6.033 Computer System Engineering

Spring 2009

6.033 2009 Design Project 2

I. Due dates

You will do Design Project Two in teams of three students who share the same recitation instructor. There are four deliverables:

1. A list of team members emailed to your recitation instructor by Recitation 17.
2. One copy of a **design proposal** not exceeding 1,200 words, due by Recitation 20.
3. One copy of a **design report** not exceeding 5,000 words, due by Recitation 24.
4. An **in-class presentation** in Recitation 25. In consultation with the chair of the faculty we have determined that the assignment follows the spirit of the end-of-term rules.

This assignment is under-specified, and part of your job is to complete the specification sensibly, given the project requirements. Your interpretation of the specifications may need some adjustment as you flesh out your design. Start early to give yourself time to think about your design and revise it. A good design will likely take more than a few days.

II. Introduction

You and your team are the founders of a new start-up company, called Chatter. The basic idea is to allow users to distribute short text messages to each other using hand-held wireless devices.

A typical use scenario is a stadium full of sports fans exchanging comments about the game. Another is students attending a lecture, exchanging questions or feedback. Students might also use Chatter to help organize their social lives, to find nearby friends at meal-times or to find out about campus events. In principle all users should be able to see all messages, though in practice your design will need to restrict distribution to conform with the requirements in the next section.

III. System requirements

You and your investors agree that Chatter must meet the following requirements:

- It should take the form of software running on a hand-held PDA-like device called the ChatterBox. The ChatterBox has a large touch-sensitive screen, a small keyboard, a WiFi wireless network interface, and a GPS receiver. The ChatterBox is not a cell phone.
- Users should see each message within seconds of when it is sent, if possible.
- Chatter should work reasonably well at extremes of scale: in a stadium with tens of thousands of users, as well as in much sparser environments.
- Chatter should provide users with tools to restrict which messages they see, perhaps seeing only messages from particular individuals, or from friends, or on particular topics, or from particular geographic areas.
- The ChatterBox's WiFi radio has a range of only about 50 feet.

- If a ChatterBox is in range of a WiFi access point that is on the Internet, the ChatterBox may use TCP/IP to talk to servers on the Internet. The access point will allocate it a temporary IP address for this communication. A ChatterBox does not have a permanent IP address.
- Chatter should be as functional as possible even when a ChatterBox is not in range of a WiFi access point. The ChatterBox WiFi interface allows sending of "ad-hoc" packets to nearby ChatterBoxes, as if they were on an Ethernet. These ad-hoc packets are link-layer broadcasts; an ad-hoc transmission is received by the ChatterBoxes within radio range.
- If a ChatterBox is not in range of a WiFi access point, it will not have an IP address. This means that your design must describe all protocols above the link layer for any ad-hoc communication that Chatter uses. Each ChatterBox has a unique 128-bit serial number accessible to the software.
- It will probably be too hard to ensure that every ChatterBox sees every message; it is acceptable if your design does not guarantee delivery. For example, if there is no path in the underlying network from one ChatterBox to another (no path through other ChatterBoxes and/or the Internet), it is acceptable for Chatter to not deliver messages from one to the other, or to delay delivery.
- You can assume that each user has exactly one ChatterBox and uses only that ChatterBox.
- In addition to supporting distribution of messages to many users, one user should be able to send a message to a specific single other user.
- Chatter should try to present messages to the user in a sensible order. For example, one possible scheme might have the following behavior: suppose user U1 sends a message M1. User U2 reads M1 and then sends M2. If user U3's ChatterBox receives M2 first, it will display M2 to the user immediately. If it then receives M1, it will indicate to the user that M1 was sent before M2.
- It is fine for Chatter to use one or more servers on the Internet, though the design must cope with the likelihood that many ChatterBoxes will not be in range of a WiFi access point.

Your design need not include security mechanisms.

IV. Your job

Your job is to design the Chatter system, focusing on the algorithms and network protocols that the ChatterBoxes use to fulfill the requirements in Section III. If your design includes servers on the Internet, then you should also design the services those servers provide. Your design should also include any aspects of the Chatter user interface on the ChatterBox that are important in fulfilling the requirements.

Here are some questions that your design should answer (and that we will expect to read about in your report):

- What happens when a user sends a message from his/her ChatterBox?
- How does Chatter cause nearby users to see each message? What is your design's definition of "nearby"?
- How does Chatter ensure that messages do not loop?
- How does Chatter cause users to see messages from other users who are not in direct radio range?

- How does Chatter cope with lost packets? With changes in radio connectivity as users move?
- How does Chatter arrange to show messages to the user in a sensible order?
- How does Chatter support a user sending a message to just a specific other user?
- How do Chatter's protocols differ (if at all) in the cases where a ChatterBox is and is not connected to a WiFi access point that's on the Internet?
- How does Chatter allow users to control and limit the messages that they see, to help them cope with large numbers of nearby users? What are the default settings, and why do they make sense?
- How would your design cope with success: what would happen if millions of people spread out over the Internet used it?
- How would your design cope with tens of thousands of people using it in a small area, such as a sports stadium?
- Does anything special have to happen when a new user starts to use Chatter?

It may not be possible to design a system with good performance in all possible scenarios: you may need to make tradeoffs, supporting some kinds of use well, and others not so well.

V. Your design proposal

The design proposal should be a concise summary (1200 words) of your overall system design. Your proposal should explain the basic mechanisms with which ChatterBoxes distribute messages.

You do not have to present a detailed rationale or analysis in your proposal. However, if any of your design decisions are unusual (particularly creative, experimental, or risky) or if you deviate from the requirements, you should explain and justify those decisions in your proposal.

You will receive feedback from your TA in time to adjust your final report.

VI. Your design report

Your report should explain your design for Chatter. It should discuss the major design decisions and tradeoffs you made, and justify your choices. It should discuss any limitations of which you are aware. You should assume that your report is being read by someone who has read this assignment but has not thought carefully about this particular design problem. Give enough detail that your project can be turned over successfully to an implementation team.

Your report should convince the reader that your design satisfies the requirements in Section III, and should tell the reader the answers to the questions in Section IV.

Use this organization for your report:

- Title page: Give your report a title that reflects the subject and scope of your project. Include your names, email address, recitation instructor, section time(s), and the date on the title page.
- No table of contents.

- Introduction: Summarize what your design is intended to achieve, outline the design, explain the major trade-offs and design decisions you have made, and justify those trade-offs and decisions.
- Design: Explain your design. Identify your design's main components, state, algorithms, and protocols. You should sub-divide the design, with corresponding subsections in the text, so that the reader can focus on and understand one piece at a time. Explain why your design makes sense as well as explaining how it works. Use diagrams, pseudo-code, and worked examples as appropriate.
- Analysis: Explain how you expect your design to behave in different scenarios. What scenarios might pose problems for throughput, latency, or even correctness? What do you expect to be the scalability limits of your design?
- Conclusion: Briefly summarize your design and provide recommendations for further actions and a list of any problems that must be resolved before the design can be implemented.
- Acknowledgments and references: Give credit to individuals whom you consulted in developing your design. Provide a list of references at the end using the IEEE citation-sequence system ("IEEE style").
- Word count. Please indicate the word count of your report at the end of the document.

Here are a few tips:

- Use ideas and terms from the course notes when appropriate; this will save you space (you can refer the reader to the relevant section of the notes) and will save the reader some effort.
- Before you explain the solution to any given problem, say what the problem is.
- Before presenting the details of any given design component, ensure that the purpose and requirements of that component are well described.
- It's often valuable to illustrate an idea using an example, but an example is no substitute for a full explanation of the idea.
- You may want to separate the explanation of a component's data structures (or packet formats) from its algorithms.
- Explain all figures, tables, and pseudo-code; explain what is being presented, and what conclusions the reader should draw.

While the Writing Program will not be grading DP2, you should feel free to ask them for help.

VII. How we evaluate your work

Your recitation instructor will assign your report a grade that reflects both the design itself and how well your report presents the design. These are the main high-level grading criteria:

1. **Clarity**. Is the design described well enough to be understood, evaluated, and implemented?
2. **Correctness and completeness**. Does the design meet the requirements in Section III?
3. **Simplicity**. Is the level of complexity in the design justified?