Week 1

The first thing that impressed me most this week was an activity in the applied course where I was assigned to a "wallet design" assignment. I was assigned to a small group and asked to design a wallet prototype based on my teammates' actual pain points, rather than just randomly coming up with features. My teammate designed a wallet that perfectly suited my needs. I could tell she listened carefully to my suggestions and worked hard to implement them. This made me very happy because it felt like a real-world example of the "user-centered design" we discussed in class. I designed a larger wallet for her, with a separate pill box and card slot. She said it was very convenient, and hearing that made me feel that my initial inquiry into her needs was worthwhile.

Looking back, however, I still see some shortcomings. I relied too much on my intuition and didn't ask more detailed questions. For example, I only asked if she needed to keep her medication, but didn't confirm the actual size and number of pills. Providing these details would have made the design more precise. Next time, I'd like to ask more structured questions.As a result, while the prototype had the right direction, the details weren't quite right for her daily life. I think this reminds us that "user-centeredness" shouldn't be based solely on intuition and a few key points; it must be translated into concrete evidence. If I were to conduct a similar activity again, I would start by making a list of three questions: What do I need to carry with me every day? When do you use it most often? What's the most frustrating pain point? I'll then draw two sketches, have her quickly compare them, and choose the most appropriate one.

The second incident happened during a Friday team meeting. This was my first official experience with LeanKit. At first, I thought it was novel; putting all the tasks on one board felt very clear. But after using it, I realized our cards were too large, and many contained goal-level items that weren't broken down into smaller tasks. The user stories weren't detailed enough either. As a result, the board looked full, but the priorities and review methods remained unclear. At that moment, I felt a little confused; it felt like I'd only changed the presentation, but the tasks themselves hadn't become clearer.

From this experience, I learned that tools alone can't solve the problem; how they're used is crucial. To make LeanKit truly effective, I had to break down tasks into granularity that could be completed within a day or two, and I had to consult with my

teammates. Each card was clearly labeled with "Specific Deliverable," "Sufficiently Small User Story," "How to Review," and "Prerequisites." My plan is to make sure that the acceptance criteria and dependencies are clearly written in the future, and that each card is fully approved by the team before comparison. This way, I can not only see the progress of the task, but also reduce rework and blockage.

Week2

The first half of the week focused on reviewing and presenting the analysis and design report. I carefully read the report and then shared the key points with my advisor. My primary responsibility was the safety plan. Although I was quite well prepared, I was still nervous during the presentation, especially because my slides and formatting weren't very polished. I was pleased with my clear logic and the flow of some of my presentations, but there were also areas where I was overly general and lacked concrete supporting evidence. Upon reflection, I realized that my preparation was more about "writing for myself" than considering potential questions an external audience might have, especially regarding the architectural diagrams. Next time, I need to practice more thoroughly before the actual presentation and anticipate potential issues.

Another highlight for me this week was the marshmallow challenge. I quickly came up with a plan, assigned my own tasks, and successfully built a stable tower on my first try. I was delighted at that moment because I didn't have to start from scratch or waste time correcting mistakes. However, looking at the other groups, I was a little disappointed because our tower wasn't as tall. I realized that I had been playing it safe, choosing stability over pushing the boundaries. I didn't create a better plan for my team.

Looking back, I gave up too early. Once I had a working version, I didn't take the time to explore how to improve it. I've learned that a quick win doesn't necessarily mean the best solution. If I had more time, I could have tested a few variations of the security design to see how far I could push it and gain the trust of my teammates. For future projects, I want to remind myself not to settle for "it works," but to ask myself, "Can it be better?"

Week 3

This week, I've essentially completed the build phase. Since I'm responsible for the front-end, the overall functionality is running smoothly. During the demo, I quickly

discovered some issues. Previously, I'd been considering "just working code" as my standard for completion, but my mentor immediately pointed out details like the misaligned text in the navigation bar and homepage options, which looked unprofessional. I was a bit frustrated at the time, as these issues were easily fixable, but I hadn't considered the audience's feelings. After being pointed out, the entire demo looked a bit rough. I was happy that I could finally present a complete version; what frustrated me was the lack of polish.

Looking back, this was because I was too focused on implementing the functionality and neglected the user experience and presentation. I find it's easy to automatically feel "done" once the code is running. But in reality, audiences and users are more concerned with the overall impression and the professionalism of the details. I've decided to create a small checklist for myself, such as ensuring consistent navigation text, aligned copy, and clear buttons. I'll run through this checklist after every build to prevent minor issues from impacting the overall effect.

This week I also recorded our onboarding team reflection video. This was my first time expressing my thoughts in video form. I was quite nervous at first and my speech was a bit halting, but as I gradually got into the flow, it became much more natural. I found this process quite unique because, as I reflected on my thoughts as I spoke, I was able to more clearly see my strengths and weaknesses. My teammates also gave me feedback, and I was happy that I was genuinely reflecting on my work, not just trying to get by. Unfortunately, the video wasn't very smooth, with occasional pauses mid-sentence, making the transitions feel a bit awkward.

I think this is because I previously thought "just chatting" would be fine, but actually, recording a video requires more preparation and planning. This made me realize that when facing an audience or camera, simply having an idea isn't enough; structure is also crucial. Next time, I'll write a three-sentence summary to tie together the key points and practice it before recording. This way, the video will come out more naturally and more organized. I hope to be able to record it in one go, eliminating the need for so much time spent on editing and re-recording.

Week 4

This week I spent some time coming up with a possible theme for my main project. I spent some time doing research and thought the idea I came up with was very creative, but in the end it was rejected. The problem is that it relies on hard-to-obtain external data, and for the short period of time we have, it might be too broad. At first,

I felt a bit discouraged because I was really working hard and truly believed it had potential. Later, I realized that I was rejected not because of me, but because of practicality. This made me realize that when I come up with an idea, I need to find a better balance between creativity and reality.

Looking back, I think I could have been better prepared. If I had searched the data source earlier and presented a brief example, it might have made this idea seem more feasible. Even a small prototype or a brief demonstration can make it look more concrete and less risky. I shouldn't just talk about concepts; I should offer something that people can actually see. Next time, I will make sure that my idea has at least a small amount of evidence or draft to support it, so that people can imagine how it works in practice.In the future, I plan to evaluate my ideas with simple questions before sharing them: Can I find the data? Can I provide a simple example? Is it feasible within the time we had?

The highlight of this week for me was speaking to industry guests in the studio. I was nervous, and I could tell my speech wasn't as fluent as I'd like. I also noticed that when the panelists asked questions, I remained silent instead of stepping up to help my teammates answer. Afterward, I felt a little disappointed with myself because I missed an opportunity to show more confidence and participate in the discussion.

I think my lack of confidence stemmed from not preparing my talking points in a concise and clear manner. I knew the material, but I hadn't practiced speaking it out loud, so when it came time to speak, I stuttered. Next time, I plan to summarize my key points in three sentences, practice it once or twice, and remind myself that even if my responses aren't perfect, I can still chime in. My goal is to become more fluent and support my teammates by sharing responsibility during the Q&A session.