FINAL PROJECT REFLECTION

Features Implemented

For the Final Project, I implemented three primary options that met the course requirements. The first was a promotions view, which I represented through the Limited Edition Potions section. This section showcased festive and seasonal potions, like Christmas-themed potions, with a countdown timer dynamically displaying the time left for each promotion. I used a GET endpoint to filter and retrieve only the limited edition potions by setting a query parameter (`is_limited_edition=true`). The visual elements, like timers and festive themes, emphasized the limited-time nature of these potions, with discounts clearly specified to engage users.

The second feature was a fully functional purchase flow. Users could buy potions from the store and receive a confirmation message indicating their total cost and that the order was on its way. This process updated the stock through a POST endpoint, which decremented the stock of ingredients on the server with each purchase. For example, when a user bought a custom potion, the stock count for the ingredients the user used was updated immediately, ensuring accurate inventory tracking and deducting required ingredients from the stock.

The third feature was product customization, which I implemented through the Brew Your Potion page. This feature allowed users to create their own unique potions by selecting a name, colour, and magical ingredients. The page dynamically fetched available ingredients from the backend using a GET endpoint, and users could see the real-time total price of their custom potion based on the selected ingredients. Once completed, the custom potion was sent to the backend via a POST endpoint, where it was saved along with the user's specifications. These custom potions were added to the cart and purchased like pre-made potions.

Hardest Features to Implement

The hardest feature to implement was the backend API. This was my first time working with APIs at this level, and I found it challenging to design and integrate endpoints that worked seamlessly with the front-end. For example, setting up query parameters for filtering limited edition potions required understanding how to structure and parse these parameters on the server side. Additionally, the logic for deducting ingredients from stock when custom potions were created added complexity, as I had to ensure consistency between the ingredients used and their availability. To approach these challenges, I referred to the work I had done for CP4, consulted El's online documentation, and used Office Hours to clarify doubts. Peers also provided valuable input when discussing implementation strategies. While difficult, this experience helped me build a solid foundation for working with APIs.

Most Helpful Resources

The most helpful resources for this project were primarily external references and a few course-provided tools. While the course slides were available, they were too extensive to sift through efficiently, so I relied on MDN Web Docs and Stack Overflow for quick solutions. The cheatsheets on Canvas were concise and particularly useful for recalling specific concepts with Node.js commands. Live coding sessions during lectures were helpful when debugging, as they demonstrated practical ways to identify and resolve errors. Tools like Nodemon taught in lecture and OH also proved invaluable, allowing the server to restart automatically during development and enabling me to test changes efficiently. OH with El provided clarity on backend logic.

Additions with More Time

If I had another week, I would focus on improving the aesthetic and functional aspects of the project. In the Limited Edition Potions section, I would add a falling snowflake animation to enhance the festive theme and make the page more visually engaging. For the Brew Your Potion page, I would refine the CSS to make the customization process more interactive, potentially adding animations like bubbling effects when ingredients are added. Additionally, I would expand the API to include more GET endpoints, such as retrieving a potion's history or most popular combinations. This would add depth to the user experience. Finally, I would work on backend validation for custom potions to ensure users only selected valid combinations and integrate these potions dynamically into the pre-made potion list, making them visible to all users. I would also try and implement the option of letting users leave reviews on potions as well.

Most Rewarding Part of the Project

The most rewarding part of the project was seeing the website function as a cohesive, interactive product. Watching the dynamic interactions, such as the cart updates, potion filtering, and the custom potion creation process, made the effort worthwhile. For instance, when I successfully integrated the countdown timer for limited edition potions and it displayed the correct remaining time, it felt like a major accomplishment. Similarly, debugging and finally seeing the "Potion added to cart!" message work seamlessly gave me a sense of satisfaction.

Breakdown and Checkpoints

The breakdown and checkpoints for the assignment were helpful in managing the workload and ensuring progress was made steadily. However, I believe the project would have benefited from more dedicated time. Ideally, there should have been a two-to-three-week period exclusively for the Final Project, with no other assignments due during this time. This would have allowed me to focus more deeply on refining the project and addressing any complex issues without feeling rushed.

Advice for Future Students

My advice to future students would be to start early, especially with wireframing and defining the API structure. Building the backend first simplifies the integration of front-end features later in my opinion. It's also important to test endpoints frequently during development to catch errors early. While working solo, I found it helpful to discuss ideas with peers and seek their input on potential features or solutions. Finally, taking advantage of resources like cheat sheets, live coding sessions, online documentation and the OH can significantly streamline the development process.

This project was a challenging yet rewarding experience that allowed me to build a full-stack application with dynamic interactivity. Despite the difficulties, I am proud of the final product and the skills I gained through the implementation process!

Please fill out the following and submit with your Final Project submission (added to help us know what to focus on when grading, what was intentionally incomplete, etc.):

1. *What is your estimated grade for the Final Project (roughly of 100)? For partner projects, do you think the workload distribution was fair?*

I completed both the back-end and front-end development independently, with some helpful guidance from El during office hours to review and improve my code. Based on the effort I put into meeting the specifications, implementing additional thoughtful details, and creating comprehensive APIDOC documentation, I would estimate a grade in the 90s. I believe the project fulfills all the requirements outlined in the spec, and I worked hard to ensure a polished final product.

2. *What is your estimated grade for the course (letter grade)? You can support your estimate with examples of demonstrated engagement in the course material, strengths (& weaknesses) over the term, and growth throughout the material (some assignment grading feedback was delayed, but we would like to know what resources you used to iterate on the material, such as lectures, OH, Discord, etc.)*

While I'm not entirely familiar with the grading system for letter grades at Caltech, I would expect a high grade overall based on my consistent engagement and effort throughout the course.

- Strengths & Growth: I dedicated significant effort to every project, documentation, and code submission. At the start of the course, I struggled with using flex, needing guidance during OH. By the end, these concepts felt second nature, and I was able to independently implement layouts and designs. This growth has been immensely rewarding.
- Engagement: I have tried to actively participate in OH, utilized lectures and cheat sheets as resources, and continuously iterated on feedback to improve my work. The projects, while challenging, allowed me to explore creative ideas and strengthen my skills. Building on the projects every week by adding different aspects to it was rewarding once I saw the finished product.

3. *To what extent would you like feedback in your project (e.g. thorough, particular areas of the project, or just a grade)?*

I would appreciate thorough feedback, particularly on areas where I could improve. Suggestions on how to strengthen specific aspects of my work (e.g., CSS or API design) would be great. Of course, I would also love to know my grade for the project!

4. *Anything else you would like El to know when grading?*

I think I have put in 3 all nighters into the project, worked the best I could and I don't think I could have honestly done more. Managing both the back-end and front-end on my own was a challenging but rewarding experience. While I recognize that my CSS could be improved, I believe I did my best given the circumstances and the workload.

5. *Are you open to sharing your project/reflection for future terms? If so, would you like named credit or anonymous?*

Yes, I'm open to sharing my project and reflection. If used, I would prefer named credit.