

The client for the project was DriverPass, represented by Liam. They were trying to solve the issue of low pass rates within the community. They believed that DriverPass could eliminate the issues of unsatisfactory training. They wanted us to create a website for training drivers, with the ability to also schedule test and buy packages so they have a better chance of passing.

I think what I did particularly well in is finding out what the client actually needed and what their customer would require. I think that was the easiest part is understanding what the client wanted as they were very matter of fact when they conversed with my leads Sam and Jennifer. It was all easy to map out in my head what the website needed to do and how it would be mapped out.

I would revise the portion of project two, where I created the class diagram. That was the weakest portion of my project I believe. I would of added more classes and specified what each classes did. I would of also added more information on what each class did to better define what it could do, such as the class Customer. I could of separated the customerInfo into customerName, customerAddress, customerCardInfo. I could of essentially made it easier on myself if I added more classes, with more through information.

I interpreted the customer users needs into two main groups, study material and scheduling, which the DriverPass company offers. The first one study study material derives from the packages and have three different classes of study materials depending on how they learn. The scheduling portion would then go through a series of options depending on your preference of instructor and vehicle. Admin users would be updating those study materials and making sure the I information up to date, while allowing them to see their customers progress. It's important to consider the users needs when designing so appropriate roles can be set to them. If the customer could edit the websites information, it would be a problem, at the same time the Admin shouldn't be able to take on IT roles either.

I approach software designing by first looking at the functional requirements. They would layout a map or a baseline what needed to be done. It essential shows what's required of the system and allows me to interpret who needs to be connected where. Then I would look at the non-functional systems afterwards to see what the customer would expect of my program, such as how fast the program goes, or what would make the system easier. Future strategies I would use in working with a team would essentially be to look at how they do things and there perspectives. As a young coder I'm inexperienced on how they traditionally do things, and looking at the proper way they do things may help me become more consistent, and help me understand what my clients may need, even if they didn't ask for it in the first place.