For this final project, our overall app was a scheme interpreter forum, where users can code scheme code and use a forum to ask related questions about their own code or otherwise. While my partner coded the main scheme interpreter and made it work, I made the forum and included multiple features inside the forum. These features include an account feature where users can have their own profiles, and of course, allowing users to post their own posts and replies. With this, I also created the main database for the project which was able to store user's information, their posts, and their replies. This was able to make our project more cohesive and suit the requirements.

Some challenges I had was making the database work and allowing users to create new posts, users, or replies with forms. This is due to my unfamiliarity with database management and coding with it through SQL-Alchemy. With us learning database management all the way back in Term 1, it was predictable that I was going to forget most of how to manage data right now. Hence, even right before I started coding my part, I had to relook the slides and notes from Term 1 about databases and how to create the optimum tables and allow for good database management. Only through this, was I able to recap how to manage databases which allowed me to create my own database and planned all the users and tables with an ER diagram.

Furthermore, not just knowing how a database works, but also coding with databases. With SQL-Alchemy, I had to learn with Mrs Neo's notes very closely and understand all the codes so that I could successfully code my project. Even after following her code and making the to do list in the notes, I still could not understand most parts of the code and had to take some time off to comprehend and make sure that I fully took in how to manage databases. Only with that, was I able to code databases with my understanding of databases. Even through this process, I still encountered many coding problems with creating users and had to debugging many times with checking users in tables, checking the syntax of the code, managing through many different py and html files, which was one of the most stressful experiences I had debugging before.

Another challenge I had with my project was that I didn't have much time to include the features that I wanted to include. Originally, our group wanted a feature where we could reply to other replies, or they can like or dislike posts, but this became unfeasible as we started the project very late and had no time to include what we wanted to include. This procrastination caused us to need to sacrifice some parts of the project and had to make us think, what is important to our project. Hence, these were the challenges that our group faced when coding our project.

One thing I learnt in the process was how to code and manage different files and aspects of the project. With the usage of different types of files meant for different functions, such as

views.py for managing the app routes and models.py for defining the database tables and diagrams, such as User or Posts in my project. Although this made it harder to debug, it made it a lot easier to manage through my code and finding and deleting only what I need to find or delete. This also allowed my code to be more cohesive and clearer to those who read my code. Furthermore, the use of multiple languages across different files also increased my multitasking abilities and taught me how to manage different things at once.

Another thing I learnt was how to work together with other people in programming projects. With the lack of multi-usage ips, where users can edit at the same time, I had to learn different ways to send my partner code so he can edit it and we can work on one project. This allowed us to come to the conclusion to split our parts up and send each other our code through Github by pushing it or send it to each other via Whatsapp, hence reducing its complexity.

One aspect of the project I am proud of is the database. As I made the whole database myself, I could appreciate how it could manage so much data by itself and allow it to group it to different groups and make the program work. Although the database was compulsory, I am still proud of myself for adhering to the notes given and following instructions so that I could make a suitable database and manage and print everything out concurrently.