

Supporting Collaborative Teams in Engineering Education

Francis C. Li
University of California at Berkeley

Abstract

Problem-based learning techniques involving small collaborative teams of students are widely regarded as being effective in teaching engineering design courses. However, the heavy requirements upon the teaching staff to support and coordinate such teams typically limit these courses to small numbers of students. In this paper, we describe the design of a computer-based tool for supporting small collaborative teams, with the ultimate goal of scaling these engineering design courses to larger numbers of students. The system, deployed on laptops with wireless networking, utilizes student self-evaluations and instrumented project management tools to provide feedback of individual and team performance to the students and teaching staff. We present an example of the usage and data collected from this system during pilot tests in undergraduate computer science design courses. Based on that experience, we discuss future plans to refine the system for early detection of problem teams and to instrument asynchronous and instant messaging tools to support social network analysis of team communication.

1. Introduction

Engineering design courses commonly combine traditional lecture delivery with team project assignments, where small collaborative groups of students work on a problem related to the course material. Although this is not problem-based learning as described by Barrows, many of the same benefits arise from the addition of team project work—greater interpersonal communication between students, knowledge sharing, and a degree of self-learning of new material [1, 2]. However, while lecture delivery can scale with increasing enrollment via larger classrooms and video broadcasts, the resources required to facilitate team project work, especially in design courses, often cannot.

Monitoring the progress and performance of individual teams in a course requires significant effort from the instructional staff. Often this involves identifying social loafers and mediating personal differences between students, in addition to evaluating learning outcomes. Usually, the resolution of these problems and conflicts come after they have already manifested itself as degraded team performance. As enrollment increases, however, it is difficult to scale the size of the instructional staff, a scarce resource in many engineering departments.

Increased enrollment also taxes instructional equipment resources, particularly computers. As computers become required tools for more and more engineering courses, traditional laboratory facilities quickly reach limits in both space and time to expand. Students must fight for resources in overcrowded lab, often leaving members of a team unable to work together at the same time and place.

We believe that computer-based tools can be useful in assisting the teaching staff to monitor the progress and performance of teams in large engineering design courses. By deploying these tools along with wirelessly networked laptops for each student, we can also help alleviate the equipment problem, while offering greater freedom and mobility for students in teams to work with each other. This provides an environment more suitable for collaboration in which we can survey and observe student work practices.

In the next section, we describe metrics for assessing team progress and performance. Then, we describe our computer-based system for collecting and presenting those metrics to both students and instructors for awareness. We have performed pilot evaluations that we detail in the next sections, followed by a discussion of the results. Finally, we discuss future plans for the system and conclude.

2. Progress and Performance Metrics

We began our design by choosing a set of individual and team progress and performance metrics. They are:

1. Individual time spent
2. Team time spent in meetings
3. Individual action items completed
4. Team milestones achieved
5. Peer evaluation of team members
6. Private comments to the instructors

Metrics 1 through 4 have been successfully used and described by Collofello and Hart for the monitoring of teams in software engineering courses [3]. The amount of time spent by students and teams is self-reported, and provides a rough estimate of effort and performance. Managing individual action items and team milestones provides awareness of progress as well as encourages what is considered to be “best practices” in software engineering [4].

Informal interviews with students working in teams confirm that they are often insecure about face-to-face confrontations with team members or the instructional staff regarding problems. The peer evaluation of team members provides a very lightweight form of feedback regarding individual contributions to the group. Each member rates their own contribution out of 100 percent, and then distributes exactly 100 percentage points to their other members. The ratings from everyone in the team (excluding the self-rating) are combined in a calculation to provide an aggregate peer rating for each member that can be compared to the self-rating. Table 1 shows an example of the calculation of the aggregate peer ratings. These ratings are commonly used by educators in many different domains, and have been successfully applied to evaluating group writing and team project work [5, 6].

Adding the ability to send private comments to the instructors provides an additional lightweight channel for students to present any concerns they may have regarding their team. Although email also provides a lightweight form of communication, integration with the project management tool provides a more immediate outlet when used in context with the other self-reported metrics.

Table 1. Calculating the aggregate peer ratings for members of a team. Each student distributes 100 points to the other members of the team (shown in the first four columns). All the points assigned for each student are totaled and divided by the total number of points distributed (400 points in this example) to get an aggregate rating showing the percentage effort in the team.

	Andrew's Rating of:	Bob's Rating of:	Carla's Rating of:	Dana's Rating of:	Total	Aggregate Rating
Andrew		55	45	40	140	35 %
Bob	33		30	30	93	23 %
Carla	33	25		30	88	22 %
Dana	34	20	25		79	20 %
Total	100	100	100	100	400	100 %

3. Initial System Overview

Our tool is implemented as a web-based client/server application. Figure 1 shows the student homepage, which is retrieved after logging into the system. The Main Menu section provides links to the individual and team report forms that collect the student-reported data. The Action Items section displays the currently active action items for the logged-in student and the other team members. The student can complete action items or add new action items from this page. Team Milestones are displayed in the upper right corner along with the due date and/or notifications that the milestone deliverables are late or done. Once each team member has submitted his or her individual report, the peer participation rating is calculated and displayed. Finally, all the individual and team hours are summed and displayed, with a comparison to the average and the other teams in the course. Team names are removed from the list for anonymity. Each section has an Add/Edit or View History link that will display a more detailed summary of the metric data and provide additional editing control in a new page.

Figure 2 shows the instructor's course monitor homepage. The instructor homepage summarizes the metric data for all the teams in the course for a particular week. For each team, the homepage shows the low, high, and average individual time spent by the members of that team as well as the total sum of each member's individual time plus time spent in team meetings. The lowest and highest peer ratings over the members of the team are summarized in the next two columns. The number of action items completed over all the members of the team is summed in the "Done" column followed by the total number of currently active (not completed) action items over all the members. The number of completed, late, and currently active team milestones are shown in the last three columns.

By choosing a particular team, the instructor can see team information and summaries of the metric data for each member, as shown in Figure 3. The milestones entered by the team are displayed at the top with their due dates and status. For each member, the individual time spent for the particular week is reported, along with a cumulative total of all the hours reported so far. The member's self rating is then displayed next to the aggregate peer rating calculated from reports from the other team members, allowing the two to be compared. Next, the number of action items completed during the week is reported in the "Done" column, followed by the number of currently active (not completed) action items. Finally, any freeform comments

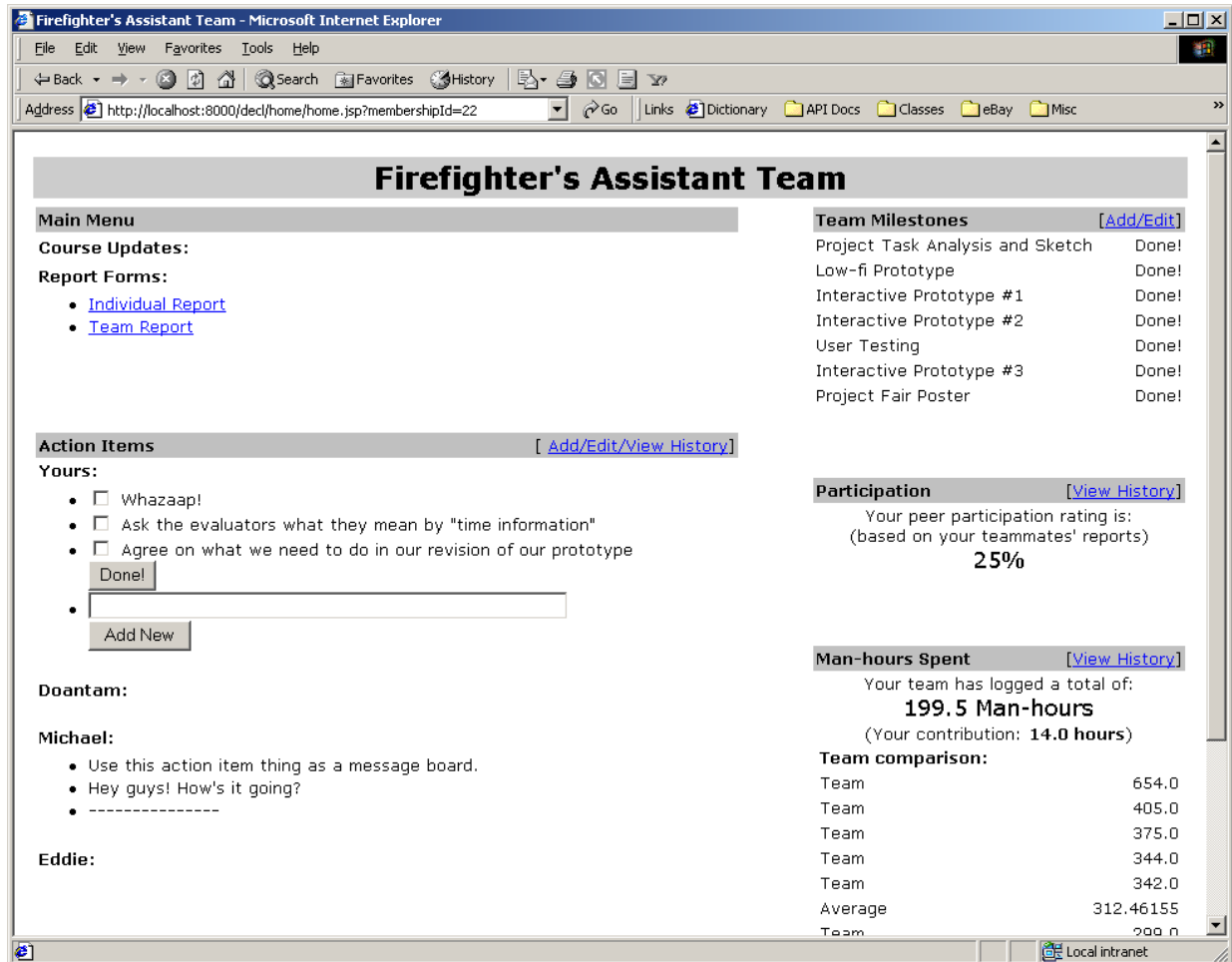


Figure 1. Student homepage showing metric summary feedback.

submitted with the weekly report are listed.

4. Pilot Evaluation 1

In the Fall 2000 semester, students in the undergraduate CS160 User Interface Design and Prototyping course at U.C. Berkeley were asked to use the system [7]. The course was composed of 48 undergraduate students in 12 teams of four students each, and a team of three graduate students. The instructional staff included the faculty instructor and one graduate student teaching assistant. The students were asked to each submit an individual report and collectively submit a team report on a weekly basis, although the content of the reports would not affect their grades in any way. In this course, each team works on a semester-long project involving the iterative design of a user interface. The performance of the team accounts for 40 percent of each student's individual grade. The demanding workload, consisting of interviewing, designing, programming, and user testing, requires that the team members collaborate effectively to produce a quality end product.

In addition to the pilot evaluation of the tool, each student in the course was loaned a lightweight laptop computer running the Windows 2000 Professional operating system. The laptops were

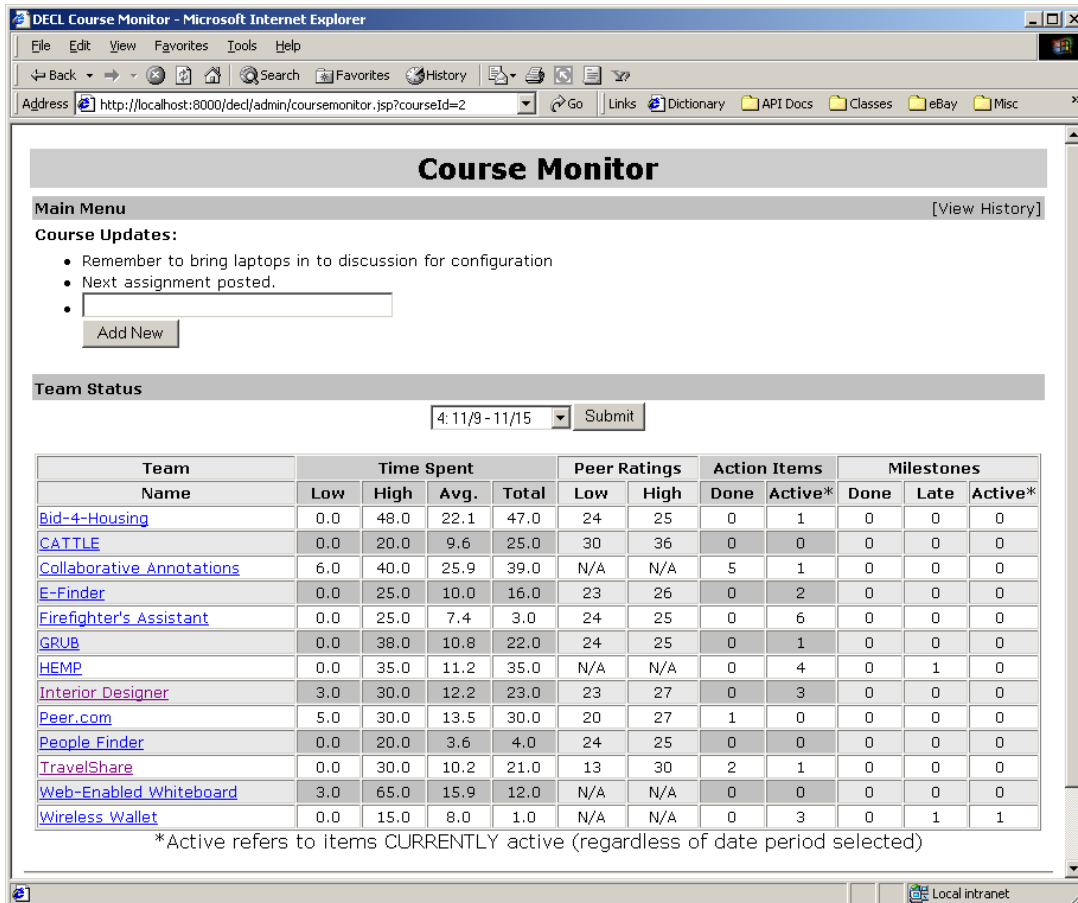


Figure 2. Instructor's course monitor homepage.

also equipped with wireless networking hardware, with coverage available throughout the computer science and engineering buildings. The laptops were deployed as an experiment to alleviate the space limitations of the departmental computer labs. However, we were also interested in the potential effects additional mobility and accessibility may have on student work practices.

The ActiveDesktop feature of the Windows operating system allowed us to configure each laptop so that the student homepage was always available as the background wallpaper of the user interface. When connected to a network, either wired or wireless, the ActiveDesktop provides an up-to-date view of the performance metrics that are refreshed every 10 minutes. When disconnected, the most recently cached copy of the homepage was displayed.

Our goals for the pilot evaluation were to identify any design issues with the system as well as analyze the data reported and collected. In particular, we were interested in how well the self-reported metrics reflect the actual work practices of the students. The student homepage provides an immediate summary of the reported data, including comparisons to other teams, which may lead to "gaming" the system so that the summaries favor particular students or teams. Finally, we conducted a number of informal surveys during and after the course to get feedback on the usage of the laptops and the system.

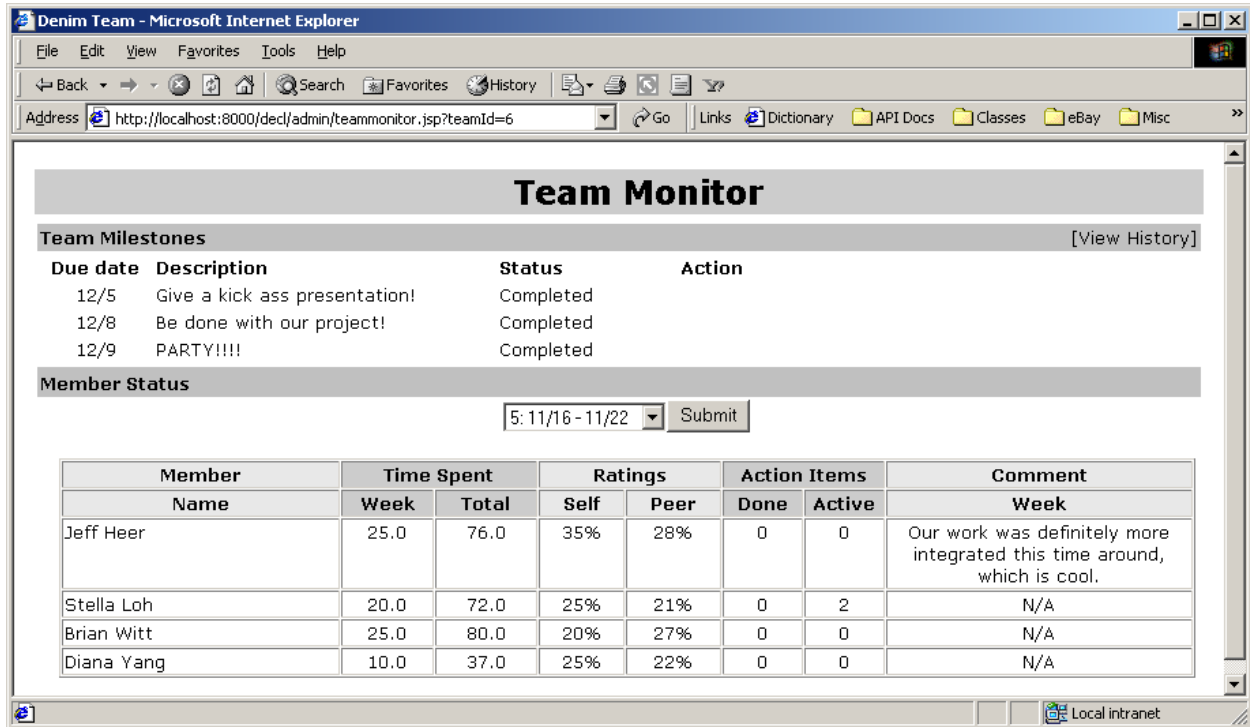


Figure 3. Instructor's team monitor homepage.

5. Pilot Evaluation 1 Results and Discussion

The pilot evaluation successfully revealed a number of bugs and design issues with the system, which unfortunately affected its usage by the students. Upon first receiving and logging into the laptops, there was an immediate outcry over the use of the ActiveDesktop feature to present the student homepage. Permissions were set such that it could not be disabled, but there was no control over the size and placement of the homepage on the desktop. As an act of defiance over that permissions policy, many students immediately resized the homepage window into a tiny box and moved it off-screen where it could be conveniently ignored. Regardless of the potential benefit of having the homepage immediately available, the desktop, considered "personal" space, is too intrusive a location.

A more significant design issue emerged from the requirement of weekly reports on a fixed schedule. Having laptops connected to a wireless network provided a greater degree of accessibility such that the students wanted to use the system more like a log or diary, submitting cumulative individual reports rather than waiting for the weekly report. The added burden of remembering the amount of time spent during the course of a week made students less inclined to utilize the reporting feature of the system. In addition, it was often easy for busy students to forget to file their report. An automated email reminder feature was designed but not implemented in time for the pilot evaluation.

Regardless of the difficulties, a total of 311 individual reports and 83 team reports were submitted over the 9 weeks of the course when the system was available. In the reports, the students reported spending an average of 10.4 hours a week individually (std dev = 10.5). Figure

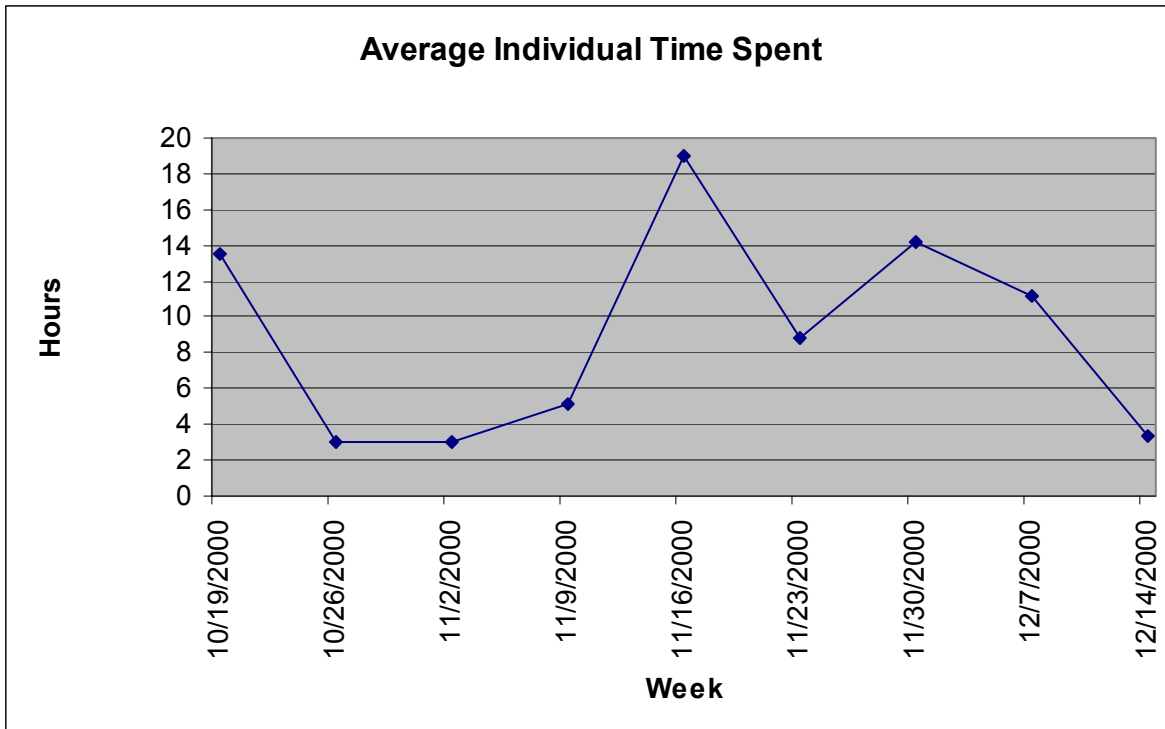


Figure 4. Average individual time spent reported per week.

4 plots the average individual time spent reported for each week. The peaks on October 19, November 16, and November 30 (weekly report periods started on Thursday) correspond exactly to weeks when a major deliverable was due in the course. This corresponds to the observed practice of last minute work and lends some credence to the accuracy of the self-reported hours, at least in relative terms between each week.

As a team, the students reported spending an average of 10.2 hours together each week (std dev = 12.4). The fact that this value is almost equal to the time spent individually suggests confusion in the definition of “team” time. Team time was defined as time spent in meetings and other coordination tasks on the project. However, the increased mobility provided by the laptops has changed the concept of team meetings for the students. Whereas it was previously difficult to work collocated in crowded labs, with laptops team members could meet and work side-by-side with much greater ease and frequency. In an exit survey conducted on the laptop usage, one student reported:

Working on the laptop made it very convenient to meet as a group, yet not everyone had to work on the project. Half could be working on their own thing, but when they were needed for advice or whatnot, they were always there.

Students often reported that they considered such time as “team” time and had difficulties determining a value to report. Another team described their work habits as “shifts”, where during intense periods there were always at least two members working together collocated. Again, it was difficult for the students in the team to separate the notion of individual and team time spent. Although we encourage formal team meetings between members as way of promoting industry

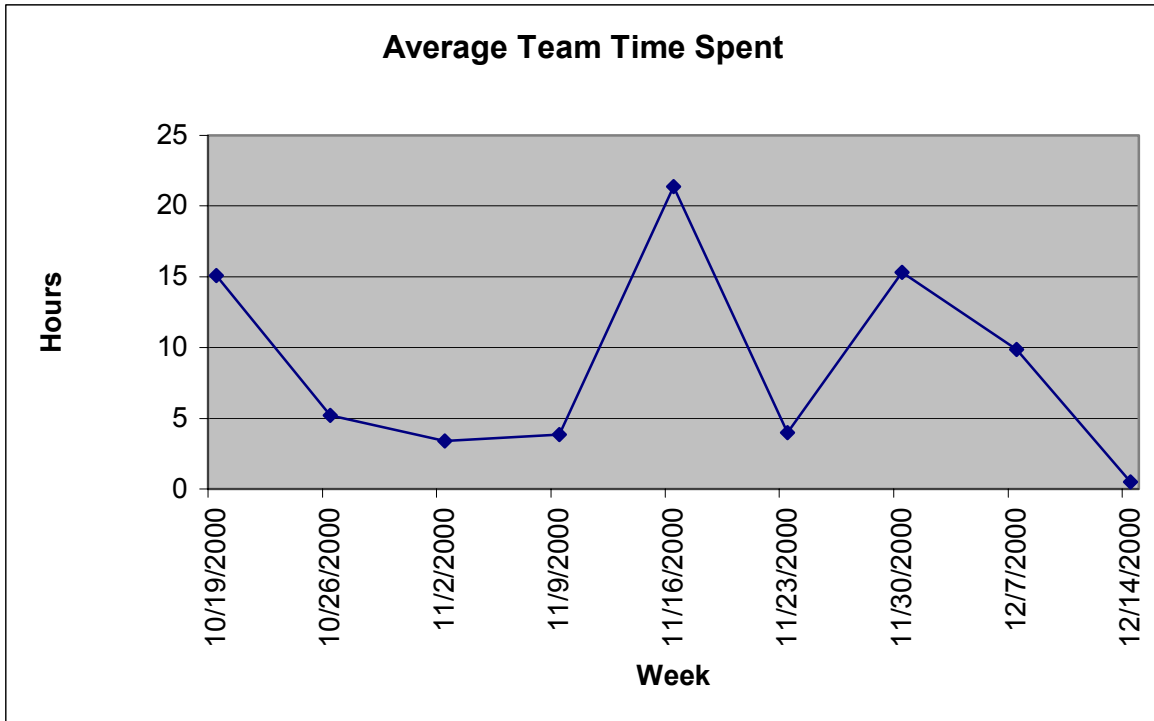


Figure 5. Average team time spent reported per week.

work practices, conflicting schedules and commitments to work and classes often make such meetings difficult. The students compensate by creating their own work practices, which should be accounted for in the design of our system.

Figure 5 shows a plot of the average team time spent reported for each week. Again, the peaks correspond exactly to weeks when a major deliverable was due in the course. The average team time reported is about the same or slightly greater than the average individual time reported.

The peer participation metric requires that each team member submit their ratings in order to perform the calculation. Since many reports were forgotten, most students were not able to have a peer participation rating displayed to them on the homepage, so it is unclear whether or not such information is useful or desirable. However, looking at the raw data and the comments submitted with the ratings, many students had no inhibitions about rating their team members' contributions (or lack thereof) higher or lower than average. This usefulness of this metric is still open for investigation.

When asked to rate their own contribution, students on average rated themselves as contributing 27 percent of the work in the team (std dev = 14.6). Given four students per team for all groups except one, this is very close to the equal contribution of 25 percent. The high standard deviation is largely due to confusion over contribution ratings during weeks when there were no team deliverables assigned. Most students simply rated their contribution as zero, while some submitted 100. Looking at the raw data and the comments submitted with the self-ratings, many students were very open in describing their contribution, often citing other course and work commitments when their contribution was low.

Free-form comments at the end of each report form were intended to give students an opportunity to reflect upon their work at the end of each week. In the first week in which the system was deployed, a number of comments were submitted suggesting potential problems:

Since it's so difficult to meet with everyone in the group at once, organization is a bit lacking.

Our group (and I am no exception) has a problem working together. We gravitate towards portioning out work and doing our assigned parts, emailing parts around once complete.

It's not as much a team environment as a group environment. This is primarily because we have such different schedules and requirements, different working styles, and lack of common inspiration for our mission statement.

I was going to come talk to you all about this, but I haven't gotten around to it—I've found that it's hard for me to get my group to actually do work on their own—I have to be there to keep things going, or do the work myself, otherwise things don't seem to get done right... I don't know if there's much you can do about it, and I'm hoping that for the next few projects I can get everyone to do a fair share.

The first two comments were submitted by members of the same team. The last comment suggests the utility of having an immediate outlet for free-form reflection as part of the reports. In each of these cases, the students were individually contacted and the problem was discussed. It cannot be determined if this early intervention was productive, but the corresponding teams did not report any problems in following weeks and produced high quality work.

Looking at the project management tools, only 23 students used the action items feature, creating 128 action items, of which 98 were completed, 8 were deleted, and 22 were not completed. Figure 6 plots the number of action items created during each week. Again, the peaks correspond to weeks where major deliverables were due. In observing the content of the action item descriptions, however, a few students used the feature to send short messages to their team members, since each member's action items are prominently displayed on the student homepage. Team message boards were considered in the design of the system, but not included due to time constraints. We consider the addition of message boards in a following section when we discuss future work in observing communication patterns.

Only seven of the 13 teams in the course tried using the milestones feature, defining only 28 milestones in total. The system starts out with no milestones defined, and only one team entered the major deliverable due dates into their schedule. Since the deliverable due dates are very clearly presented in the course through handouts and the course homepage, there was likely little benefit for the students to add them. More importantly, none of the milestones entered represented clear incremental steps between major deliverables. This may be an effect of having major deliverables almost every two weeks in the course, but better instruction on its use and possibly seeding the milestone schedule in advance may encourage the use of this metric.

The sporadic usage of the system dissuaded us from trying to find correlations between usage and

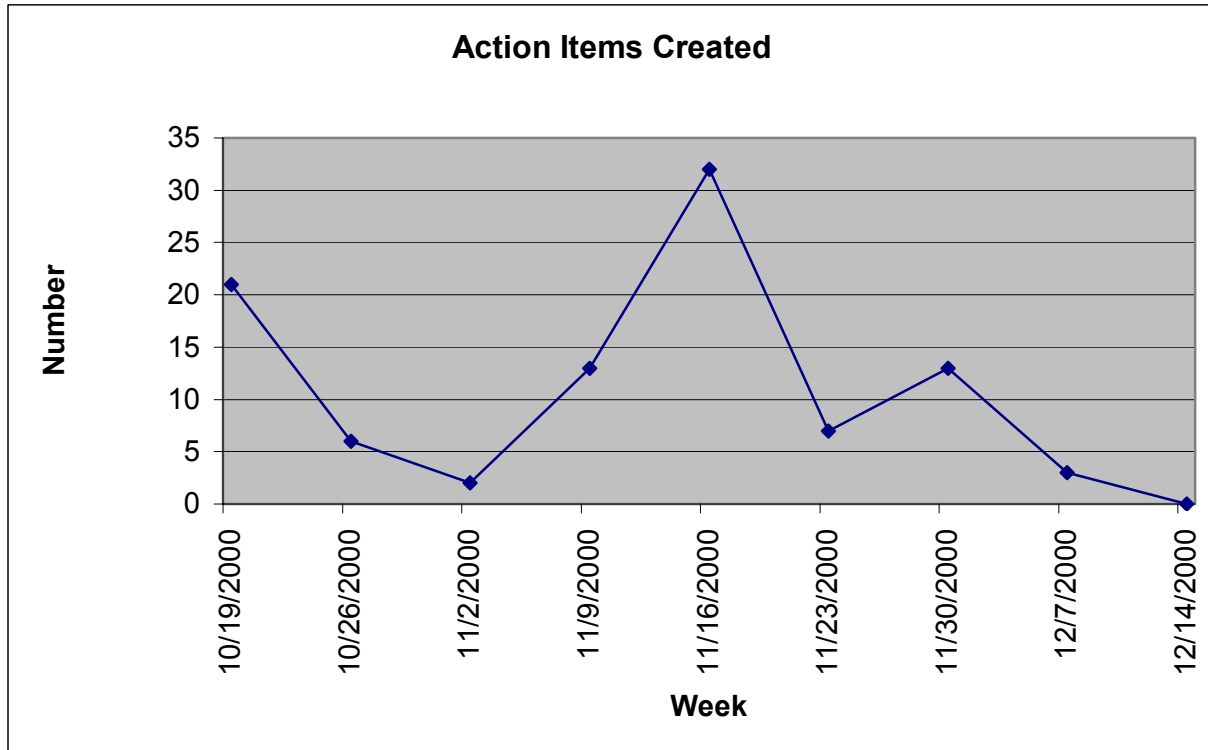


Figure 6. Number of action items created per week.

assessed team performance in the course. Team performance is assessed via written reports and short presentations that are subjectively graded by the instructional staff. The lack of specific and objective grading criteria or rubrics makes it difficult to normalize the grades to a common baseline for comparison with grades from previous offerings of the course.

6. Revised System Overview

The pilot evaluation revealed a number of design issues with the system design, some related to the introduction of laptops with wireless networking into student work practices. The break between semesters provided just enough time to incorporate bug fixes as well as a few design changes into the system for a second pilot evaluation in the spring of 2001.

The most significant design changes occurred in the collection of data from the students. Reporting the individual time spent is now separate from the weekly individual report (the weekly individual report is still required to collect peer ratings and freeform comments from the members of the team). Figure 7 shows the Time Spent Report form, which allows the student to submit the number of hours worked on the project since the previous report. The form also shows the previous reports submitted. It is expected that the students will use this as a cumulative log as described earlier; when a student finishes working on their wirelessly networked laptop, they can log into the system and immediately report the amount of time spent.

DECL Team

Time Spent Report

Since your last report (if any) until now you have spent:

Hours:

Comment: (optional)

Previous Reports

Report Date	Time Spent	Comment
1/24	5.0	Busy with project in CS162 this week
1/25	15.0	Spent an all-nighter to catch up

Return [home](#)...

Figure 7. New individual time spent report form.

The concept of team time spent in meetings was refined so that it explicitly involves discussion and decision making rather than simply working on the project collocated with team members. The new Team Meeting Report form, shown in Figure 8, allows a member to fill out a report that includes the amount of time spent, the location of the meeting, the team members present, and a short summary of the meeting. The form also shows previous meeting reports.

The instructor monitoring pages were unchanged except for the addition of a “Send Reminders” feature on the course monitor page. When viewing the summary for a particular week in the course, the instructor could use this feature to send a reminder email notice to students that had not submitted an individual report for that week. The reminder notice included a URL to the team homepage that the students could immediately click on to open the browser and submit their report.

7. Pilot Evaluation 2

In the Spring 2001 semester, the same instructor from the previous evaluation agreed to use the

Team Meeting Report Submission - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Folders Search NBCi QuickClick Off Dictionary

Address bmrc.berkeley.edu:8000/decl/home/report/meeting.jsp?membershipId=1 Go Links Dictionary

Team Meeting Report

1. Date

Date: / /

Time: :

Location:

2. Time Spent

How long was this meeting?

Hours:

3. Attendees

Please check the box for each team member that is **present** at this meeting. Provide a brief explanation if a member is not present.

<input type="checkbox"/>	Francis	<input type="text" value="Attending a talk..."/>
<input checked="" type="checkbox"/>	Jimmy	<input type="text"/>
<input checked="" type="checkbox"/>	Jonathan	<input type="text"/>
<input checked="" type="checkbox"/>	Jason	<input type="text"/>

Done Internet

Figure 8. New team meeting report form.

revised system in the undergraduate CS169 Software Engineering course [8]. Similar in structure to the CS160 User Interface Design and Prototyping course, 33 undergraduate students worked in seven teams on a semester long project. There was no teaching assistant, but seven MBA students from the Haas School of Business on the U.C. Berkeley campus each acted as a “manager” for one of the teams. The MBA students received credit for participating in this course as well as taking a complementary technology management course taught by the same instructor.

All students, including the MBA managers, were asked to use the system to manage their project and fill out weekly individual reports (now separate from individual time and team meeting reports). Again, the content of the reports was not graded. Instead of making the team homepage the active desktop on the laptops, it was accessible through a link on the general course homepage and through the reminder email notices.

8. Pilot Evaluation 2 Results and Discussion

Although the system was received with less resistance in this second pilot evaluation, it was compensated for with apathy. Even with regular email reminders before each weekly report deadline, a total of 383 individual reports were submitted over the 16 weeks of the class, or only 60% of the total number of reports expected with full participation. Over the course of the semester and in a post-semester survey, most students cited inconvenience as the main deterrent to the use of the system. The cost of effort to be online, browse to the website, and log in with a username and password proved to be too high when compared to self-reported student work practices such as “set[ing] the milestones on a scrap of paper and hang[ing] it on the wall or ... by the keyboard” or simply “mentally set[ting] a goal for myself”.

As a means for communicating with other teammates, the system did not provide the largely synchronous communication needs of the students. In the post-semester survey, 12 of the 32 respondents explicitly reported using face-to-face meetings as the primary way to maintain awareness of their team members. Four respondents explicitly reported using email, two using phone calls, and one using instant messaging. Some who tried using email found it insufficient for their needs:

We tried to communicate using group emails. Some of us got annoyed by tons of emails that didn't have to do with us; we [had] split up tasks based on big chunks of the functional specs, so we felt that only a few had to communicate. It would have been better to keep a more open line of communication, besides just the emails. When it came time for integration, there seemed to be a bigger need for communication than we originally thought.

I used e-mail and phone calls to find out where my teammates were. E-mail wasn't very effective because of the response time or no response at all. Phone calls were better in terms of communication and response time.

We discuss the implications of these findings and ideas for integrating both synchronous and asynchronous communication tools into the system in the future work section.

Despite having only 60% of the expected data in the system, we performed a preliminary analysis of both the individual and team data. On an individual basis, we looked at the following metrics aggregated over the course of the entire semester: number of weekly reports submitted, number of action items completed, number of individual time reports submitted, total individual hours logged, mean self-participation rating, mean peer participation rating, the offset of the mean peer participation rating from the mean over all members of the team, and the final individual grade score. The mean peer participation rating over all members of the team represents the value for equal participation, so the offset shows the member's participation relative to that “ideal.”

Analyzing the participation ratings exposed a slight complication due to the participation of MBA student managers into the team. Since they were not directly involved in the implementation of the software project, their participation ratings were usually low, in the 5-10% range. These low ratings skew analyses of how equally the members are participating. Therefore, analyses of peer participation ratings in this paper have been performed with the manager rating removed. If having a manager is to be a common practice, the system must be revised to support

Table 2. Standard deviation of peer ratings and instructor performance rating for each team. The performance rating scale is from 1=“not well at all” to 7=“very well.”

Team	Std. Dev. of Peer Ratings	Instructor Performance Rating
1	0.71	7
2	1.24	5
3	0.35	4
4	0.49	4
5	2.30	1
6	0.52	5
7	0.21	4

team members with different roles so that analyses can be performed appropriately.

The Pearson’s correlation coefficient was calculated for each pair of these metrics for each student. Significant (0.05 level, 2-tailed) weak to moderate correlations were found between the following metrics and the final individual grade score: total individual hours logged ($r = 0.510$), mean peer participation rating ($r = 0.356$), and the offset of the mean peer participation rating ($r = 0.423$). Although the total individual hours logged may not be accurate, as some students did not log any hours at all, it would appear that the students who expended the effort to log their hours are also the ones who perform well in the course.

On a team basis, we looked at the following metrics aggregated over the course of the entire semester: number of meeting reports submitted, total meeting hours logged, number of milestones reached, total individual hours logged from all members of the team, the mean standard deviation of the peer participation ratings, a rating of overall team performance by the instructor, and the final team grade score. The instructor was asked to rate how well each team worked together in a post-course survey. The standard deviation of the peer participation ratings describes how much they vary from the mean, where the mean is the value for equal participation amongst the members of a team. Thus, a larger standard deviation suggests a greater inequity in participation.

The Pearson’s correlation coefficient was calculated for each pair of these metrics for each team. No significant correlations were found, but a subjective inspection of the data suggests that the standard deviation of the peer participation ratings may still be a useful metric. As seen in Table 2, Team 5 had a large deviation that matched a very low rating in performance by the instructor. Although the large deviation for Team 2 is a false positive, it is an error on the side of caution as instructor intervention on a false positive wouldn’t hurt, but missing a poorly performing team would.

9. Future Work

Many difficulties in team collaboration are often due to poor communication between members,

as seen in some of the comments quoted previously. An area for investigation that we are interested in pursuing is the observation of both synchronous and asynchronous messaging between students for social network analysis. Such an analysis would provide a metric of communication “connectivity” between students in teams, and may be useful in identifying social loafers or other dysfunctional members. For example, a few students used the action items feature on the homepage for posting short messages to each other. By providing message board services as part of the homepage, we can facilitate asynchronous communication and monitor the participation of each member of the team via their postings.

An additional source of communication information is synchronous “instant” messaging. Each student in the course was asked to sign-up for a Yahoo! Instant Messenger account and each laptop was configured with the corresponding messaging application. The main incentive for students to sign-up for the account was to have an additional means of contacting the instructional staff. Significant usage of instant messaging was observed between students and the teaching assistant, primarily for asking questions on course material and assignments. In addition, a fair amount of usage was observed between students in the course. We are interested in deploying an instrumented instant messaging system for logging and analysis of these communication practices as well.

Other areas of interest include monitoring shared artifacts, such as source code and other documentation files, to analyze their access patterns by members of the team. As the scope of the analyses and the number of students monitored increases, it has become apparent that maintaining the development of our current research system may not be feasible. Although the source code can be made available for the current system to encourage development, there are many other collaboration tools that may provide more robust bases upon which to add our functionality. Products currently being investigated include the ArsDigita Community System, which is available for development under an open source license [9], the Basic Support for Collaborative Work system developed at GMD [10], SourceForge from VA Linux [11], Groove from Groove Networks [12], and Microsoft SharePoint Portal Server and Team Services [13].

For the instructor, we need to address the problems of attention and usability. The design of the instructor’s monitoring pages presented too much information for the instructor to process. The usability of these pages needs to be further evaluated to streamline the information being presented, possibly using more automated analysis, based on our preliminary results, to summarize the data. Other ways to minimize instructor effort include automated alerts via email that draw the instructor directly to potential problem teams.

Finally, we are actively recruiting faculty and student researchers in the U.C. Berkeley School of Education to participate in this project. In particular, we are seeking their assistance in planning a more thorough assessment of learning outcomes using the system. As the system becomes more robust and functional, we would like to deploy it in incrementally larger courses in the computer science department.

10. Related Work

Project management and team collaboration is a large commercial enterprise as well as an

academic interest, with many industry consulting companies, process developers, and software packages available. Software products such as Microsoft Project provide managers with tools to schedule and organize resources and budgets as well as disseminate this information to team members using Gantt charts and other common project management processes [14]. However, by focusing on areas such as resource and budget allocation, these tools often provide more benefit to managers than the team members, leading to a lack of adoption and use by the members.

Software that is more directly focused on actual use by team members includes the aforementioned collaboration tools Groove, SourceForge, and Microsoft SharePoint. These tools provide collaboration facilities such as shared document repositories, asynchronous messaging, and bug and feature tracking. The extent to which these tools can be customized and instrumented so that we can analyze usage patterns will determine whether or not we can utilize them in our own studies.

The need to support collaborative teams in engineering education has led many educators in academia to create their own tools. These include the web-based software engineering management tool by Collafello and Hart which we based many of our metrics upon, as well as tools like Team Facilitator by Kehoe and Guzdial from Georgia Tech [15]. These tools offer most of the same features we have deployed, but we aim to extend such functionality by including greater automated analysis of the data being collected and presented.

Commercial tools designed for educators include WebCT [16] and Blackboard [17], two highly popular systems for online course management. These tools provide support for publishing common course materials on the web, including announcements, assignments, grades, and quizzes, in addition to communication and collaboration tools such as message boards and chat rooms. Intended for educators that are not necessarily familiar with web publishing, these tools are closed proprietary systems deployed on a campus-wide basis or hosted off-site. There is no customizability beyond choosing from the functionality they provide, and they provide minimal support for supporting groups and teams within a course. As a result, these tools are not easily modified for our purposes.

The use of laptops by students has been studied as well. The highly publicized Rockman study sponsored by Microsoft showed that the use of laptops in high school promoted many advantages including greater motivation, better problem-solving skills, and increased collaboration among students [18]. More recently, studies performed by the Nomad research group at Cornell University have been looking into wirelessly networked laptop usage in higher education [19]. Their investigations into information access patterns via web browsing have shown mixed results for the advantages of laptop use, depending upon the context of the usage (e.g., during class or while studying) and the course being evaluated [20]. Whereas the Cornell studies have so far looked primarily at individual student performance, we have been surveying and observing laptop usage in the context of team collaboration work practices.

11. Conclusion

We have described a computer-based system for monitoring the performance and progress of student teams in engineering education. The system utilizes student self-report and instrumented

project management tools to collect data supporting a set of metrics of team performance and progress. Our pilot evaluation of this system, as used in undergraduate design courses, has shown that the data reported and collected using the system does reflect the work habits informally observed by the instructional staff. In addition, a number of design issues have been discovered that reflect changing work practices introduced with the addition of wirelessly networked laptops into the course.

12. Acknowledgements

This work was funded by grants from Intel Corporation and the National Science Foundation (#EEC-008052).

13. References

1. Barrows, H.S. and R. Tamblyn, *Problem-based Learning: An Approach to Medical Education*. 1980, New York: Springer Publishing Company.
2. Springer, L., M.E. Stanne, and S.S. Donovan, *Effects of Small-Group Learning on Undergraduates in Science, Mathematics, and Technology: A Meta-Analysis*. Review of Educational Research, 1999. **69**(1): p. 21-51.
3. Collofello, J.S. and M. Hart. *Monitoring Team Progress in a Software Engineering Project Class*. in *29th ASEE/IEEE Frontiers in Education Conference*. 1999. San Juan, Puerto Rico.
4. McConnell, S.C., *Rapid Development: Taming Wild Software Schedules*. 1996, Redmond: Microsoft Press.
5. Schulz, K.H. and D.K. Ludlow, *Incorporating Group Writing Instruction in Engineering Courses*. Journal of Engineering Education, 1996: p. 227-232.
6. Trytten, D.A. *Progressing from Small Group Work to Cooperative Learning: A Case Study from Computer Science*. in *29th ASEE/IEEE Frontiers in Education Conference*. 1999. San Juan, Puerto Rico.
7. *CS160 Fall 2000 course homepage*. <http://bmrc.berkeley.edu/courseware/cs160/fall00/>
8. *CS169 Spring 2001 course homepage*. <http://bmrc.berkeley.edu/courseware/cs169/spring01/>
9. *ArsDigita Community System*. <http://www.arsdigita.com/>
10. *Basic Support for Collaborative Work*. <http://bscw.gmd.de/>
11. *SourceForge*. <http://www.sourceforge.net/>
12. *Groove Networks*. <http://www.groove.net/>
13. *Microsoft Sharepoint Server and Team Services*. <http://www.microsoft.com/sharepoint/>
14. *Microsoft Project*. <http://www.microsoft.com/office/project/>
15. Kehoe, C., M. Guzdial, and J. Turns. *What We Know About Technological Support for Project-Based Learning*. in *27th ASEE/IEEE Frontiers in Engineering Education Conference*. 1997. Pittsburgh, PA.
16. *WebCT*. <http://www.webct.com/>
17. *Blackboard*. <http://www.blackboard.com/>
18. Rockman ET AL, *Powerful tools for schooling: Second year study of the laptop program*. 1996, Rockman ET AL: San Francisco.
19. *Nomad Project*. <http://www.nomad.cornell.edu/>

20. Grace-Martin, M. and G. Gay, *Web Browsing, Mobile Computing, and Academic Performance*. Educational Technology & Society, 2001. 4(3): p. 95-107.