Goals:

* To try your hand at writing/assembling a Software Requirements Specification (SRS) document
* To have a portfolio piece that demonstrates your ability to produce a software lifecycle-related document
* To have practical practice dealing with both functional and non-functional requirements
* To understand the relationship between requirements, design, and implementation
* To practice the role of the business analyst / requirements analyst / systems analyst / functional analyst

Overview:

In this assignment, you will be producing a Software Requirements Specification (SRS) document for a software system (or a portion of a software system) that you already know. All of you are currently working on a capstone project and/or completed a project in your Agile Development Methods class. Typically in requirements engineering, one would take a significant amount of time and effort to elicit requirements from a user. But in an effort to shortcut a bit, we will leverage all of your previous conversations with your client(s) as you have already done requirements elicitation. What's left to do is translate those business-related conversations into a specification that can be given to a technical team to build the product.

Create a Software Requirements Specification (SRS), using one of the models/examples that were provided in class (the traditional "shall statement" model or the more informal "use case" based requirements model). The SRS should be complete enough where you would be able to hand it to another developer (contractor) to build the software product (or component) that your team is building. You have the advantage of knowing what should be built, but your challenge is to make sure it is expressed in the SRS.

Some projects are larger than others. You may choose to write an SRS on one component of your project if your project is large. I recommend checking with your instructor to make sure the scope of the requirements document is appropriate.