Adrien Protzel
Benjamin Hutkoff
Erick Branner
Chitali Buge

Reflections

Adrien Protzel:

  I have learned how to make clean prototypes using Figma and how to make a basic framework for mobile apps using Kotlin and XML. I also learned more about scheduling and organization tasks on Asana and Discord. Lastly, I have learned more about administering user tests and walkthroughs of prototypes. This project is very close to my major in CS applied data science. I happen to be taking App Development and Data Visualization classes during this project timeframe which helped me mix together techniques and learn efficient ways of working together in a team. Assigning tasks and making sure people know what portions of work they are responsible for is very important when working in a team so that people don't step on each other's toes and that work is not overlapped.

  If I were to restart this project, I would probably be tighter on who does what. As we were a small team, we had our individual roles but more or less assisted in all aspects of the project. That is to say, our project did work smoothly and would not really have needed changing. It would have been good to be able to shadow, interview, or otherwise work with a senior team that has made and tested prototypes so we could have a better understanding of how such work is carried out and what little pitfalls we could have avoided for the future teams picking this project up.

Benjamin Hutkoff:

  What technical information did you learn? Over the course of nine months I have accumulated a number of technical skills. First, this project required a large amount of research before any design decisions could be made. Because of this I learned to identify positive heuristic qualities for user apps as well as how to design prototypes to fit those qualities. This includes making low, medium, and high fidelity prototypes all with the users best interests in mind. Second, in making the high fidelity prototypes I had to become accustomed to using and developing within Figma. While I had been aware of the existence of Figma beforehand, I had never personally worked with it, which is why I am thankful that this project gave me the opportunity to explore.

  What non-technical information did you learn? Additionally, because our team used a multitude of communication tools (Discord, Asana, Teams, etc.) I was able to practice my communication skills and worked on maintaining the stream of communication between teammates and our project partner. This is by far the most important skill to have, as a good stream of communication can help maintain work schedules, avoids project confusion, and sets a professional tone for the team and partner.

  What have you learned about project work? I have learned that project work in a group can become quite the hassle when everyone wants to be a part of everything. While it is much better to have a team that wants to work on each aspect of the project then not, it is sometimes difficult to find a job for every person (especailly one that makes them feel valued and that the time they spent doing it is worthwhile). With this in mind, it is important to be clear what roles are available for people and to give them time to understand each roles purpose before handing off tasks. This allows each person time to speak up about a particular role that they might want swapped as well as time to understand exactly what it is that they are undertaking.

  What have you learned about project management? From a professional standpoint, I think that this project failed to effectively communicate its progress to the project partner on a constant basis. While our project partner considered our communication to be adequate, they were consistently out of the loop for a majority of the project (either expecting us to contact them or contacting us directly to ask about showcasing the groups work). From this, I have learned how necessary it is that communication between developers and clients be maintained. If the client feels like they have fallen out of the loop then it is likely that they will feel uneasy about the projects current progress and may even assume that no work has been done since the last communication.

  What have you learned about working in teams? Similar to the paragraph about project work, it is important that people find the work they are doing to be meaningful and "worth it". If someone feels like the work they are doing is not fit for them or their time, then it is much more difficult to deliver upon.

  If you could do it all over, what would you do differently? I think that with the time we had and the speed at which our project was able to finish, that our group would have been prepared to develop a working prototype with functionality. I would even go as far as to say that an application could have been developed that

could be showcased on an actual mobile device. However, I do recognize that app development was somewhat new to all of the group members and that my assumptions of what could have been done might just be wishful thinking.

Erick Branner:

What technical information did you learn? I learned how to use Figma templates. Figma's templates make the prototype designing experience less time-consuming by allowing for quick changes to some element type that propagates to the rest of the elements existing in the prototype, that type. Due to taking a mobile app development class during the year, I was able to learn Kotlin and Android studio. Specifically, I applied my knowledge from this app development class to implement part of our prototyped app.

What non-technical information did you learn? I learned more about my communication skills with customers (project partner). I have already had this background, but I learned more throughout the year to get better and communicate with customers. A consistent meeting schedule is essential for keeping the customer updated with what you're working on. This also allows the customer to identify any changes they would like to make. I learned that you should set meeting times far in advance – if you end up not needing the time, you may just cancel. I learned that sometimes it's OK to not have a one-hundred-percent set schedule for every single task of the project – I tend to want a plan for everything I do. A lot of this information overlaps with *what have you learned about project management.* **Something I've learned about project management** is that it is essential to have the team consistently communicating. Related to this, **something I've learned about working in teams** is that it is critical to know what others on the team are working on. This is critical for preventing duplicated work, and helps with overall awareness. Many of the things previously mentioned contribute to what I've learned about project work**. Something I've learned about project work** is that including the entire team's voice – including the customer and their team – is necessary for creating a quality product. Each person has their own unique perspectives and experiences – these will manifest in the product.

If you could do it all over, what would you do differently? If I could repeat this project from the start, I would have the project management be more concrete. A lot of the project, I was not sure who was working on what, and what I was supposed to do. The sprint schedules were *there* for the spirit of the course, but were not strictly followed. Having worked in industry with a team that used scrum and sprints, I recognize the extreme potential agile methodologies have in producing quality products quickly and efficiently. A clear distribution of well-defined tasks may make a team perform at a higher velocity.

Chitali Buge:

I learned how to use Figma and how to create prototypes. There are many shortcuts and tools that I learned on Figma that I did not know before. It was also interesting to think from a user standpoint and make sure that everything is easy to understand and intuitive to use. As someone who is familiar with the application, there are a lot of instances where I thought a button was easy to find, but actually it was not intuitive at all. Additionally, I learned how to use Qualtrics to create surveys. I did not realize that one can put conditional questions on qualtrics, so it was fun learning how to do that. I also did not know that they had powerful data analysis, which I like a lot better than what they have for Google Forms that I use normally.

There are also many different things that I learned from a non-technical standpoint as well. In terms of project work and working in teams, I learned the importance of communication. Most of our communication was over Discord and sometimes I did not see the messages until later, so I am getting better at checking my phone for messages. Also, communication with the project partner is important as well. While we updated our project partner at least twice a term, we learned that our project partner would like more communication and information. Hence, it was important to check in and see if there is any more information or communication that they would like so we could keep them in the loop the whole time. This leads to one thing that I learned about project management, which is checking in with the clients to make sure that the project meets their expectations and requirements. Anothering thing that I learned about project management is creating and using gantt charts. I had never heard about gantt charts before, and now I have grown to love them. It helps clearly divide sprints and make sure that we are on track. This was really helpful and kept our group on time to complete our project. Additionally, I learned the importance of creating outlines. In other teams, I never had the idea of creating an outline for an assignment before meeting to work on it. I found that it really helped speed up the process and it was really nice to see if we are on the same page since we can start working on it immediately. This is something that I will take to my next group project.

If I could do this all over again, I would try to restructure some of the sprints in the gantt chart. There were some sprints that took longer than others and we kept each sprint for three weeks. There were some sprints that should have been two weeks and others four weeks. Additionally, I would also try to incorporate more user testing for the admin web final prototype. However, the web version was something that was added later due to user feedback, so we did not have extra time to test it out. Last but not least, it would have been nice to also include a last sprint in the gantt chart to accommodate any new features that needed to be added last minute or anything else that came up last minute.