STUDENT ASSESSMENT IN A REAL-WORLD-PROJECTS CAPSTONE COURSE IN COMPUTING

AUTHORS REDACTED TO PRESERVE ANONYMITY

ABSTRACT

The capstone computing course at our university provides students with experience working on computing and information systems projects with real-world aspects, and the students have the opportunity to develop both the hard and soft skills that are sought after by industry. A unique peer evaluation system and various assessment tools have been developed recently to facilitate student assessment. This paper describes the student assessment techniques used in capstone courses at other universities and then describes the assessment techniques currently used in our capstone course in computing.

INTRODUCTION

The capstone computing course at our university has been offering masters-level students experience with the development of real-world computing projects for over fifteen years. The course has evolved from face-to-face delivery in the first five years to a web-assisted course involving dispersed teams today. With dispersed teams we needed to revisit the way we graded the performance of team members. It is well known that projects undertaken by groups lacking co-presence presuppose a higher level of organizational and process skills among their members [5].

A capstone course is a course offered as part of an academic major aiming to bring together major aspects of the academic disciplines related to the said major [7]. The aim of our capstone course in computing is to familiarize students with how their trade is plied in organizations, so that the program of study delivers "the practice" part of the promised "theory and practice." The projects are "real world" in every respect as they entail the development of an application desired by a real world customer. As in industry, applications are developed by a small, collaborative team which needs to communicate with the customer, coordinate its activity, attend to internal decision-making, and, as observed by [6], be sensitive to delivering value. The applications press into service current technology. This is technology with which the students are often unacquainted inasmuch as it may be specialized, new, or at least new to them. Students learn about real-world technology through their own group's experiences as well as through the reports from other groups. A soft skill of transcending importance, emphasized by activities throughout the capstone, is the ability to communicate on technical concepts and issues; orally, in written reports, and via Web media; to both peers and lay people. The soft skills acquired through a capstone course are perhaps one of the greatest rewards of this academic experience. These include problem solving, communication, and teamwork skills which are becoming essential for work in industry [8].

A 2015 paper [10] provided a fifteen-year review of our capstone course in computing[1]. In the fifteen years (2001-2015) since the capstone course assumed its project-based form, the most significant change has been in its presentation. For the first five years the course spanned the fall and spring semesters and was face-to-face. In 2006 the course was condensed into a one semester offering. For projects, this meant accelerating requirements elicitation, system development, and testing. We responded with agile

[1] Some of the background material here comes from this earlier paper.

methodology. In 2006 the course delivery shifted from face-to-face to "hybrid", where students collaborate remotely except for three meetings – at the beginning of the semester for orientation, at the middle of the semester for midterm project status reports to the class, and at the end of the semester for final project presentations.

Our project teams are largely geographically dispersed. A geographically dispersed team (also known as a virtual team, distributed team, or remote team) is a group of individuals who work across time, space, and organizational boundaries with links strengthened by webs of communication technology [12]. These teams are similar to traditional teams but are geographically dispersed and rely heavily on virtual methods such as email and virtual conference applications. Instructors believe experience working on a virtual team prepares students for the growing business demand [9].

The focus of this paper is on student assessment where significant changes have been made over the last two years. The remaining sections of the paper cover the following material: section 2 surveys student assessment in capstone courses at other institutions; section 3 describes the current procedures for student assessment at our university; and conclusions are drawn in Section 4.

CAPSTONE PROJECT STUDENT ASSESSMENT AT OTHER INSTITUTIONS

An earlier study contacted 84 universities having capstone courses in computer science or engineering to get a sense of how various academic institutions implement their capstone course [2]. Of the 84 universities polled with a set of 16 questions, 34 responded, including CMU, U. California, U. Maryland, U.S. Air Force Academy, and U.S. Naval Academy. For schools that did not respond, information in some cases was obtained from the internet, resulting in appropriate information from up to 49 institutions, depending on the information. To put this work into context, the information from that study concerning student assessment is described here.

Most of the capstone instructors polled required some sort of progress report to be turned in regularly. The most common frequency was weekly at 48% since this coincides with a typical course schedule and is frequent enough to allow the instructor to track progress. Bi-weekly progress reports were required in 17% of the polled schools, and 9% allowed more than two week intervals between progress reports. Surprisingly, 26% of universities did not require progress reports, indicating that time management was the responsibility of the student team.

Capstone project grades are usually based in part on peer reviews designed to evaluate the performance of team members as seen through the eyes of team members. Peer reviews take into consideration the project goals, team communication, and division of labor. Geographically near teams have the opportunity to meet in person on a regular basis. Team members can take cues from one another through in-person interaction and the responsibility of each team member is clear. Mid-semester peer reviews assist the instructor in identifying contribution or issues among the team members that need to be addressed before the project can be impacted negatively.

The weekly time required of students varied greatly from two hours per week to forty hours per week. A majority (60%) of programs expected more than ten but less than twenty hours weekly.

CAPSTONE PROJECT STUDENT ASSESSMENT AT OUR UNIVERSITY

Student assessment is as follows: individual quizzes (20%), first quarter team project (10%), second quarter team project (20%), third quarter team project (20%), and fourth quarter team project (30%). Note that the project grades increase percentagewise as we get further into the semester – Q1=10%, Q2=20%, Q3=20%, Q4=30% – so students can more easily recover from poor early grades. Thus, 80% of a student's grade is based on their contribution to the project team effort with the quizzes, based primarily on the textbook material, providing the only direct individual assessment. The team has the ultimate responsibility for the

project work and is graded accordingly. Grades on team events are determined by first assigning a team grade and then adjusting an individual student's grade up or down based on peer, customer, and instructor evaluations.

Assignment of Team Grades

A team grade is assigned to each team at each of the four quarterly checkpoints. The quarterly evaluations are based on the following:

- Weekly team submission of project activity report.
- Quarterly team submission of a project technical report that, by the end of the semester, grows into a technical paper that for most teams is published in an annual internal computing conference.
- Individual student responses to questions for the specific quarter time interval – the number of hours per week spent on project work, their specific contributions, their strengths and how they were used, their areas needing improvement, and what has enhanced and/or challenged their team's performance.

A team grade is based on the quantity and quality of the team meetings, and an evaluation of the team's project technical report. Each team submits their draft of the technical report on turnitin.com. To check for potential plagiarism, turnitin.com uses advanced machine learning technologies to generate similarity reports for the drafts submitted compared to texts available on the Internet [11]. If a team has a similarity score of greater than 25%, the team can resubmit their draft. All teams have the option of working with the university's writing center before submitting their drafts. A special arrangement has been made with writing center coordinators to work with the teams on how to write a technical paper. Drafts are given a letter grade based on their technical correctness, appropriate citations, how well the draft follows IEEE format, etc. For the first two drafts, each team is provided with extensive feedback on their drafts so they can improve on their writing in subsequent drafts.

Assignment of Individual Team Member Grades

Peer, customer, and instructor evaluations are used to assess the project contributions of each team member. The peer evaluation procedure is first described and then modified to include customer and instructor evaluations. Peer evaluations, although used earlier when the course was conducted in the classroom, are even more critical for distributed teams because some team members have minimal direct contact with the customer and instructor. Obtaining individual student grades on teamwork has been reported in the literature. For example, one system is somewhat similar to ours but uses more granular numerical input [3], another is an elaborate web-based system that records and tracks self and peer evaluations [4], and yet another uses survey instruments [13].

The students are required to provide peer evaluations four times during the semester, one at each of the quarterly checkpoints. Table 1 shows the recent two-year evolution of peer evaluation charts with associated grades for a four-member team. Each of the four evaluation columns shows the evaluation of a team member's evaluation input. The summary ("Sum") column shows the adjusted sum of each row of evaluations, and the grade column shows the student grades. Here, a team grade of 85% is first determined and then individual team member grades are adjusted relative to the team grade.

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Sum	Grade
1	+	=	−−	++++	+++	91
2	=	=	−	−−−	−−−−	77
3	−	=	++++	−−	+	87
4	=	=	−	+	=	85
Average	=	=	=	=	=	**85**

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Sum	Grade
1		=	−−	++++	++	89
2	+		+	−−−	−	83
3	−−	=		−	−−−	79
4	+	=	+		++	89
Average	=	=	=	=	=	**85**

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Sum	Grade
1		4	2	8	4	89
2	5		4	0	-1	84
3	0	3		2	-5	80
4	5	3	4		2	87
Total	**10**	**10**	**10**	**10**	**0**	**85**

Table 1. Evolution of team peer evaluation charts:
prior to fall 2015 (top), fall 2015 (middle), and current (bottom).

In the top chart, used prior to the fall 2015 semester, each column must average "=", and student grades were adjusted up or down 2% for each "+" or "–" sign obtained in the summary row. The middle chart, used in the fall 2015 semester, varied slightly from the top chart with the difference being that the students could not evaluate themselves, and in this example the inputs (+, -, =) have been adjusted slightly from the top chart to average "=". The top-chart method favors and thus encourages students to give themselves high evaluations. The middle chart corrects this problem and shows that the student who did not give himself/herself a high evaluation in the top chart now has a better grade. The bottom chart, first used in the spring 2016 semester, employs a numerical system where each team member allocates a total of 10 points among the other team members (again no self-evaluation) so that each evaluation column adds to 10. Each entry in the summary column is obtained by adding the row values and subtracting 10, so that the average of the summary column adds to zero. The summary value is then used to adjust the team grade for each team member. This numerical method simplifies the awkward accounting for the +, -, and = signs and is basically the one described by [1].

The student peer evaluations are obtained at each quarter over the Internet through a Google Form (Figure 1). In the Google form, students are required to select their team identification number, enter their name, and rate the team contribution of each of their fellow team members (peers). In rating their peers, a total of 10 points, using integer values, must be distributed among the other team members. Students can fill

out the form more than once, but only the latest version of the peer evaluation is considered in the grading process. After the submission deadline, a response sheet is generated by Google in the form of Google Sheets. This document is then downloaded in the form of Microsoft Excel spreadsheet. The data are initially sorted by name so that only the latest version of the peer evaluation is considered. After removing duplicates, data are again sorted by team identification number.

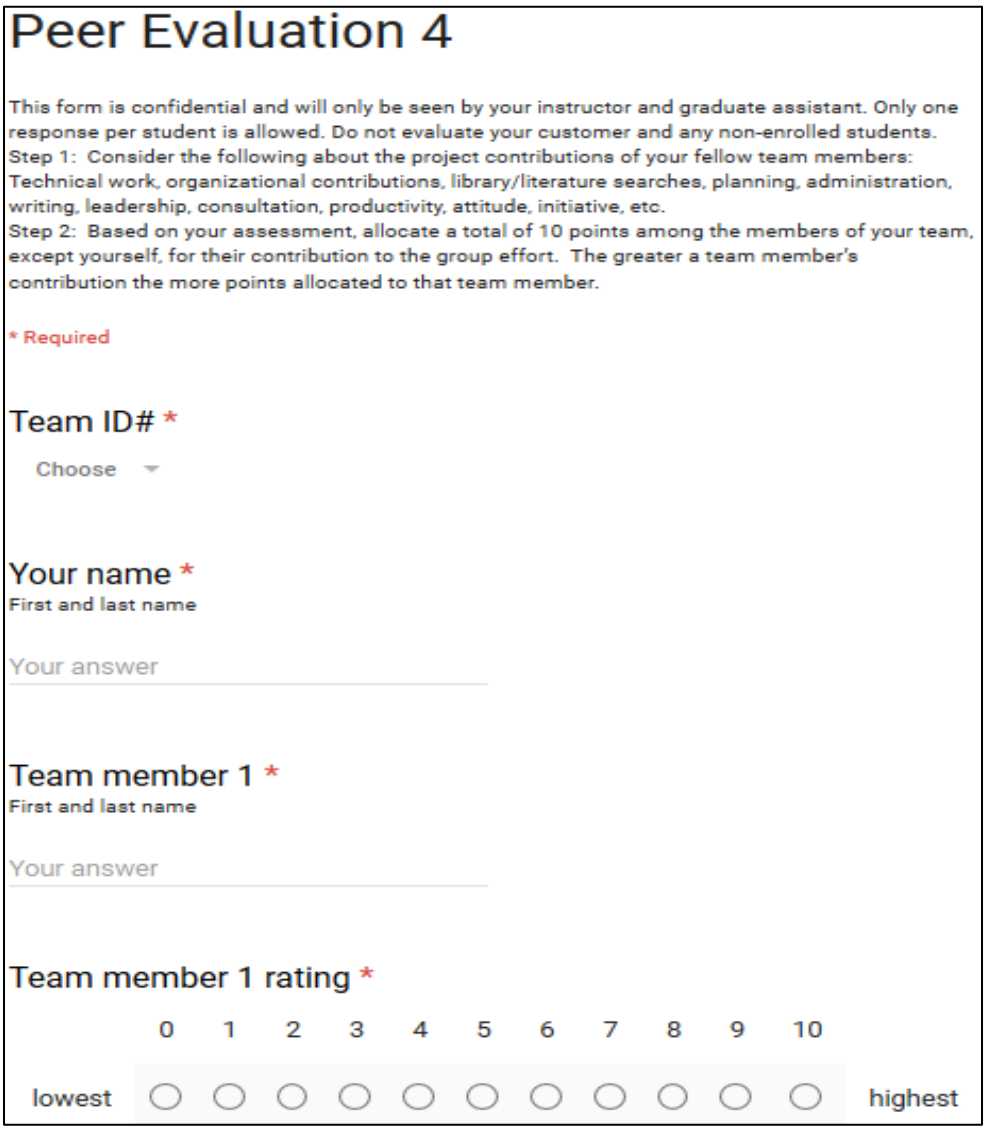

Figure 1. Peer Evaluation Google Form for Quarter 4.
The rating input area for only one team member is shown here.

Occasionally, a student does not submit proper peer evaluations and this is handled as shown in Table 2. For comparison, the upper chart shows all team members evaluated equally, and decimal evaluations are necessary to do this. The lower chart shows how a student who does not submit peer evaluations or submits them in error (not using integer values, values not adding to 10, etc.) is evaluated. The student at fault, in this case team member #2, is assigned -3 points and the other team members assigned values so the total distributed adds to 10, and non-integer values can be used here.

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Sum	Grade
1		3.3	3.3	3.3	0	85
2	3.3		3.3	3.3	0	85
3	3.3	3.3		3.3	0	85
4	3.3	3.3	3.3		0	85
Total	**10**	**10**	**10**	**10**	**0**	**85**

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Sum	Grade
1		4.3	3.3	3.3	1	86
2	3.3	-3.0	3.3	3.3	-3	82
3	3.3	4.3		3.3	1	86
4	3.3	4.3	3.3		1	86
Total	**10**	**10**	**10**	**10**	**0**	**85**

Table 2. Handling a student who fails to submit a proper peer evaluation. The student in question, highlighted in yellow, receives -3 points and the remaining students receive points to yield a sum of 10 points.

Customer and instructor (and possibly graduate assistant) evaluations are similarly included as shown in Table 3. The instructor's evaluation of a student's contribution to the team effort can be influenced by the student's responses to the general questions, such as the number of hours per week spent on project work, and additional input can be obtained by discussing team member contributions with the team leader. The top chart shows the evaluations without customer and instructor additions (from Table 1, bottom chart). The bottom chart shows the customer and instructor additions when they both agree team member #1 was best and #4 second best.

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Eval Cust	Eval Inst	Sum	Grade
1		4	2	8			4	89
2	5		4	0			-1	84
3	0	3		2			-5	80
4	5	3	4				2	87
Total	**10**	**10**	**10**	**10**			**0**	**85**

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Eval Cust	Eval Inst	Sum	Grade
1		4	2	8	5	5	9	94
2	5		4	0	1	1	-4	81
3	0	3		2	1	1	-8	77
4	5	3	4		3	3	3	88
Total	**10**	**10**	**10**	**10**	**10**	**10**	**0**	**85**

Table 3. Adding customer and instructor evaluations (yellow).

On rare occasions, students will try to "game" the system, and this is usually a pair of students who give each other a high evaluation. A review of the summary charts will usually discover such anomalies because these high evaluations appear as outliers relative to the other evaluations. This is illustrated in Table 4. In the top chart of the table, team members #2 and #3 give each other 10 points and the others none, and this clearly does not agree with the evaluations assigned by the other team members and especially those assigned by the customer and instructor. Such situations are handled in the same manner as the students not submitting evaluations, by assigning the "gaming" students -3 points and assigning the other team members values so the total distributed adds to 10, as shown in the bottom chart.

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Eval Cust	Eval Inst	Sum	Grade
1		0	0	8	5	5	3	88
2	5		10	0	1	1	2	87
3	0	10		2	1	1	-1	84
4	5	0	0		3	3	-4	81
Total	**10**	**10**	**10**	**10**	**10**	**10**	**0**	**85**

Team Member	Eval 1	Eval 2	Eval 3	Eval 4	Eval Cust	Eval Inst	Sum	Grade
1		4.3	4.3	8	5	5	11.7	96.7
2	5	-3.0	4.3	0	1	1	-6.7	78.3
3	0	4.3	-3.0	2	1	1	-9.7	75.3
4	5	4.3	4.3		3	3	4.7	89.7
Total	**10**	**10**	**10**	**10**	**10**	**10**	**0**	**85**

Table 4. Team members 2 and 3 tried to "game" the system by giving each other the full 10 points (top chart), and the revised chart (bottom).

Individual student grades are posted each quarter as obtained from the appropriate peer evaluation summary charts. These charts are created in a spreadsheet that automatically makes all the computations once the project grade (lower right-hand corner) and the peer, customer, and instructor evaluations are entered. We are investigating methods of automatically entering the information from the Google Form.

CONCLUSIONS

Capstone courses are particularly important to computing and information systems education. Students develop hard and soft skills, are exposed to a wide range of topics, and foster interdisciplinary collaboration. Student assessment can be difficult in capstone courses where most of the work is performed in geographically dispersed teams The assessment techniques described here have been tested and shown to be successful over the last two semesters and should be of value to other institutions.

REFERENCES

[1] Brooks, F.P., Keynote Address, EDSIG 2015 Conference, Wilmington, N.C., November 2015.

[2] Brewer, T., Campoverde, N., Hanna, J., Humphries, B., & Tappert, C.C. (2015). Analysis of Capstone Computing Courses. Proc. Student-Faculty Research Day, CSIS, Pace University.

[3] Brown, R.W. (1995). Autorating: Getting Individual Marks from Team Marks and Enhancing Teamwork. *Proceedings of the Frontiers in Education Conference*, IEEE/ASEE, Pittsburgh, Pa.

[4] Clark, N., Davies, P., & Skeers, R. (2005). Self and Peer Assessment in Software Engineering Projects. *Proceedings of the 7th Australasian Conference on Computing Education*, Newcastle, Australia.

[5] Cusumano, M. (2008). Managing Software Development in Globally Distributed Teams. *Communications of the ACM*, 51(2), 15-17.

[6] Denning, P.J. & Dunham, R. (2001). The core of the third-wave professional. *Communications of the ACM*, 44(11), 21-25.

[7] Ford, M. (2002). *Beyond the Modern University: Toward a Constructive Postmodern University*. Praeger.

[8] Gardiner, B. (2015). Soft skills now an essential focus of IT recruitment. CIO Magazine, 2015.

[9] Goldberg, J. & Howe, S. (2014). Virtual Capstone Design Teams: Preparing for Global Innovation. *Proc. Capstone Design Conf.*, Columbus, Ohio, June 2014.

[10] Redacted to preserve anonymity.

[11] Turnitin, Wikipedia, https://en.wikipedia.org/wiki/Turnitin, accessed 2017.

[12] Virtual Team, Wikipedia, https://en.wikipedia.org/wiki/Virtual_team, accessed 2017.

[13] Wilkins, D.E. & Lawhead, P.B. (2000). Evaluating Individuals in Team Projects. *Proceedings of the 31st SIGCSE Technology Symposium on CS Education*, Austin, Texas, 172-175.