* Briefly summarize the DriverPass project. Who was the client? What type of system did they want you to design?
* What did you do particularly well?
* If you could choose one part of your work on these documents to revise, what would you pick? How would you improve it?
* How did you interpret the user’s needs and implement them into your system design? Why is it so important to consider the user’s needs when designing?
* How do you approach designing software? What techniques or strategies would you use in the future to analyze and design a system?

The Driver pass project was a system that allowed a driving school to create driving and learning packages for students to purchase and learn from. The system is supposed to offer driving packages as well as track the progress of students and give them the option to schedule appointments.

I feel like I have a strong grasp of what the client wanted and was able to interpret the interview transcript very well. I feel that my UML diagram strength is very weak and could use more improvement. I struggle to convey my thoughts and ideas via diagrams.

The way I ended up interpreting the needs of the user was by reading the interview transcript completely once and writing down the system requirements that stuck out to me. Once I had written everything I could remember from my first read, I re-read the transcript and anything that I missed, I added as I read. It’s extremely important to consider the user’s needs for system design because they are the ones that are going to be using that system and in this case they are paying for it. The creation of a system is to simplify the user’s life and if the user’s needs aren’t met, the developer had failed.

The system is being designed from the ground up to meet the requirements of the user. Even if one simple thing is missed, it could set a developer back quite a bit. That is why it is extremely important to pay attention to the needs of the users and make sure everything they need is accounted for. That is why Agile development is so cool because the product owner and the scrum master work together to make sure those needs are met.

The way I approach creating software would be to write out the features I want it to have and then start development. I need to work on getting more comfortable with UML diagrams and I think that it would help me grow as a developer as well as help my peers or future developers work on those same systems. It is critical that I use UML diagrams and comment my code with future projects in my programming career.