# Brainstorm stories:

- Learning flexbox, having Stu walk me through the importance of having your elements in containers, and how each container effects the one below it.

- Creating my project, and utilizing classes to build the information for both the NFT and the artists that are demonstrated on my page.

- Taking a squads morning prompt and turning it into a functioning login page, that encrypts and decrypts passwords w/ bcrypt.

- Understanding the concepts of endpoints.

- Understanding data modeling, and the concept of junction tables, was really enlightening.

# First drafts:

## Flexbox:

Working with CSS has always been a struggle. I was working on an experiment to try and build a functioning budget builder, using an image as a goal. The functionality was less difficult to get going, so I built as much of that out first that I could. However, all of the elements on the page would not be placed where I wanted them to. Doing a meeting with one of the mentors, helped walk me through the concept of Flexbox, and the importance of organizing my HTML in a more structured way. This helped me then use display flex to put everything where I wanted it.

## Login-page and Postman:

One morning, in our squads, we have a prompt to build out a password checker, using bcrypt (I think the day after we did the security lecture). I wanted to use it as a way to practice both my CSS skills (I was just getting comfortable with Flex) and my frontend to backend calls. So I was determined to build a webpage where you could register with a username and password, then use that account info, and login and get a success message if it worked properly. This was the first time I tried simply building a functioning backend first, and I ensured that the express calls functioned properly using Postman, before I bothered typing a single line of code on the frontend, or even before having any HTML to work with. This was a big learning lesson.

## NFT artist/art classes:

Reflecting on what we learning about classes a couple weeks ago, I realized that I could use them to structure my project data in a consistent way. I knew I wanted to have the server (in a real project I would store it on a database, but I didn’t get that far), store the important information about these, so I created two classes that would help the project scale in the future. On my current project, I created some objects within the code, which isn’t ideal, but since I have no database and I needed data to receive from the server to show on my webpage once I formed that, it seemed like the most simple process. I built in both classes a method on construct that pushes the class-object to an array to be referenced. If I scaled the project, these classes could easily be used to create actual profiles for artists and art, and be sent to a database.

# Final Drafts

## Flexbox:

Working with CSS is where I have experienced the most struggle in my coding experience (so far). One day, we were given an option challenge to build a functioning budget builder app, using only a single image as our guide. The functionality was less challenging for me, so I built out the ability to add a name and amount of a transaction, the ability to keep track of the total amount of all transactions, and the ability to delete a transaction. However, when I worked on the frontend portions, getting everything to load onto the page, where I wanted them to load, was difficult. Everything was completely unorganized.

With the help of one of our mentors, he walked me through the concept of using Flex on a single element to get it to show on the page how I wanted. In addition, he also taught the importance of structuring your HTML in a way that makes it easier to organize within CSS. After the walkthrough of getting one element on my page to do what I wanted, it took me a much shorter amount of time to finish the rest and got it functioning how I wanted that afternoon.

## Classes:

After learning about classes, I understood why they were important in data analytics. However, I struggled to find a practical example of how to use them in what I was working on.

When building my project, the NFT Hub, for my foundations course, I knew that I wanted to have consistent data about my app’s artists and artwork stored on the server, and potentially a database if I build that out one day. It gave me the opportunity to create two classes with the relevant information for each, and prompted me to experiment with methods, including one that gets called on the object when it’s instantiated, which adds it to an array I could use on my front end, so I didn’t have to create a function to add each object myself.

It make it much easier to organize the data I received from the server, and I believe is scalable if I ever add a database to expand the project.