Lab 2. Requirements Document.

Due Date: 30 January, 2002

Please complete this assignment with your project group members.

In this assignment, you will define the requirements of a restaurant reservations and table assignment software system. This assignment will form part of the course project.

Audience: Write for both junior software engineers such as yourselves *and* any potential customer of your software. This document will be taken seriously by any customer and should be written with great care.

Main Point and Purpose: The purpose of this assignment is for you to identify the requirements of the software you will be building later in the quarter. Requirements analysis is the first step in our iterative, risk-reducing software engineering process.

Procedures:

1. Read Topic 36 in the textbook, pp. 202-211. Please do this as soon as possible, so that you have ample opportunity to get any questions answered by the instructor.
2. Did you read Topic 36 yet? Don't even think about looking at the rest of this assignment until you have done the reading. Go read Topic 36.
3. Arrange the Requirements Document as a set of web pages in HTML. You can use an HTML generator such as Microsoft FrontPage or write your own HTML. Please do make ample use of hyperlinks – they are a good way to avoid duplication in your documents. Host this document somewhere in your personal or CSE web space. If your group cannot host the document anywhere, contact the instructor.
4. Prepare the document as follows:
   (a) Introduce your system by describing exactly what it will do. The reader of this introduction should fully comprehend what this software is intended to do, although they may not understand the exact details of how a user would operate the software. Please be thorough.
   (b) Describe the scenarios of use associated with this software. For each scenario, list the individual actions that comprise the scenario.
   (c) For each individual action listed in the scenarios, define a *use case*.
   Use the template in Figure 7.1 of the textbook to structure each use case. Use Figure 7.2 as an example. Follow the template exactly, with the following exceptions:
      - You should omit the following items from the use cases you write – Scope, Level, Superordinate Use Case, Subordinate Use Case, and Schedule. All other items listed in Figure 7.1 should be addressed unless they are not relevant to your specific use case, in which case you should write "Not Relevant" after the item heading.
      - You can add items not listed in Figure 7.1 to the use case if you think they are helpful and do not duplicate any of the other items.
   (d) Define any *non-functional requirements* that exist for your software as a whole.
   (e) Identify *useful subsets* of your software. The Scenarios and Use Cases should describe the most full-featured software that you can envision completing this quarter. In this section, describe viable subsets of this software system. Make a list starting with a software system that does

almost everything you want and end the list with a software system that does the bare minimum needed to still be remotely useful. Fill in the list with every sensible choice in between. Remember, a subset has to be useful, it can't just be any random piece of the system.

(f) Include a Project Glossary. You may want to italicize words whose definitions appear in the glossary; even better, you may want to hyperlink definitions to appearances of words throughout the document. Design the glossary so that it is easy to use.

Standards and Criteria: In completing this assignment, demonstrate that you can clearly identify the requirements of the software system you are to implement for your course project. Clarity matters – vague wording can be interpreted in different ways and will lead to misunderstandings among your group members and potentially between you and your customers.

We will expect your Requirements Document to serve as your guide throughout the software engineering process we are undertaking this quarter. This assignment is not a stand-alone entity – you (and the graders) will be referring to the document throughout the quarter. Thus, do not include functionality that you have no intention of implementing. Rather, concentrate on clearly identifying the functional and non-functional requirements that you do intend to complete. We understand that you may not be able to properly estimate what you can and cannot complete by the end of the quarter. Just try your best.

Place a hard copy of your Requirements Document in each of your binders. Turn in a hard copy of the Requirements Document as well. Only one submission per group is needed. Before class begins on Wednesday, 30 January, 2002, email Konrad (konrad@cs) and send him the URL of the web version of your Requirements Document. Links to all Requirements Documents will be published on the 403 web page.

Extra Credit: The following activity is very **optional**.

Visit a large restaurant and find out how they handle reservations and table assignment. You must visit a **large** restaurant – examples include The Cheesecake Factory, Dimitriou's Jazz Alley, and The Old Spaghetti Factory. Prepare a web page containing the information that you find and send the URL to Konrad.