CSCE 740 - Project 1 - Draft Requirements

Due Date: Wednesday, September 27th, 11:59 PM (in PDF format, via Dropbox)

Problem Statement

Ancient bearded professors (who are you staring at?) may philosophize and ruminate on the value of education. However, as a student, you know the value - it's what you paid at the beginning of the semester! It is important that students be able to see their current bill, understand what they owe, and be able to pay their bill (at once, or in increments).

Your new software development firm has been contacted by the University of South Carolina to build such a system. This system will serve as a back-end, offering an API that is accessible by a variety of other systems (and user interfaces). This back-end API will offer services such as viewing and editing of student profiles, viewing of bills, and payment of bills.

Overall Project Description - the Billing Interface Linkable Library (BILL)

Your mission, should you choose not to fail the assignment(s), is to create a student billing system, known as BILL. Here you will get a set of informal "user requests" of what this system shall do. Note that this is purposely incomplete and vague, and that it is your job to sort it all out (i.e., by eliciting requirements). Other features may emerge as part of the online elicitation session.

1. The system shall offer an API interface, compatible with a variety of calling systems. It should be intuitive and easy to use.
2. The system shall allow users to view their current bill.
3. The system shall allow users to pay all or part of their current balance.
4. The system shall allow users to view their profile information.
5. The system shall allow users to view a detailed list of past charges and payments, within a selectable range.
6. The system shall allow users to edit their profile information, within limits.

Later in the semester, you will be given further interface guidelines, the database abstractions (data that can be loaded into your system), and a description of the inputs to the system.

A listing of fees is located here: http://sc.edu/bursar/fees.shtml
Note that not all of these are required to be supported.

Your Task

Your task in this assignment, as outlined above, is to develop use-cases to help you elicit the requirements, and then to develop a requirements document for BILL. You may choose any organization of your document, but always keep in mind that is must be readable, changeable, and capture all the essential information we have discussed in class. The same applies to your requirements - your requirements must be detailed, consistent, complete, and testable. Remember, you will build this system later, and if you have a poor requirements document, you will run into problems. You will be graded on what the system ought to do (as opposed to what you think it should do).

A requirements document template will be made available on the web page for the class. In addition, a template for individual requirements and examples will also be made available on this page. You may adjust the templates to fit your needs, as long as necessary information is delivered clearly. There are also some examples of use-case diagrams and scenarios in the slides and a use case template will be available on the same page. Finally, several checklists will be made available that you can use to determine if your requirements are up to professional standards.

The Requirements Elicitation

On Thursday, September 14, a digital elicitation session will be opened on the Dropbox site. This will be your opportunity to ask questions directly to the “customers” and formulate your requirements. Before this date, be sure to come up with questions to ask (and start coming up with requirements).

Helpful Hints

Do not invent many unneeded requirements. Focus on the core functionality of BILL and do not add “things that would be nice to have”. “Gold plating” the requirements by adding all kinds of nice, but unneeded, functionality and checks will lead to an excessively large and complex document.

Focus on the features asked for in this document and in the elicitation session. You must write detailed requirement specifications that capture what the behavior of the final system should be, including defining functionality, specifying properties that the output of the system must follow to be considered correct, and posing non-functional requirements that govern how the system operates with regard to performance, security, etc.

Deliverables

You are required to turn in the requirements document, your use-cases, and a use-case diagram. You can put your use-cases and diagram as a chapter in your requirements document, or you can submit it as a separate document. All documents must be in PDF format.

Note that the first deliverable is worth a relatively small portion of your grade, and is intended for you to get feedback so you can prepare suitable document for the second deliverable. That said, take this assignment seriously, ensure you have time to complete the assignment, and focus on demonstrating that you can write good requirements.