Achieving Effective Communication between Diverse Disciplines in Small Teams

Robert Pastel
Computer Science Department
Michigan Technological University
Houghton, MI, USA
pastel@mtu.edu

Wei Zhang
Cognitive and Learning Sciences Department
Michigan Technological University
Houghton, MI, USA
wzhang5@mtu.edu

***Abstract*—This "Full Paper" in the "Innovative Practice Category" describes a 3 year study of communication and collaboration within small teams composed of CS and Communication students developing citizen science applications for sponsoring scientists. To evaluate the communication within the teams, an end of the semester survey asked students to rate the quality of their communication within their own discipline and with the other discipline. Our goal is that students should rate the quality of communication equally. Our method for improving the collaboration uses a variety of ethnographic techniques to discover the likely influences and barriers to team collaboration and communication. We then modify the collaboration of the classes to mollify the barriers. After three years of collaboration, we learned that classes meeting at the same time in adjacent classrooms, so as to create time and place for teams to meet, were not sufficient to improve communication. For the collaboration within the team and the communication quality to reach parity between the disciplines, the instructors need to clearly demonstrate shared goals for the courses and create a common language by giving multiple joint lectures.**

Keywords—HCI, multidisciplinary team learning

I. INTRODUCTION

Communicating in diverse contexts is an essential skill for software engineers. Throughout the development process, software engineers must collaborate, communicate and negotiate with a variety of stakeholders. They must elicit requirements from clients, communicate system constraints and negotiate solutions with clients and program managers. Within the programming team, they must collaborate on the design of the system, and then schedule and task the project. The communication is multi-tiered and challenging: it encompasses team communication, client communication, as well as broader public communication.

Learning to communicate and collaborate with team members from various disciplines is difficult for CS students. Their lower level classes focus on technology and programming languages. Typical programming assignments require CS students to make an individual effort to learn specifics and tricks of the language or system. The variety of majors that CS students have to interact with in their CS classes is narrow. The consequence of this academic experience is that the typical CS graduate prefers to work in isolation (without a team), can speak only using technical languages and is not inclusive or accepting of others' ideas.

CS students need experience communicating with a variety of stakeholders in a project during software development. We created these experiences from a collaboration of courses from three disciplines: a CS undergraduate course in UI Design and Implementation, an undergraduate Communication (COMM) course in Usability and Instruction Writing, and a CS graduate course cross-listed with Psychology in Human Computer Interaction (HCI). Students from the CS and COMM undergraduate courses formed small teams to design and implement mobile citizen science applications (apps) for sponsoring scientists. One or two graduate students work with each team as user experience consultants.

We used a variety of ethnographic techniques to study and evaluate the communication between the CS and COMM undergraduate students. In particular, the end of semester survey had two questions asking the students to rate the quality of communication with students on the team within their same discipline, and with students from the other discipline. Our goal is that the CS students should have equal quality of communication with the COMM students in their teams as with the CS students within their teams. If we did not achieve our goal then for the following year we modified the coordination between the courses attempting to achieve parity in the communication.

Although communication is only one component of the collaboration within the teams, we focus on communication because it is critical to modern agile software development [1], [2]. We focus on the communication between the CS and COMM students within the teams, in particular, because they worked together intimately, and yet their communication is challenged because of their differences in education, language and communication style.

We used self-evaluation of the communication quality because we feel that the team members engaged in the communication are the best judge of the communication quality. We focused on the end of the semester evaluation of the communication because we assumed that the teams would

978-1-5386-1174-6/18/$31.00 ©2018 IEEE

learn to communicate during the semester, and we hoped that by the end of the semester the teams would achieve effective communication.

II. Related Work

Many attributes of the collaborating classes influence the communication within the teams, such as the coordination of the assignments and the organization of the teams. Consequently, we briefly surveyed the CS education literature for techniques to manage multidisciplinary-team-based project courses. There is a fair amount of literature describing multidisciplinary collaboration between CS and engineering students [3] - [5], but we focused primarily on the collaboration between more diverse disciplines because we believe that the challenges are greater in teams composed of diverse disciplines. Reviewing the literature, we found collaborations between CS students and art majors [6] - [8], business majors [9] - [13], a variety of majors in interdisciplinary gaming courses [14] - [16], and graphic designers [17]. Our survey uncovered many recommendations for managing multidisciplinary teams that we divide into four categories: team organization and project ownership, project scheduling and coordination, team support and education, and team communication.

A. Team Organization and Project Ownership

Generally, teams are formed early in the semester [4], [9], [12]. In a collaboration between CS and graphic design courses, the instructors regret forming the teams until mid-semester after the CS students had decided on the theme for the applications [17]. In all cases, students from the different disciplines were evenly distributed across the teams, but generally the membership was not balanced between the disciplines. Teams can be formed based on student skills [3], availability [4] or personal interests [12].

In structured courses, teams can work on smaller projects before starting the larger project [6]. Teams can also build team websites as an initial project to get to know each other [9]. A collaboration across multiple universities can have "meet-and-greet" parties at restaurants so that the students can become acquainted [10].

In collaborations between CS and Business majors, the business students typically assume the role of clients to the CS students [9], [13]. In multidisciplinary capstone courses, the instructor can assume the role of product owner [3]. But in general, multidisciplinary teams self-form, and team structure is informal. Consequently, who makes the decisions within the team is not clear and can lead to conflicts within the team. In a collaboration between CS and art students, the CS students can feel like "code slaves" while the art students can feel like "art slaves" [12]. In a course with CS, Digital Media, and Education majors, the CS students displayed "technical arrogance" towards the education students [15]. The instructors believed that the arrogance was due to CS students' lack of appreciation for the educational aspects of the game. In the collaboration between CS and graphic design courses, the CS students were dissatisfied with the graphic designs while the graphic design students were dissatisfied with the implementation of their designs [17]. The instructors explained the conflict as a lack of project ownership by the graphic design students because they were brought into the project late, and their lack of appreciation of the challenges faced by the CS students.

B. Project Scheduling and Coordination

The assignments for the different disciplines should be balanced and well-coordinated. In the course with CS, Digital Media and Education majors, the tasks for the different disciplines were well-defined and were adjusted during the semester [15]. The collaboration between CS and graphic design students suffered because of a mismatch in the deadlines between the disciplines [17]. The instructors believe that better synchronization of deliverables would have helped the disciplines focus on the same thing at the same time. In an animation course with CS and art students, initial assignments had sections that were easy and hard for the different disciplines, so the disciplines could help each other [6]. Students in an interdisciplinary robotics course recommended that project work should be equal between the disciplines [4].

C. Team Support and Education

Teams need instruction on collaboration and team management. The instruction can be formal lectures on teamwork early in the course [3], [6], [4]. In addition, instructors can meet regularly with the teams to identify problems and provide support either weekly [3] or biweekly [6]. At a minimum, instructors should track teams and resolve problems early [17].

D. Team Communication

Effective communication requires that the disciplines share a common language. CS and art students in a virtual reality course were required to attend all lectures even if the lecture was technically geared to the CS students [7]. The art students did not enjoy learning the details of the technology, but the instructors felt exposing the art students to the technical challenges helped the team. The subject can be presented breadth-first, so that all disciplines can comprehend the topics [12]. In addition, instructors from the different disciplines can share the lecture, presenting on their disciplinary topic [6].

Teams use a variety of media to communicate. Face-to-face communication requires a place. Scheduling classes at the same time can offer teams opportunity to meet during class and make joint presentations [17]. A dedicated room for the team can become a "real workplace environment" [14]. Teams also need digital communication such as an email list, especially distributive teams which could also use video conferencing [10]. Status reports between the disciplines can also be used as a communication medium [9].

Teams also need time to communicate and make decisions [14]. Although meetings are effective for brainstorming and avoiding misunderstanding, teams do not necessarily schedule regular meetings [17]. Regular meetings can be encouraged by making assignments essentially meeting minutes [12] or by requiring brainstorming.

E. Summary

As a pedagogical aid, we summarized all the recommendations from the instructors teaching multidisciplinary-team-based courses in Table I. Given the circumstances of the multidisciplinary course, it may not be possible to adhere to all the recommendations.

TABLE I. DIVERSE TEAM PEDAGOGICAL RECOMMENDATIONS

Team Organization and Project Ownership
— Form teams from all disciplines at the beginning of the project.
— Consider skills, personalities, and availability while creating teams.
— Organize teams and ensure a team leader.
— Ensure members value each other's contributions.
Project Scheduling and Coordination
— Synchronizing assignments and project deadlines.
— Ensure balanced workloads across the disciplines.
Team Support and Education
— Educate students early about teamwork.
— Meet regularly with students to provide support.
— Watch for conflicts.
Team Communication
— Create a common language by sharing course material and lectures.
— Schedule collaborating courses at the same time.
— Use face-to-face meetings, joint presentations, and electronic media.
— Schedule regular meetings to ensure regular communication.

III. METHOD

Our method for developing the collaboration between the classes is iterative. After coordinating the collaboration, we used an end of the semester survey to evaluate the communication within the teams. If the results of the survey indicated that the communication within the teams were not as effective as it could be then an analysis of the other questions in the survey were used to determine the possible barriers to communication and how to modify the following year collaboration between the classes.

The end of the semester survey had questions that probed many aspects of the courses and the collaboration within the teams. In particular, two questions asked the students to rate the communication quality with team members in their same discipline, and with team members from the other discipline using a 7-point scale. The scale was anchored at 1 to "very poor" and at 7 to "very good". Responses to this question were used as the measure of communication quality. If the students did not rate the communication quality with the other discipline equal to the communication within their discipline, we assumed that the communication was not as effective as possible.

The survey also had two opened ended questions asking the students about their experience collaborating with the other discipline and suggestions for improving the courses. We used the responses from opened-ended questions in the survey and interviews with project stakeholders to gain insights to potential barriers to the communication.

In addition, the second and third iteration of the survey had several choice questions asking students to select what aspects of the app they were satisfied with and dissatisfied with. (The choices are delineated in fig 2 & 3 showing the results.) Another choice question asked students to select what influences there was on their project. (The choices are delineated in Table IV showing the results.) We used responses from the choice questions to detect differences in the mental model about the development process and the goals for the project between the CS and COMM students.

A Student's t-test is used to determine significance between the differences in the communication rating within the discipline and with the other discipline for the same student group. Welch's t-test, which does not assume equal variance, is used to determine significance across disciplines. Fisher's exact test is used to determine significance of counts in contingency tables because the small number of COMM students led to low counts in the tables. The 95% confidence intervals for the proportions of choices is obtained by inverting the likelihood ratio test because it is robust to proportions close to zero. Students' responses to the open-ended question are categorized and reported as the number of responses by category.

Participation in the undergraduate courses ranged from 21 to 28 students in the CS course and 10 to 12 in the COMM course. Survey compliance varied from 71% to 92%. The study was approved by our Internal Review Board and students gave consent.

IV. INITIAL COLLABORATION

A. Original Courses

All three courses existed prior to the collaboration. The undergraduate CS UI course and the graduate Human-Computer Interaction course were taught by the same instructor at the same time. The collaboration between the undergraduate and graduate CS courses is described in detail in [18], [19]. Briefly, the CS undergraduate students formed teams that were responsible for the design and implementation of an app of their own choosing. One or two graduate students were assigned to each team and assumed the role of user experience consultant. The roles of the undergraduate and graduate students were established by carefully coordinating the assignments for each course. The courses concluded with the graduate students conducting usability tests on the apps with the assistance of CS students. The collaboration was very effective at teaching the need for user-centered design and illustrating that users do not have the same perspective of the app as programmers. Ultimately, the students did not learn the subtleties of designing for others because the teams would conceive and design apps for themselves.

The teams needed app projects from clients. When an opportunity arose to use scientists sponsoring citizen science apps as clients, the undergraduate UI course was modified so that the teams could meet with the scientists to gather requirements and get feedback from the scientists after presenting designs. Soon after adding citizen science app

TABLE II. CS AND COMM SCDHEDULE OF ACTIVITIES

Week	CS Activities	COMM Activities
1	Form teams and assigned app	
2	Meet with scientists and gather requirements	
3 4	Design app, make team website and paper prototype	
5	Cognitive walkthrough and present paper prototype	
6	Feedback from scientists and graduate students	
7	Feedback from CS instructor	App promotional paper Finish instructional content
8 9	App implementation	
10		App promotional presentation
11	Final design presentation	
12	Feedback from scientist	
13	App usability testing	Instructional usability testing
14	App and instructional usability test results presentations	
Finals	Feedback from scientist	

development to the course, it was clear that the apps needed more content, specifically instructional content, and the CS students needed more help designing the apps. The Humanities professor suggested that the COMM course in Usability and Instruction Writing should join the collaboration. The course material and the structure of the Usability and Instruction Writing COMM course meshed well with the UI and HCI courses. Prior to the collaboration, students in the COMM course wrote documentation for clients and tested the documentation for usability at the end of the semester. In addition, the COMM course used the same textbook as the CS course. The COMM students could join the CS team assuming the role of content creator, assisting with the design of the app, and responsible for the instructional content.

B. Coordination of CS and COMM Classes

The coordination between the CS and COMM courses is described in detail in [20]. The first year of the collaboration was 2013, Year 1. Briefly, Teams self-formed and were assigned projects by the end of the first week. The teams had their first meetings with the scientists at the end of the second week to gather requirements for their app. During the third and fourth weeks, CS and COMM students worked together designing the app, making a paper prototype, and making the team website. During the evenings of the fifth week, teams conducted cognitive walkthroughs presenting their paper prototype, which were recorded for the remote scientists. Teams got feedback from the scientists, and graduate students during the sixth week. The CS students got feedback and implementation goals from their CS instructor during the seventh week, while the COMM students begin developing a promotional for the app. From the eighth to the eleventh week, the CS and COMM students worked separately; CS students implementing the app, and the COMM students writing the app promotional and finishing the instructional content for the app. During the evenings of the eleventh week, teams met to present their final design of their app, which was a partial implementation and included some instructional content. Usability tests were conducted in the thirteenth week separately by graduate students and the COMM students and the results presented in the fourteenth week. During finals week, teams would have their final meeting with the scientists to discuss the results from the usability tests. The schedule of the CS and COMM students' activities is outlined in Table II. Rows in Table II that span the CS and COMM columns without a boundary represent collaborative work between CS and COMM students, while rows with a boundary between the CS and COMM column's represent CS and COMM students working separately. Broadly, the CS and COMM students worked closely together during the first six weeks of the semester and then worked separately on their own part of the project until the eleventh week of the semester when the final design presentations were given. The course was faced paced and the schedule had to be strictly adhered to for the apps and instructional content to be ready for usability testing by the thirteenth week of the semester.

We followed many of the pedagogical recommendations in Table 1. The undergraduate teams were organized in the first week of the semester, composed of five CS students and two COMM students. We assumed that the teams had skills to self-organize because both the CS and COMM students had prior experience working in teams in their respective disciplines. We also assumed that both CS and COMM students would have project ownership because they both contributed to the design of the apps, and the CS and COMM students would share a common language because the course material was similar between the two courses. We also felt that making a team website during the third week would help with team consolidation and communication. Regrettably, the CS course and COMM course could not meet at the same time because the courses had already been scheduled. Consequently, the CS and COMM courses did not have any shared lectures. The teams did meet in the evening to present the paper prototype during the fifth week and a nearly completed app during the eleventh week.

C. First Year Results

Table III shows the communication ratings of the CS and COMM students. For Year 1, both CS and COMM students' ratings for the quality of communication within their discipline is significantly higher than with the other discipline, p-value = .01. This result is contrary to achieving an effective interdisciplinary team. An effective interdisciplinary team should have equal quality of communication between all teammates. There is no significance difference between the CS and COMM students' ratings for within the discipline [$t(10.8) = 0.89$, p-value = .38] or for with the other discipline [$t(13.2) = 5.9$, p-value = .56], which suggest that the CS and COMM students are using the same scale for rating the communication quality and that the lower communication quality with the other discipline is similar for both CS and COMM students. Responses to the open-ended question, "What suggestions do you have for improving the course?" also indicate that the communication between the disciplines could be improved. Half of the CS students' suggestions for improving the course and all of the COMM students suggestions were to have better communication between the disciplines and better coordination of assignments. We believed that the CS and COMM students not having the time to meet during the class hour was the major cause of both CS and COMM students rating the quality of

communication with the other discipline lower than within their own discipline.

D. Second Year Modifications

Three modifications were made to the second year of the collaboration between the CS and COMM courses. First, to allow more time for communication, the classes were scheduled at the same time in adjacent classrooms so that the CS and COMM students could meet during the last 10 minutes of class. Second, to educate the teams, two joint lectures were given to all students about team formation and dynamics during the first week of the semester. After concluding the lectures, the teams wrote a team charter and selected a team leader. These were the only joint lectures during the semester. Third, to ensure shared goals, the CS and COMM students shared the same assignment descriptions through the fourth week that focused on designing the apps rather than separate but parallel assignment descriptions for each class.

V. Second Year Results

Again, both the CS and COMM students' ratings of communication within their own discipline is significantly higher than the communication with the other discipline on the team. See Year 2 in Table III. There is no significance difference between the CS and COMM students' ratings within the discipline [$t(12.3) = 0.0$, p-value = 1] or with the other discipline [$t(16.1) = 0.12$, p-value = .91]. This is consistent with CS and COMM students using the same rating scale and rating the lower quality of communication with the other discipline the same.

This survey included an open-ended question asking the students for their experience with the other discipline on the team. Twenty-one CS students expressed a positive or negative experience working with the COMM students. Twelve CS students expressed positive experiences, while 9 CS students expressed negative experiences due to frustration with late or no work from the COMM students. Some CS students were very verbal saying that the COMM "students were not necessary" and "not sure what they did." Six of the 10 COMM students expressed a definite positive or negative experience with the CS students. Five COMM students expressed a positive experience, one COMM student expressed a negative experience saying, "no communication, lack of responsibility, [and] lots of procrastination." See Fig. 1 for the bar plot of the proportions of positive and negative experiences from CS and COMM students and the contingency table of counts. The estimated odds ratio 3.9 is not significant with p-value = 0.25. We believe the insignificance in the contingency table is due to the small sample size of the COMM students.

TABLE IV. Year 2 Influences on Development Process

Influences	CS	COMM
Communication with Client	15 (54%)	2 (20%)
Team collaboration	14 (50%)	3 (30%)
CS team members' skills	15 (54%)	4 (40%)
COMM team members' skills	3 (11%)	1 (10%)
Time to complete the app	9 (32%)	1 (10%)
Team's members' personalities	4 (14%)	1 (10%)

TABLE III. Communication Ratings

Year		Within discipline (*mean* ± *sd*)	With other discipline (*mean* ± *sd*)	*t*-value	*df*	*p*-value
1	CS n=19	5.3±1.1	4.1±1.6	2.70	29	.01*
	COMM n=8	5.8±1.4	3.7±1.6	2.93	14	.01*
2	CS n=27	5.8±1.1	4.3±1.6	3.81	51	.0004*
	COMM n=10	5.8±1.6	4.0±2.1	2.10	17	.04*
3	CS n=23	5.1±1.4	5.7±1.2	-1.58	43	.12
	COMM n=9	5.7±1.7	5.7±0.7	0	11	1

Thirteen of the 28 CS students' suggestions for improving the course expressed difficulty with Android programming and needing more time for implementing the app. Four of the CS students addressed the collaboration between CS and COMM students suggesting either more or less collaboration with the COMM students. Nine out of 10 COMM students' suggestions for improving the course was better coordination between the classes and better communication with the CS students. One COMM student said "the teachers need to collaborate together and be on the same page."

Anticipating the possibility that the collaboration might not improve, we added questions to the second year survey to study the mental model of the CS and COMM students. One question asked, "What influenced your app development?" Students could choose more than one of the following: Collaboration within your team, Communication with your scientist, Documents from your scientist, CS team member's skills, COMM team members' skills, Team members personalities, Time to implement your app, and Instructional content. Table IV tallies the choices of the CS and COMM students. The percentages are the percent of students from the class that chose that influence. Both CS and COMM students agree that team collaboration and CS team members' skills influenced app development. Particularly noteworthy is only 11% of the CS and 10% of the COMM students felt that the COMM student's skills influenced the app development. Many, (32%) CS students felt that the time for implementation was an influence on the development of the app.

Two questions on the survey asked the students what aspects of the completed app satisfied or dissatisfied them. The

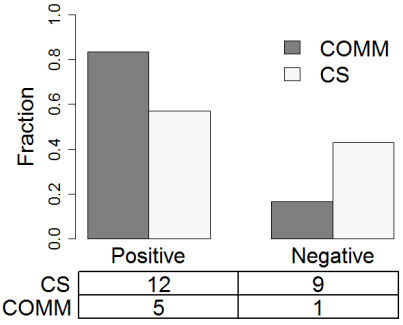


Fig. 1. Bar plot of proportions for Year 2 CS and COMM students expressions of positive or negative experience with the other discipline and counts in the contingency table.

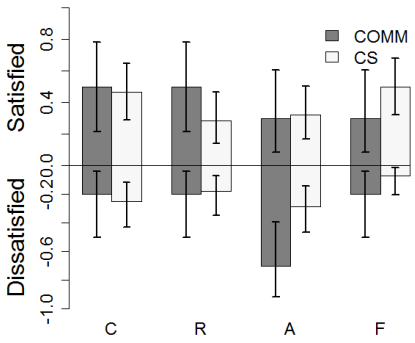


Fig. 2. Year 2 CS and COMM students' satisfied and dissatisfied aspects of the app. Aspects and keys are:

C = Completeness of the app
R = Covering all scientists requirements
A = Appearance of the app
F = Bug-free correct functioning app

Error bars represent length of the 95% confidence intervals by likely ratio test.

choices of the aspects were: the completeness of the app, covering all the scientist's requirements, the appearance of the app, and "bug-free" correct functionality of the app. Students could choose more than one aspects. Fig. 2 plots the fraction of students from the CS and COMM classes that were satisfied, positive extent of the bar, and dissatisfied, negative extent of the bar, for the aspect. A student feeling either satisfied or dissatisfied represents a concern for that aspect of the app, so we can interpret the length of the bar as the collective amount of concern for that aspect. The shift of the bar to positive or negative from a balanced neutral position, *i.e.* equal satisfied and dissatisfied lengths, indicates the collective satisfaction or dissatisfaction with that aspect. Generally, the CS and COMM students had the same collective concern for C: the completeness of the app, R: covering all the scientists' requirements for the app, and F: the "bug-free" correct functionality of the app. And generally, both CS and COMM students were collectively satisfied with those aspects. COMM students collectively had more concern about A: the appearance of the app than the CS students and collectively the COMM students were dissatisfied with the appearance.

To get insights into the dynamics of the team collaboration, we interviewed one graduate student assigned to each team to discuss the collaboration between the CS and COMM students. The graduate students were impartial observers of the teams as they were consultants and knew the teams well because they attended the team meetings at the end of the lectures. They reported that at the beginning of the semester the collaboration was strong for all teams, but in four of the six teams collaboration had deteriorated by the end of the semester. The causes varied from CS students failure to complete the app to COMM students not finishing the instructional design in time for them to be incorporated into the app. During the interviews with the graduate students, we tried to determine the time that the deterioration occurred. While the CS and COMM students were working together on the paper prototype the collaboration was strong, but after the presentation of the paper prototype, the collaboration began to deteriorate. At this time, CS and COMM students stopped sharing assignments. The CS students began implementing the app, and the COMM students started writing a promotional for the app.

VI. Third Year Collaboration

A. Discussion of Second Year Results

The CS and COMM classes meeting at the same time in adjacent classrooms did not improve the communication quality between the disciplines. It did increase the communication frequency between the disciplines [21]. Although all the teams began the semester with good collaboration between the disciplines, several teams' collaboration deteriorated by the end of the semester. The differences in the CS and COMM students' concerns about the app could help explain some of the conflicts between the disciplines. Although both the CS and COMM students are equally concerned about completing an app that covered all the scientist's requirements correctly, the COMM students were more concerned about the appearance of the app than the CS students, and they were dissatisfied with the appearance. CS students weren't interested in the appearance, leading them to see little value to the COMM students' contributions. In the words of one COMM student, "The COMM students wanted the help documentation implemented and the CS students didn't see that as applicable." CS students not valuing the work of the COMM students would lead to conflict.

Another possible cause of the deteriorating collaboration can be learned from the differences in suggestions for improving the course. Many of the CS students suggestions addressed Android programming and the time allowed to implement the app. The CS students felt the challenge of completing the apps, while none of the COMM students felt that challenge. This difference is also reflected in the CS and COMM students' choices for what influenced the development process. After the successful face-to-face collaboration with the COMM students making the paper prototype, the CS students had only a month to implement the app designs in a platform that was new for many of the CS students. The time pressure probably contributed to the breakdown in communication between the disciplines. A team of CS students faced with a project to complete in a short time span will change their collaboration technique to a divided collaboration where team members work in isolation coding a component of the app. If this change in the collaboration method was not understood by the COMM students, they could feel left out of the project.

An agreement between the CS and COMM students' for what influenced the development process could have also led to the deterioration of the collaboration. CS and COMM students agreed that CS students' skills had more influence on the development process then the COMM students' skills. In other words, the CS students were more valuable than the COMM students. The app would not get made without the CS students' programming skills. The CS students decided what got implemented, so in effect, the CS students owned the project. While designing the paper prototype, the COMM students probably had ownership of the project because the CS students do not have much experience with paper design. Some COMM students may not have felt good about losing ownership of the project.

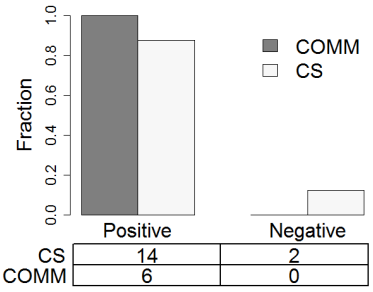


Fig. 3. Bar plot of proportions for Year 3 CS and COMM students expressions of positive or negative experience with the other discipline and counts in the contingency table.

COMM students feeling loss of project ownership, left out of the implementation of the app, or that their contribution was not valued by the CS students would lead them to having negative feelings about the collaboration. In addition the assignment description for the final version of the instructional content did not have a definite due date. So when the CS students felt the time pressure increasing to finish the apps before the usability testing and asked for the final version of the instructional content, the COMM students would not feel the same urgency to respond. Non-responsive COMM students could frustrate the CS students.

A solution to this spiraling deterioration of the collaboration is for the COMM students to be engaged in essential aspects of the app throughout the implementation. COMM students would then not be left out of the implementation of the app, could maintain project ownership, and not be undervalued by the CS students. This approach was not adapted. For the third year of the collaboration of the classes, a new Humanities instructor taught the COMM course. This Humanities instructor felt that an essential component of the COMM course was an independent project for each COMM student, and the natural time to schedule the independent projects was while the CS students were implementing the app.

B. Third Year Modifications

The CS and Humanities instructor met many times prior to the third-year collaboration. The Humanities and CS instructors wanted full agreement on the collaboration and the schedule of assignments. Four major modification were made to the collaboration. First, to ensure shared language and goals, the instructors gave many joint lectures. For the first five weeks leading to the paper prototype presentation, all three classes met in one room, and the lectures were given by both instructors. Each instructor gave their perspective of the topic. Second, to assure that the instructional content was finished in time by the COMM students, the assignment description for the final version of the instructions had a definite due date. Third, to help alleviate the time pressure that the CS students felt implementing the app, the platform was changed from Android to web, in order to take advantage of the scaffolding offered by the web framework. Fourth, to increase team coordination and help ensure project ownership, each team had two leaders, a CS technical lead and a COMM product owner.

VII. THIRD YEAR RESULTS

In the third year of the collaboration, the communication ratings between the CS and COMM students reached parity. See Year 3 row in Table III. COMM students rated the quality of communication with the CS as good as with their fellow COMM students. CS students rated the quality of communication with COMM students as being slightly better as the communication with their CS teammates (although not significant, p-value = .12). There is no significance difference between the CS and COMM students' ratings within the discipline [$t(12.5) = -0.94$, p-value = .36] or with the other discipline [$t(25) = 0.0$, p-value = 1]. This is consistent with CS and COMM students using the same rating scale.

Twenty-one CS students expressed a definite feeling about their experiences with their COMM teammates in the open-ended question, 14 were positive, and only two were negative complaining about late work from the COMM students. All six COMM students that expressed a feeling about their experiences working with the CS teammates were positive. Fig. 3 summarizes the percentages of positive and negative experiences with respect to the discipline and the contingency table for the third year of the collaboration. The CS and COMM students' positive and negative percentages appear similar, and Fisher's exact test estimates a 0 odds ratio with p-value = 1.

Three of the 9 COMM students' suggestions for improving the courses were for more time for collaboration with the CS students and two COMM students suggested better apps or app descriptions. Of the 23 CS students suggestions for improving the courses, seven CS students complained about the web framework, and seven CS students either wanted more time for implementation or help with the implementation. Changing from Android app development to web app development did not relive the CS students' time pressure or anxiety about completing the apps.

Year 3 CS and COMM students satisfaction or dissatisfaction with aspects of the completed app is shown in Fig. 4. CS and COMM student have about the same amount of concerns for the completeness of the app, C, and covering all

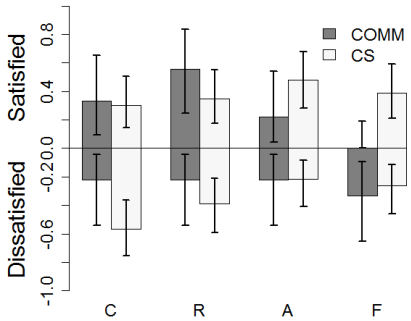


Fig. 4. Year 3 CS and COMM students satisfied and dissatisfied aspects of the app. Aspects and keys are:

C = Completeness of the app
R = Covering all scientists requirements
A = Appearance of the app
F = Bug-free correct functioning app

the requirements, R. This year, CS students have more concerns about the appearance of the app, A, than COMM students, but collectively CS are satisfied with the appearance while COMM students are neutral. CS students are also more concerned about "bug free" correct functionality, F, then COMM students, but CS satisfaction and dissatisfaction is collectively equal. COMM students only express dissatisfaction with the correct functioning of the app.

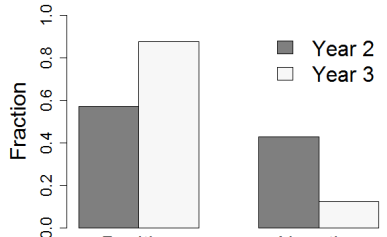
VIII. Discussion and Conclusion

A. Comparison of Year 2 and Year 3

The CS and COMM students' ratings on the quality of communication with the other discipline is 5.7, which is significantly higher than the neutral rating 4 [for CS students $t(22) = 6.8$, p-value = 7.9e-07 and for COMM students $t(8) = 7.3$, p-value = 8.5e-05]. They improved from the previous two years, which were approximately neutral ratings. In addition, CS students had more positive experience with the COMM students during Year 3. Comparing CS students' percentages of positive and negative experiences across Year 2 and Year 3 (see Fig. 5), there is a significant improvement in the CS students' experiences with the COMM students [Fisher's one sided exact test p-value = .0482 and odds ratio of 5.025]. Suggestions for improving the course for Years 1 and 2 were negative about the collaboration. For Year 3, neither CS nor COMM students were suggesting better coordination of the courses, nor better collaboration between the instructors. CS students were no longer saying that the COMM students were not necessary, and the COMM students did not express that their contributions were not valued. The COMM students only wanted more time to collaborate with the CS students. In general, the team collaboration and communication improved the third year of the collaboration.

B. Discussion

Of the four major modifications to the collaboration between the classes, changing the development platform probably did not contribute to the improved quality of communication with the other discipline because more than half of the CS students complained about using the web platform or the time for implementation. It is not clear that having two team leaders, one CS student and COMM students, contributed to improving the COMM students' project



	Positive	Negative
Year 3	14	2
Year 2	12	9

Fig. 5. Bar plot of proportions for Years 2 & 3 CS students expressions of positive or negative experience with the CS students and counts in the contingency table.

ownership or communication with the CS students. Scouring all the open-ended questions no COMM student mentioned team leadership, but four CS students mentioned wanting more meetings and better coordination of the work which are responsibilities of the team leader. The CS students had the challenging task of managing all the components and programming the app so team leadership was critical to them during implementation.

It is clear that the COMM students did not have to remain directly engaged with the projects during implementation of the apps in order to sustain team communication and collaboration. The COMM students could be engaged in another project during implementation of the app as long as the due dates for the deliverables were clear. Whether the eight joint lectures by the CS and Humanities instructors at the beginning of the semester created shared goals and a common language between the CS and COMM students is hard to definitively prove. The shared presentations did change the dynamics of the classroom. Both CS and COMM students could and did join the discussions on the topics of the lectures. This open communication between the disciplines probably helped to create a shared language between the CS and COMM students. The time that the CS and Humanities instructors spent coordinating the courses and scheduling the assignments did create shared goals between the instructors and enable them to set the proper due date for the COMM students' deliverables. The instructors giving joint lectures was another manifestation of the shared goals for the courses that were clearly perceived by both the CS and COMM students.

C. Conclusion

We believe that definite due dates for deliverables and the clearly perceived shared goals between the instructors of the collaborating classes made the biggest contributions to sustaining the teams' communication and collaboration. The CS and COMM students would not have been able to clearly perceive the instructors shared goals without the joint lectures, which was made possible by scheduling the courses at the same time in adjacent classrooms. The classes meeting in adjacent classroom also allowed the teams to meet regularly and build their collaboration. But space and time for teams to meet is not sufficient to build strong collaboration. We believe that the students need to perceive directly the collaboration and shared goals between the professors so barriers between the disciplines do not form. To the list of Pedagogical Recommendations for Diverse Teams, we add and emphasize the recommendations in Table V.

TABLE V. Additional Pedagogical Recommendations

Team Organization and Project Ownership
— Clearly perceived shared goals between the instructors
— Use joint lectures to create shared goals

Project Scheduling and Coordination
— Adjacent classrooms to ensure regular team meetings
— Clear due dates for deliverables between the disciplines

Team Communication
— Use joint lectures to create a common language

REFERENCES

[1] N. J. Cooke, E. Salas, P. A. Kiekel, & B. Bell, Advances in Measuring Team Cognition. Team Cognition: Understanding the Factors that Drive Process and Performance, 2004, 83-106.

[2] E. Salas, N. J. Cooke, & M. A. Rosen, "On Teams, Teamwork, and Team Performance: Discoveries and Developments" in Human Factors: The Journal of the Human Factors and Ergonomics Society, 2008, 50 (3), 540-547.

[3] R. Miller and B. Olds, "A model curriculum for a capstone course in multidisciplinary engineering design", Journal of Engineering Education, 1994, vol. 83, no. 4, pp. 311–316.

[4] J. B. Weinberg, W. W. White, C. Karacal, G. Engel, and A.-P. Hu, "Multidisciplinary teamwork in a robotics course" in Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education (SIGCSE'05). ACM, New York, NY, 2005, pp. 446–450. DOI = 10.1145/1047344.1047488 http://doi.acm.org/10.1145/1047344.1047488.

[5] M. Lukowiak, S. Radziszowski, J. Vallino, and C. Wood, "Cybersecurity Education: Bridging the gap between hardware and software domains. Transactions on Computing Education, 2014, vol. 14, no. 1, Article 2, 20 pages. DOI = 10.1145/2538029 http://doi.acm.org/10.1145/2538029

[6] D. S. Ebert and D. Bailey, "A collaborative and interdisciplinary computer animation course", SIGGRAPH Computer Graphics, 2000, vol. 34, no. 3, pp. 22–26. DOI = 10.1145/359293.359325 http://doi.acm.org/10.1145/359293.359325

[7] G. W. Zimmerman and D. E. Eber, "When worlds collide!: An interdisciplinary course in virtual reality art", in Proceedings of the 32nd SIGCSE Technical Symposium on Computer Science Education (SIGCSE'01). ACM, New York, NY, 2001, pp. 75–79. DOI = 10.1145/364447.364545 http://doi.acm.org/10.1145/364447.364545

[8] J. McDonald and R. Wolfe, "Using computer graphics to foster interdisciplinary collaboration in capstone courses", Journal of Computing Sciences in Colleges, 2008, vol. 24, no. 1, pp. 83–90.

[9] K. Anewalt, Utilizing interdisciplinary teams in teaching e-commerce. Journal of Computing Sciences in Colleges, vol. 19, no. 2, pp. 288–296, 2003

[10] L. J. Burnell, J. W. Priest, and J. R. Durrett. 2003. Assessment of a resource limited process for multidisciplinary projects", SIGCSE Bulletin vol. 35, no. 4, pp. 68–71, 2003. DOI = 10.1145/960492.960530 http://doi.acm.org/10.1145/960492.960530

[11] J. T. Catanio. 2006, "An interdisciplinary practical approach to teaching the software development lifecycle", in Proceedings of the 7th Conference on Information Technology Education (SIGITE'06). ACM, New York, NY, 2006, pp. 3–8. DOI = 10.1145/1168812.1168816 http://doi.acm.org/10.1145/1168812.1168816

[12] G. Domik, "Who is on my team: Building strong teams in interdisciplinary visualization courses", in ACM SIGGRAPH ASIA 2009 Educators Program (SIGGRAPH ASIA'09). ACM, New York, NY, 2009, Article 9, 7 pages. DOI = 10.1145/1666611.1666620 http://doi.acm.org/10.1145/1666611.1666620

[13] J. Kietzmann and H. H. Tsang, "Minding the gap: Bridging computing science and business studies with an interdisciplinary innovation challenge", in Proceedings of the 15th Western Canadian Conference on Computing Education (WCCCE'10). ACM, New York, NY, 2010, Article 14, 5 pages. DOI = 10.1145/1806512.1806533 http://doi.acm.org/10.1145/1806512.1806533

[14] L. Ficocelli and D. Gregg, "A group project—with a twist!", in Proceedings of the Western Canadian Conference on Computing Education (WCCCE'05), 2005. Retrieved October 31, 2015 from http://web.unbc.ca/wccce05/html/proceedings.html. [Anonymous 2009] Details omitted for double-blind reviewing.

[15] Q. Brown, F. Lee, and S. Alejandre, "Emphasizing soft skills and team development in an educational digital game design course", in Proceedings of the 4th International Conference on Foundations of Digital Games (FDG'09). ACM, New York, NY, 2009, pp. 240–247.

[16] S. A. Kuhl, R. Pastel, R. George, C. M. Meyers, M. L. Freitag, J. M. Lund, and M. P. Stefaniak. "Teaching interdisciplinary teamwork through hands-on game development" In Proceedings of the ASEE Annual Conference, 2014.

[17] F. McCown and S. S. Gibson, "Using interdisciplinary teams in a mobile application development course", Journal of Computing Sciences in Colleges, 2013, vol. 28, no. 5, pp. 85–93.

[18] C. Brown & R. Pastel, "Combining distinct graduate and undergraduate HCI courses: an experiential and interactive approach", in Proceedings of the 40th ACM Technical Symposium on Computer Science Education (SIGCSE'09). ACM, New York, NY, 2009, pp. 392–396.

[19] R. Pastel, C. Brown, M. Woller-Carter, and S. Kumar, "Teaching human factors to graduate and undergraduate computer science students", in Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 2012, vol. 56, no. 1, pp. 595–599.

[20] R. Pastel, M. Seigel, W. Zhang and A. Mayer, "Team building in multidisciplinary client-sponsored project courses" ACM Trans. Comput. Educ. 15, 4, Article 19 (November 2015), 23

[21] W. Zhang and R. Pastel, "Communication is more than verbal: The role of clients' documents in requirement solicitation" in Proceedings of the Human Factors and Ergonomics Society Annual Meeting (HFES'14), 2014.