Project 1 Debrief

15-113 – Monday 2/2

20-25 minutes: Talk with your neighbors

- Break into groups of 4-6 people
- Take turns briefly showing your website
- Discuss which models you used, any difficult parts, anything that seemed particularly effective
- Take notes! We'll ask people to report back briefly

We'll (anonymously) share these results

- Fill this out! https://bit.ly/4tduVKY

A few very cool portfolios

We tried to showcase a broad range of designs, so please don't feel bad if you're not featured here

- https://rionkuri22.github.io/
- https://koljahribar.github.io/
- https://rachaelchung.github.io/
- https://ivyflosvv.github.io/
- https://solllllz.github.io/github.io/
- https://treybrosnan.com/

Some observations from the TAs

Models

- Deepseek and Claude yielded the least problems
- ChatGPT common but typically required more debugging /prompting
- Pro models typically outperformed free-tier
- Several students initially tried several models before settling

Some observations from the TAs

Some common strategies

- Provide detailed prompt including all features, plus context like other websites, project requirements, resume, project list, etc
- Create a website with placeholder content and then manually modify
- Start with a basic structure and then iteratively prompt for additional features one at a time
- Ask for AI to generate snippets of code rather than rewriting the whole website each time
- Note: Starting with a website template seemed difficult. AI sometimes had trouble making sense of the structure.

Some observations from the TAs

Debugging

- Ask for AI to generate snippets of code rather than rewriting the whole website each time (sometimes avoids introduction of additional bugs)
- Make small adjustments manually to avoid ‘fighting the LLM’
- Pivot the design if AI can’t seem to achieve the desired result
- Try debugging with different models
- Try asking a human for feedback

Some observations from the TAs

General suggestions

- Don’t be content with the first result. Know when to re-prompt. Having a strong starting point is critical.
- Try different models even if you have a favorite
- Context is important! Additional examples and documents can yield much better results.
- When iterating, watch carefully for unintended changes
- Not all suggestions are useful. Know when to discard changes and re-prompt or change course.

API intro

Let's try to implement a basic API call. APIs allow your code to get data from the web and/or use useful services.

Some options:

- Stocks / Finance data
- Weather
- Advice / Inspiration / Quotes

Some Python APIs:
https://github.com/realpython/list-of-python-api-wrappers

More general APIs by category:
https://github.com/public-apis/public-apis