**Group Experiences**

This project has certainly been a challenging one, to say the least, especially when considering we’re essentially a start-up company with no prior experience that’s learning along the way. It’s been rough trying to meet the deadlines when you consider that each group member has other classes to deal with, a job, and various familial responsibilities; collaboration isn’t always the easiest thing. Luckily, though, each member has been more than willing to do their fair share of work, and then some, to ensure that this project goes as smoothly as possible.

We’ve each been more-or-less on the same page when it comes to the direction we want to take the project. That doesn’t mean, however, that we haven’t spent hours debating over everything from whether or not a class should exist to the semantics of a single relationship. In those regards, we’ve had few issues, but we have made some interesting discoveries as we continue on our journey of analysis, all of which have had their subtle impacts on where our project is heading. Most of our design changes have come as we near the implementation stage and have mostly just involved the addition of a class here and there that will smooth out the implementation process and provide and bit more abstraction from the users.

To us, the term “group leader” is a bit of a superficial one, and we prefer to use our cohesive teamwork to drive ourselves forward rather than relying on a single person to divvy out work. We’ve made extra sure everyone pulls their own weight, but we each do so in our own ways. As you’ve covered in your lessons, we’re a team, but each member has his own strengths and weaknesses. These are things we’ve been very mindful of as we progress. We each know what we’re good at, and what we’re not so good at. During our meetings, when deciding how the work should be assigned, we each offer to take a certain amount of work based on what we believe we can handle and what we each believe we are the most proficient at, that way the work is done as quickly and as best as possible. We each then share our work through its various stages with one another to get feedback.

**Final Report Walkthrough**

Doxygen: contains all the auto generated html documentation.

-html: contains all the individual class documentation

-UCCS Doxygen Comments: A shortcut that brings you directly to the documented list of classes

Source Code: Contains all the class templates

ClassDiagram v1.vsdx: The finalized diagram of all of our classes

Extended Use Cases v3.docx: Expands each use case we included in our use case diagram

Use Case Diagram v2.vsdx: The finalized diagram containing all of the use cases