# Mixer Sprint 2 Dev Notes - Austin

Moving over from sprint 1, we knew there would be a lot of work to put in. We needed to add proper drag and drop (no in-line “onclicks” or anything), prevent item stacking, allow all three divs on the sheet to be droppable, queue the sounds on the drop, and allow them to loop. Some things as simple as adding “loop” to each audio source. Some things were tedious and definitely not optimized, like adding the audio triggers. And some things were just plain undoable for right now, like getting the play and pause to work. Check below for a in depth analysis of each change, bug and whatever else I can think of.

## Drag and Drop

So before the puzzle fix, this would have been a nightmare. In the end we used the first half of the puzzle fix to run the drag and drop, as it was already established that it worked. I cleaned up the code for the stack prevention so that it just consists of 3 lines instead of the 8 (I think) it was previous. The experimentation phase provided a ton of valuable information though, and I plan on starting a new branch specifically for testing cloning after we do our review on March 11th. I also tested with dragging the whole div instead of just the image, and while this worked, it would only work on the first div. Definitely need to figure out how to target all divs and allow the drag, that way more than just the image moves. But with cloning, this wouldn't be a problem as I could match the audio and cloned Id together and have the user be able to drag and drop multiple of the same sound.

## Sound Queues on Drop

We used a medieval way of allowing the drop. This is something I really need to work on. I just added “if” statements for when that particular image is dropped, and it fires the sound and loops it. There is for sure a way to bind these to just a few lines with reviewing the DnD Puzzle as a reference, but I could not figure it out. But it works, and it works quite well. This is obviously an overworked portion that I personally Need to work on. As for the loop, that’s simply just adding the “loop” tag to each audio element.

## Play and Pause

Upon review, the pause button can be used, by just giving each audio element the same class, targeting the class with querySelectorAll and using the (arrayname).pause trigger. Although I have yet to put this in practice. As for play, this could possibly be done by targeting the audio if its inside of the drop zone and just playing it, but at the time this was overkill for where we are right now. We felt that it was a bit counter-intuitive to have a play and pause button since the audio plays on drop, but that’s something for design to talk about.

## Conclusion

In the end, where we are is in a great place. We have every core piece working as intended, we have a solid design that could use some slight tweaks down the line, and any change here on out is just additions/wanted changes, not needed.