Lab 3. High-Level Design Document.

Due Date: 13 February, 2002

Please complete this assignment with your project group members.

In this assignment, you will outline the high-level design of a restaurant reservations and table assignment software system. This assignment will form part of the course project.

Audience: Write for junior software engineers such as yourselves. Not all readers will be familiar with your project.

Main Point and Purpose: The purpose of this assignment is for you to demonstrate that you understand high-level design and can complete the high-level design of a software system.

Procedures:

1. This document does not have to be written in HTML, but instead can be written in whatever format your group chooses. Structure the document as follows:

   (a) Introduce the high-level design of your system with a brief review of what the software is supposed to do and a very short description of the major components of your high-level design.

   (b) Describe each component of your system. The purpose of each component should be well defined. List which modules communicate with each other, and explain why communication is necessary. If your architecture follows a common architecture pattern, or is a variant on an architecture pattern, describe the pattern that your architecture follows or is based on and describe how the pattern is instantiated in your specific design. (If your architecture is a variant, explain what diverges from the pattern and why.) Briefly describe alternative architectures and then explain why the one you chose is best.

   You are highly encouraged to use diagrams. If your diagrams are complicated, you may want to include a legend that explains what the various shapes and lines of your diagrams mean.

   (c) Outline your change strategy. This may include the use of metadata to remove details from the code. This may also include properties of your chosen architecture that ensure that certain changes are easy to do. List likely changes and how you have planned for them.

   (d) Outline your system topology. Indicate how stable this topology is (in other words, how likely is it that this topology might change?). Is this topology superior to all other possible alternatives? If yes, explain why. If no, describe the pros and cons of some alternatives and explain why you chose the one you did.

   (e) Describe any software that you'll be using that you didn't write yourselves. Explain why you are using this software. List some alternatives and

   - explain why the software you have chosen is the best for your system – OR
   - describe how this software is just one option you have and what other options you may use instead should any troubles with use of this software occur. Explain the pros and cons of some of the options.

2. Throughout this assignment, you are asked to compare elements of your high-level design to other possibilities and perhaps explain why your choice is best. Please make these comparisons on the basis of software engineering criteria – for example, is your choice more robust? easier to understand? more

accommodating to likely changes? more orthogonal? avoid duplication? remove details from the code? Is your choice a better fit for the functional requirements? Non-functional requirements? Is your choice the best one given your resource (time, money, and computing power) constraints?

If you think your high-level design, or parts of your high-level design, are only one among a number of possibly good choices, say that and explain why you think so. If you think your high-level design, or parts of your high-level design, have disadvantages, but the advantages outweigh the disadvantages, describe what you think the disadvantages and advantages are and why you think this way. In other words, don't feel like you have to show that your plan is the best and all others are scum. Just show that you have considered other ideas and have based your decision-making on software engineering criteria.

3. Decision-making on a scale as grand as the high-level design should be a group activity. Group members can be assigned specific issues to research or specific parts of the writeup, but decisions should be made by the entire group. Be sure to plan enough meeting time to make good decisions.

Standards and Criteria: In completing this assignment, demonstrate that you can clearly define a high-level design that is internally consistent and meets the needs of your requirements. You will be expected to provide a quality document that is current with respect to what your existing software looks like and your future plans. You should demonstrate your ability to make decisions based on software engineering criteria.

Place a hard copy of your High-Level Design Document in each of your binders. Turn in a hard copy of the High-Level Design Document as well. Only one submission per group is needed.