Overall, I am very happy with my finished product — a functioning fullstack e-commerce site that addresses a niche and underserved market: the intersection of the gaming and anime communities. The hardest features to implement were: the modal, the contact form (backend in particular), and determining an appropriate SQL data schema. My general approach in all cases was to begin with the slides given in CS 132 and when I could not find solutions, I would turn to the internet with a careful eye on what code would be considered good style vs. bad style. For example, there are many examples of modals online but many of them have redundancies, scrappy bode, and poor style. I developed my own implementation of a modal which, in my opinion, is cleaner than anything I have seen on the internet. Overall, I would say my ability to combine knowledge from the lecture slides in the class and carefully determine what on the internet can be used and what must be tossed out was a key component of my success in this project.

As for failures, I do not think I ended with any since I spent an obscene amount of time on this project making sure every nook and cranny was ironed out to perfection — from code abstraction to rewriting some GET endpoints into POST endpoints once I realized POST was a better fit to rewriting part of cart.js to use Promise.all to ensure the ordering of the cart stayed in the same order while not taking too long to load, I feel like I addressed every possible concern both from a user external perspective and a code-cleanliness internal perspective.

I would say the most rewarding part of it all is to finally have a functioning full-stack project up and running where I wrote the backend. I generally work on full-stack projects with Eugene (my roommate) and I usually do the frontend while he

does the backend. We originally wanted to collaborate on the API together but due to scheduling differences, we each wrote the entirety of our own backends ourselves. This makes this project the very first fullstack project where I have written every bit of code from scratch, and it feels nice as hell.

I felt that the final project proposal checkpoint was awesome. I ended up writing ALL of the frontend code I would need for my final project by then even though the requirements were significantly less. For future students, I would recommend doing the same. I naively believed that the backend would be as trivial as a few function calls, but setting up the backend took more time than I originally thought (relearn SQL, make your data schema, understand Express, write GET APIs, write POST APIs, do fetch calls in your frontend, properly handle errors, etc.) Backend takes longer than I thought so I would definitely advise students to have all the frontend code finished by the final project proposal deadline.

To wrap everything all up, this project was a great way for me to explore both frontend and backend, and I now feel like I have an awesome grip on how to be a fullstack dev. It's a great way to finish my time off at Caltech, and I'm excited to use these skills going forward