1. **What did you do last week?**

One thing I forgot when making the handler for /list\_furniture is that I needed to associate each listing with the user that created it; I needed to attach the userID to the listing before inserting it into the DB, and I just forgot that completely. So, I made sure to fix that. After that was done, I pretty much finished the handler for the /list\_furniture endpoint and the integration test for it.

I also revised some types for my session manager and made sure the tests for that worked as well. And, I also implemented the /logout handler and wrote a test for that.

1. **What do you plan to do next week?**

I plan to implement the handler for /account, which returns information about the user, such as their username, email, phone, and all their furniture listings. I have an Account dashboard on the frontend, so this endpoint is necessary. /account might be changed to something else, but I don’t know yet.

There might be other things I do if I finish it early, such as implementing the handler for checkout requests (or at least defining the inputs and outputs) and learning about Stripe

1. **What is blocking you from going forward?**

I am at a block at how I should return data for certain requests. For example, in my /account handler, or whatever it would be, I plan on making it so that the client can fetch all of their furniture listings, which includes information for every furniture they’ve listed for sale, such as cost, descriptions, titles, materials, images, etc.

The question is, should I not care about scaling and just return all that data at once because this is not a project that’s going to be put out into the world for people to use? Or should I care about scaling and learn methods to deal with sending large amounts of data over the network to the client because I’m also taking distributed systems and this would be great knowledge?