provide a relatively comprehensive documentation of what you did and, if applicable, why you did it. Ensure you allocate sufficient time and energy to writing a `README.md` that you are proud of and that documents your project thoroughly, and that distinguishes this project from others in the course and defends its complexity. Be proud of it!

- In a `README.md` in your project's main directory, include a writeup describing your project, what's contained in each file you created, and (optionally) any other additional information the staff should know about your project. This file should also provide your justification for why you believe your project satisfies the distinctiveness and complexity requirements, mentioned above.
- If you've added any Python packages that need to be installed in order to run your web application, be sure to add them to a `requirements.txt` file!

Beyond these requirements, the design, look, and feel of the website are up to you!

How to Submit

Hold on! If you have already submitted and received a passing grade on the **prior version of the Final Project (https://docs.cs50.net/web/2020/x/projects/final/final.html)**, please stop here. You already have received an equivalence credit for this project, and you should not submit this assignment, as it will have no impact on your progress in the course and will therefore only slow our graders down!

1. Visit this link (https://submit.cs50.io/invites/89679428401548238ceb022f141b9947), log in with your GitHub account, and click **Authorize cs50**. Then, check the box indicating that you'd like to grant course staff access to your submissions, and click **Join course**.
2. Install Git (https://git-scm.com/downloads) and, optionally, install `submit50` (https://cs50.readthedocs.io/submit50/).

When you submit your project, the structure of your `web50/projects/2020/x/capstone` branch should match, in general, the file structure of Projects 1, 2, 3, and 4. Your branch should also not contain any code from any other projects, only this one. Failure to adhere to this file structure will likely result in your submission being rejected.

Your `README.md` file must also be at the highest level of your project. That is to say it should exist at `https://github.com/me50/USERNAME/blob/web50/projects/2020/x/capstone/README.md` (where `USERNAME` is your own GitHub username as provided in the form, below). If it doesn't, reorganize your repository as needed to match this paradigm.

3. If you've installed `submit50`, execute

```
submit50 web50/projects/2020/x/capstone
```

   Otherwise, using Git, push your work to `https://github.com/me50/USERNAME.git`, where `USERNAME` is your GitHub username, on a branch called `web50/projects/2020/x/capstone`.
4. Record a screencast (https://www.howtogeek.com/205742/how-to-record-your-windows-mac-linux-android-or-ios-screen/) not to exceed 5 minutes in length, in which you demonstrate your project's functionality. Be certain that every requirement of the specification, above, is demonstrated in your video. There's no need to show your code in this video, just your application in action; we'll review your code on GitHub. Upload that video to YouTube (https://www.youtube.com/upload) (as unlisted or public, but not private) or somewhere else.
5. Submit this form (https://forms.cs50.io/e935db49-f50a-4693-9b18-95ed1fdbf381).

You can then go to https://cs50.me/cs50w (https://cs50.me/cs50w) to view your current progress!

IMPORTANT note regarding your gradebook at <cs50.me/cs50w>! Soon after the start of each calendar year (we estimate during the week of 11 January 2021), we typically refresh all of our gradebooks. *All of your work from 2020 will be saved and will carry forward*, but your gradebook may appear temporarily empty or unavailable until you have submitted work in 2021 that has been graded and returned to you by CS50 Bot. So don't be alarmed!

If you finish CS50W during the final days of December 2020, it is very important that you claim your free CS50 Certificate before this gradebook reset takes place, so don't delay! (Verified certificates are unaffected by this.)