Brown University
Spring 1999
Software System Design

CS 190
Handout #1

Instructor:	Head TA:	TA:
David Laidlaw	Jimmy Ho	My Lan Payson
Office: CIT 449	Office: CIT 402	Office hours: wed 19:30-21:30
Hours: by appt.	Hours: mon 18:00-20:00, thu 15:00-17:00 or by appt.	Office: 2nd floor TA room
Phone: x7647	Office phone: x7681	e-mail: mlp@cs.brown.edu
e-mail: dhl@cs.brown.edu	Home phone: x6037 (not 23:00-09:00)	
	e-mail: jkh@cs.brown.edu	

Course Description CS 190 is a course in which students working in teams implement significant software applications. Students learn and practice techniques of project management, specification, presentation, analysis, design, coding, documentation, testing, and maintenance. This class is one of the last in the undergraduate curriculum and it can tie together many of the earlier classes very practically. In addition to the focus on a group programming project, we will discuss, debate, and think about the group software development process. Groups introduce a non-determinism that can be refreshing but can also make a project more challenging that you might first imagine.

Aims I have three aims for people taking CS190. First, learn enough of the issues about creating software in groups to successfully implement future projects. Second, become one of the team members on future projects to whom others turn for advice on software engineering and development. Third, continue to think clearly about the issues, articulate your conclusions, and adapt to changes.

Objectives When you finish the course, you will have implemented a moderately large software systems as a group. That is the primary objective of the course. Along the way, you will need to determine requirements and specifications for a project in small groups and present it to the class. In a larger group, you'll design the overall projet, design the pieces, implement the pieces, integrate them, and demo the result. Throughout, we'll discuss the problems that crop up and see what we can learn from them. We'll also try out a set of in-class programming experiments to evaluate some different software engineering strategies.

Assessment

- Project success and demo: base for class grade. This will be a group grade.
- Every two weeks during the course of the term each group will get a predicted project grade based on the progress they are making. These reports will be based on the individual log files of each person in the class and on progress meetings held with the staff.
- Each group leader and the TA's will evaluate each group member at the end of the term. This evaluation can affect an individual's grade by 1/2 letter.
- Each handin and presentation: can alter your grade by up to 1/2 letter. A late handin will cost 10% of a letter grade off the final course grade for each class meeting it is late.

- Class participation: up to 1/2 letter grade increase.

Course Content Overview Classes will be divided between lectures, student presentations, design discussions, and group meetings. See the project handout and the course schedule for more details.

Textbooks The required texts are:

1. The Mythical Man-Month: Essays on Software Engineering, by Fredrick P. Brooks, Jr. (Addison-Wesley, 1995).
2. Debugging the Development Process: Practical Strategies for Staying Focused, Hitting Ship Dates, and Building Solid Teams, by Steve Maguire (Microsoft Press, 1994).
3. Large Scale C++ Software Design, by John S. Lakos (Addison-Wesley 1996).

Also recommended:

1. Writing Solid Code, by Steve Maguire (Microsoft Press, 1993).