

## CSE 403 – Winter 2011 Assignment 2 - SRS

The goal of this assignment is to establish a solid definition of your project from which to base a design and implementation.

**Due date: Friday January 21<sup>st</sup>, 11:00 pm**

### Overview

Congratulations! Your team has been funded to produce the software product outlined in your earlier proposal. The "customer" hiring you to complete the project is a conglomeration, CSRocks Inc., which includes the executives (Alverson, Brun, Soren and Lu) and upper-level managers (another team of the class), who will meet with you periodically to discuss and evaluate your progress.

Now, you need to define the project, as a basis for your later design and implementation. Your first deliverable is a set of requirements documents (sometimes called "Software Requirements Specification" or SRS). These describe the goals of your project and how users will interact with it (the high-level UI design). You will also document your plans for completing the project.

### External Requirements

While the customer likes the high level features outlined in your proposal, they are happy to leave the next level of refinement to you. They do, however, have the following requests:

- The product should be as usable as possible, even for people who are not expert computer users (with the exception of projects that are designed specifically for experts, such as development tools).
- The product must be robust against errors that can reasonably be expected to occur, such as invalid user input, lost network connections, etc.
- As before, the product must be a networked (client-server) application.
- The product must be installable by a user, or if the product is a web-based service, the server must have a public URL that others can use to access it. If the product is a stand-alone application, you will be expected to provide a reasonable means for others to install and run it. You can expect that the user will have installed any necessary libraries and tools (such as a Java Virtual Machine, or a Ruby framework runtime, or a simulator for a mobile device), but after that, the user should be able to download and run your system easily.
- The product ultimately be made open source. As a step towards this, the software (all parts, including client and server) should be buildable from source and installable by others, and well documented to enable new developers to make enhancements. Documentation will include design documents, test cases, and bug reports.

- The scope of the project is of good size for the resources assigned.

Beyond these requests, you are largely free to take the next turn of the product development spiral and firm up your product requirements. This requirements document will essentially be a contract with CSRocks for what you plan to deliver. Consequently, you should talk to your customer as you plan in order to make sure your product meets their needs.

### **Software Requirements Specification (SRS)**

A Software Requirements Spec is a tool for capturing requirements of a product. We have placed a link to the SRS template to use for this assignment on the class web site.

The main product components to capture in the 403 SRS are:

1. High level description of the product, along with its scope.
2. Set of use cases, including at least one use case summary diagram and two formal use cases of the most important scenarios.
3. Feature list, including what features you expect to be available for the alpha and beta releases as well as the final 1.0 release. You should also specify a set of “stretch” features that could slip to 2.0 if necessary.
4. UI prototype. At least two diagrams containing rough sketches of your product's user interface.

This document will be a living document. You will be asked to provide updates to it at periodic points in the development cycle.

### **Deliverables and Grading**

1. An SRS document.

A requirements document based on both the description in this assignment and the SRS template on the class web site. While some SRS are hundreds of pages long, yours should be 4-5 pages (not including the use case and UI diagrams) at most, given the quarter scope of the project and the iterative development process we are taking.

2. Customer discussion artifact.

Records showing that you had at least two discussions with your customer about the product (one with the execs (course staff) and one with the upper managers (other class team)) about the product requirements. Example artifacts include meeting minutes, email threads, and paper prototyping. It should be clear how these discussions shaped your requirements. These records can be submitted in hard copy if they are not easily available online.

Please have one person from your group submit your deliverables so that both files will be stored in the same place. The preferred names for these files are

*YourProjectName\_SRS* [doc|pdf], *YourProjectName\_CustomerArtifacts*.[text|pdf].

Please have all group member names visible in each file. Use the turnin tool on the class web page: <https://catalyst.uw.edu/collectit/dropbox/alverson/13423>. **This**

**assignment is due before 11:00 PM, Friday January 21<sup>st</sup>, 2011.**