**EXPERIENCE**

This project ended up being much longer and more difficult than it originally looked. Each individual part wasn’t necessarily difficult, but when you try to stack featured on top of each other, and moreover with multiple programmers, it becomes very difficult to make each feature work with others. Obviously the essential use cases specified in the project description were the ones we mainly focused on. The “going farther” extra use cases were a fringe goal of ours when we started to run into other troubles, though we did manage to get a couple of those use cases in, like adding someone to and removing someone from a mailing list.

We initially ended up breaking the project between our three members into the front-end, back-end, and database. This model didn’t last too long, as the front-end member built the pages quickly so he was able to move onto the back-end portion. I think this is a good approach to take, though the front-end building is definitely a bottleneck as the front-end communicates directly with the back-end via bean bindings.

One of the most valuable things I think we learned was how to code with other people. When you are working on your own small/medium-sized assignment, you know exactly what you have done, where things are, and what needs to be done. This is obviously not the case when working with others. It is of utmost importance to keep communicating with each other to make sure that two people don’t have conflicting ideas about a project, or aren’t working on the same feature and stepping on toes. This actually happened with our group early on.

Another challenge and learning experience that we encountered was the challenge of sharing, using, and combining code with each other. We ended up using Git, though we did have some challenged with corrupted JAR files and other merge issues. We all had to independently learn the git language, and how to use this with GitHub. This ended up taking a significant amount of time and energy since really none of us have had any experience with this issue. By the end, we were semi-effectively sharing code with each other, however.

We found PrimeFaces to be very helpful in some situations, but when we began trying to use it with other features, the lack of documentation/experience was noticeable. We were already learning how to use Git as well as SQL, so our PrimeFaces use was fairly limited as we wanted to focus on that core functionality. I was able to implement some PrimeFaces in our project, however.

Ultimately, however, we got most of our core functionality in that we wanted. I think the biggest thing that our group will take away from this class is the MVC/DAO models and how they fit into development. Personally I have a hard time thinking about projects any other way because the MVC model fits so well, as does the DAO. We had a little bit of trouble with scoping variables and using them in the DAO, but we worked around it OK. All in all we had a difficult time scoping information/variables since we didn’t really do much with it in classes and we were running low on time. I’m happy with our results as we worked very hard and learned a *lot* on our own for this project.