Exporting the Studio Model of Learning

Teaming Architecture with Computer Science

Wassim Jabi [1], Theodore Hall [2], Katia Passerini [3], Cristian Borcea [4], Quentin Jones [5]
[1-2] School of Architecture, New Jersey Institute of Technology, USA, [3] School of Management, New Jersey Institute of Technology, USA, [4-5] College of Computing Sciences, New Jersey Institute of Technology, USA
http://www.njit.edu
[1] wassim.jabi@gmail.com, [2] twhall@twhall.com, [3] pkatia@njit.edu, [4] borcea@njit.edu, [5] qjones@njit.edu

Abstract. *We have conducted a series of interdisciplinary studios that partner students in the School of Architecture with peers in the College of Computing Sciences, with two principal goals: to foster creativity in the development of information technology, and conversely, to support creativity through information technology. Our studio project focuses on ubiquitous social computing as a topic of interest to both communities that requires their collaboration to realize a physical implementation. There are administrative as well as cultural hurdles in conducting such a studio. To assess the impact of the pedagogical approach, we employed qualitative observations as well as quantitative survey data. Best results depend on achieving a degree of parity in studio experience across disciplines.*
Keywords: *interdisciplinary design studio; ubiquitous social computing; computer supported collaborative work; human computer interaction.*

Background

Maintaining innovation, in a world where globalization, outsourcing, and a networked society are prevalent, often requires a creative design approach that takes into account user needs and social customs (Castells, 2001). Typically in the professional world, small teams of creative individuals design applications in a highly iterative fashion and seek feedback on their ideas from a wide range of stakeholders. The need for innovation and creativity is also expanding to computing science education. Relevant design skills, however, are difficult to teach through traditional lecture-classroom or project-based independent study approaches. Lecture-classroom formats do not provide adequate feedback of iterative designs from peers, the user community, and experts and thus fail to emulate a collaborative design workspace. Architectural and fine arts education, on the other hand, has long employed the studio paradigm to foster creativity (Cuff, 2003). We aimed to apply the studio paradigm to achieve similar benefits in computing science. Rather than attempt this in isolation, without prior experience or a studio 'culture' in computing science to draw from, we created an interdisciplinary studio that partners students in the School of Architecture with peers in the College of Computing Sciences.

The pedagogical approach included an emphasis on user-centered, learning-by-design of various social ubiquitous computing paradigms, tools and training in interaction and interface design, blending of digital, physical, and social learning spaces across a university campus, and interdisciplinary educational interaction among undergraduate students, graduate students, and faculty to foster creativity and innovation. There were several issues in planning for this joint studio including matching the different educational cultures so that collaboration can proceed effectively, coordinating conflicting schedules, choosing a design project that stimulates mutual interest, and deploying an infrastructure that supports interdisciplinary work. Our studio project focused on ubiquitous social computing (USC) (Persson, 2001), which is a topic of interest to both communities. Rather than force all of the students to meet in the same place at the same time (which is difficult to achieve due to spatial constraints, scheduling conflicts, and divergent interests in the two academic units), it was natural to rely on ubiquitous social computing itself to provide the necessary synchronous and asynchronous communication between the studios. In particular, we used casual communication software (i.e. e-mail, instant messaging) and a custom-developed annotation system for an interactive public screen to foster interaction between the studios and seek feedback from the wider community respectively.

The objectives of this research are: (1) to investigate the benefits of interdisciplinary design-centered studio-based education on students' creative ability to identify and solve real-world problems, and (2) to create the initial specification of an experimental research and learning educational model that synthesizes an interdisciplinary approach with in-studio and out-of-studio activities using casual social interaction through ubiquitous computing technologies.

Interdisciplinary Studio

Since the studio culture in computing science is minimal, we decided to form an interdisciplinary design studio in order to take advantage of the experience accumulated in the field of architecture. In this way, both the computing science students and faculty can learn the studio culture in a very direct way through interactions with architecture students and faculty. In collaboration with architecture students, computing science students designed and implemented novel digital/physical systems and applications that took into consideration broader issues such as the relationship of technology to physical context, ergonomics, and human behavior. Furthermore, their creativity was stimulated through semester-long interdisciplinary design projects and real-world problem solving in a more interactive environment where they could freely exchange ideas.

Physical Settings

Studio settings are expected to improve the productivity and creativity of computing science students. Researchers have reported that physical settings had a direct impact on students' satisfaction and productivity (Carbone and Sheard 2002; McCoy and Evans 2002). Similarly, innovative designs of physical spaces can improve business operations and employee productivity (Horgen et al., 1999). Unfortunately, the grant supporting this project did not include provisions for modifications to the physical setting or ergonomic furniture purchases. Thus, the studio spaces remained less than ideal in their configuration (Figure 1). Despite an initial inclination to house all students in one common studio, we decided to maintain two physically separated studios on campus, one for computing science and one for architecture. The reason stems from the significant differences in the students' interests and requirements. While USC provides a good common ground, we believe that students should work not only on common interdisciplinary projects, but also on separate

Figure 1
Formal and informal design collaboration in the interdisciplinary design studio.

projects that accommodate their specialties. Another factor was an administrative hurdle: the College of Computer Science did not approve the granting of more than 3 credits to the studio course; in contrast, a design studio in a school of architecture is usually granted 6 credits.

Inter-Studio Communication

The physical separation allowed the students to rely heavily on, and thus analyze, the role of cyber infrastructure in supporting collaborative work. The degree of coupling of the two studios varied over the 3-semester research duration as well as within one semester due to the reasons outlined above. When needed, the two studios were tightly interconnected to ensure a continuous exchange of ideas between the two groups of students. We set up several formal design reviews throughout the semester, in which the two groups met in one studio and provided feedback to each other. Besides these face-to-face meetings, students were encouraged to use both synchronous and asynchronous communication tools to maintain a continuous interaction, review each other's ideas, and provide quick feedback. Casual communication software such as e-mail and instant messaging were used for asynchronous and synchronous communication (Lawson 1980, Jabi 1996). A wiki was attempted for asynchronous communication, but was not heavily used. This may be due to the architecture students' unfamiliarity with the format.

Community-Studio Interaction

Design studio students customarily seek feedback on their projects from instructors and fellow students. Feedback is given formally (e.g. periodic studio reviews and final design presentations) and informally (e.g. desk critiques and casual interaction). In many instances, however, impediments exist to soliciting and obtaining constructive feedback:

- Initial design work is done more privately and is usually unavailable to persons outside the studio.
- Periodic design reviews are usually limited to a few invited critics whose feedback may or may not prove helpful depending on many compounding factors, such as the particular interests of the critics and their personalities. Fatigue sets in during lengthy review sessions, and the quality of the real-time discussion often deteriorates towards the end.
- Other faculty and students may not have enough time to provide detailed feedback at the moment it is requested.

In order to remedy this problem, we pursued a tripartite approach to providing community feedback to studio students: (1) expert-student feedback, (2) public presentations and questionnaires, and (3) custom-developed annotation software deployed on a public interactive screen situated in public areas.

Expert-student feedback

We followed the traditional approach of conducting desk critiques and occasional formal design reviews. We also experimented with Skype™ (an instant messaging and video-conferencing application) for conducting desk critiques. The students sent their presentations to the instructor and then conducted a video-conferencing session. We found that the technology and the bandwidth have advanced to the point where the communication between the participants was seamless and natural. This is a far cry from the early days of video-conferencing experimentation where network delays and software glitches interfered with the natural flow of the conversation.

Public student presentations and survey

As part of their coursework, the students visited various locations around campus where they envisioned installing the systems they were designing (Figure 2). They installed a rear-projection screen to demonstrate their design ideas and solicited feedback on public perceptions of usability through a standardized survey. The survey grouped questions based on perceptions of project value; intention to use the technology; organizational fit such as alignment with NJIT's culture; visual appeal of the interfaces; the existence of possible alternative applications; and finally, interest and satisfaction of the respondents with the presentation held by the teams. In total, over 70 passersby stopped to review the projects. In general, respondents were highly impressed (Figure 3). The respondents also provided open-ended feedback that clustered around an excitement about the use of large interactive screen technology but highlighted the need to improve the visual appeal of the software interfaces.

Figure 2
Students from the College of Computing Sciences demonstrating their design project.

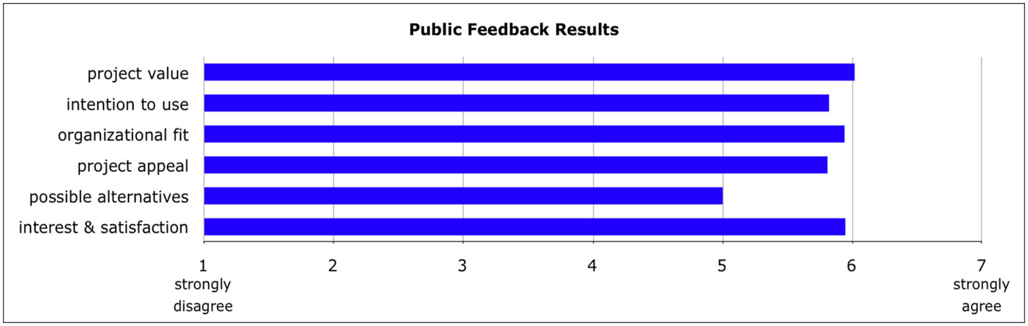

Figure 3
Public feedback results.

Annotation system for un-attended casual feedback.

We installed interactive display kiosks outside the studio in public venues to enable the students enrolled in the studios to seek feedback from the university community. Inspired by earlier work on plasma posters (Churchill et al, 2003), our interactive displays enable interested passersby to casually and anonymously leave comments regarding the designs proposed by the studio students.

The strategy we have adopted is to allow designers to submit posters as HTML web pages with minimal restrictions on their structure or layout. When a passerby chooses to annotate a poster with feedback, the application captures a bitmap 'snapshot' of its current state and uses that as a background upon which the user can sketch freely (Figure 4-A).

Each designer creates his or her poster content, uploads it to a web server, and submits its web address to the poster administrator. The administrator reviews the submission, and if there are no problems with it, adds the submission to a queue on a web server. The attributes of each poster include the poster web location, the e-mail address to which feedback should be sent, and a delay interval before the application should proceed to the next poster in the queue.

Annotation System Development and Deployment

The annotation kiosk comprises a standard personal computer equipped with a large flat-screen display and a touch-sensitive overlay (Figure 4-A). The application retrieves the poster queue from a remote server at regular intervals and displays the posters in sequence. A passerby can then interact with a poster, including scrolling, navigating through pages in a series, or finger-painting feedback. Once done, the user can either press a button to send the feedback to the author of the poster or cancel the action. The author of the poster receives feedback as a digital image attached to an e-mail message that contains the original poster with an annotation overlay (Figure 4-B).

Our first deployment was at a university 'research showcase' held in the Campus Center. This consisted of demonstrations by one of the authors, as well as supervised use by visitors. Following that event, we deployed the kiosk in the Architecture Library and left it largely unattended for a period of three weeks. Excluding tests and demonstrations by the authors, the kiosk attracted a total of 128 feedback events during this four-week period, of which 22 occurred during the 'research showcase' and the remaining 106 occurred while unattended in the library. In the library, by far the busiest days were Mondays, with the second Monday being the busiest day of all, and a precipitous drop-off in the following week (Figure

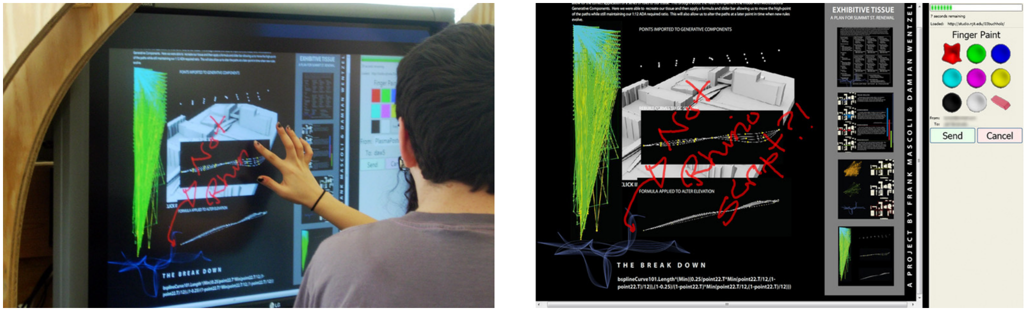

Figure 4
A: On the left, a user gives feedback using the interactive kiosk. B: On the right, a screen capture of the feedback image within the interface.

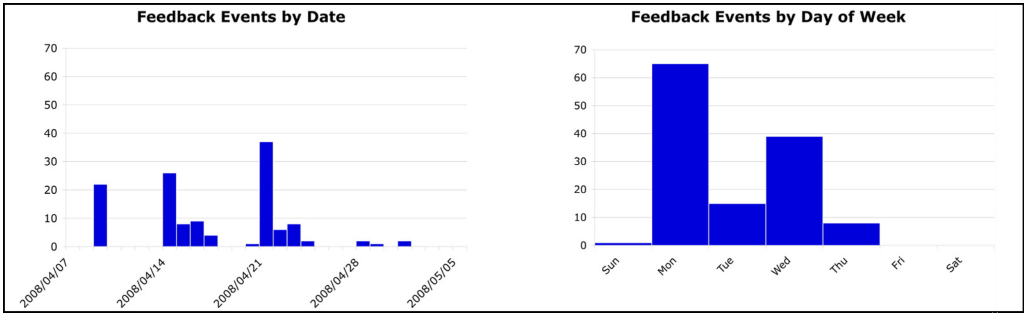

Figure 5
A: On the left, feedback events by date. B: On the right, feedback events by day of week.

5). Some students in the associated studio class encouraged their friends to try it out. We surmise that news about the kiosk spread during the first week, coupled with the novelty of new posters, led to the large number of feedback events. No new posters were added after that date. Moreover, the students were focusing on their final presentations and not taking time to update the contents of their existing posters; the time for feedback and design changes had ended. In the final week, we surmise that the loss of novelty led to the cessation of feedback.

Impact of the Interdisciplinary Studio on Creativity

In order to assess the learning and behavioral impact of the pedagogical approach on architecture and computer science (CS) students, we collected baseline data through observations, interviews and surveys. Semi-structured interviews were used to understand how students view creativity. We also collected baseline survey data on creativity-related perceptions in both groups. Our key goal was to identify (and document) whether differences in creativity among the two student populations exist, and whether our pedagogical model (bridging the studio model from architecture to computer science) increases the original creative learning outcomes. A pre-test was used to measure whether the populations of computer science and architecture students differed in their initial creativity scores. In order to answer this question, we adopted and modified survey instruments from the literature to measure creativity and cognition indexes. The surveys included both visual exercises and Likert-scale questions aimed at capturing creative behaviors, motivation, and personality differences that may facilitate innovative thinking and deliverables.

We collected baseline data during Fall 2007 and early Spring 2008. A total of 43 students participated in the baseline surveys and additional 17 students took part in an end-of Spring 2008 semester post-test evaluation. Enrolment caps determined the small sample size that studio instruction requires (typically 10-15 students).

The results from the data collection in the pre-test showed that significant differences exist among the two groups in terms of perception of creativity, originality, fluency, new idea generation, and similar constructs used to measure creative outcomes. In general, computer science students saw themselves as independent decision makers seeking limited external validation of their ideas. Architecture students perceive themselves as more creative, although they felt they might not be able to clearly express their ideas in a public forum (fluency / articulation). They also reported higher dependency on external feedback. When tasked with completing an imaginative exercise (creating shapes out of abstract forms) the architecture students outperformed their CS colleagues in terms of number of new ideas and effective use of the original undefined shapes (with an average ten versus seven new forms created by each group).

Students completed post-test evaluations at the end of Spring 2008. While differences between the two groups still exist, the frequency distribution of the data shows that the average gap has generally decreased. However, the analysis also highlights that the architecture students generally improved less in a number of post-test items. This may be related to the fact that architecture students were at plateau level in the pre-test, or to fatigue with the exercise. Perceptions of creativity and ability to create innovative artifacts increased for CS students, but in some cases decreased for architecture students. On the other hand, architecture students showed a higher level of confidence in articulating their ideas (the fluency construct). For both groups the number of creative activities students reported to be involved with decreased, which may again be related to fatigue with the exercise. Additional analysis will be completed in our future research to further articulate these results.

Challenges and Conclusions

Like any research project, this one has encountered many challenges. Some had little to do with the premise of the project, but with the administrative and educational environment at the host institution. However, the project's particular concern with interdisciplinary collaboration led to several unique issues worth presenting. First, the notion of solving problems using a design studio setting was not readily accepted or understood by those outside the realm of the field of architecture. Students and faculty in computing science and university administrators are accustomed to lecture classes that meet for only a few hours per week. Studios, on the other hand, demand that students inhabit the physical space, take ownership of their work area, and approach the environment as a semester-long work environment with no segregation of class contact hours from homework hours. Studios require different furniture that is arranged differently from a lecture room. The computing science department did not have such a space. Setting up the actual physical space and obtaining the approval and support of the administration was time consuming.

The computing science students' notion of solving problems through a design approach differed from that held by the architecture students. Computing science faculty members and students are accustomed to scenario-based methodologies that envision a particular fictitious scenario that gets acted out (Rosson and Carroll, 2001). A software/hardware solution is then designed to address this particular scenario. In architecture, on the other hand, scenarios such as described above are not usually implemented. Instead, the context of the problem is analyzed, several basic conceptual issues are delineated, and then the design progresses by deploying these conceptual ideas (Schön, 1987). Once the first iteration of the design is created, it proposes its own problems and questions to be addressed. One of the challenges we faced in this project was the skepticism we had regarding each other's design methodology. The architects insisted that they knew best how to creatively design while the computing science side insisted that their scenario-based methodology was superior. These differences in the educational culture need to be overcome for interdisciplinary work to succeed.

Another challenge involved the scale and nature of the project. Given that accreditation requirements dictate that architects should be concerned with 'buildings,' and given their training, they had an expectation of working on large-scale physical projects. Architects are neither product designers nor software engineers. Consequently, the two studios could not work on the same project semester-long. Rather, they proceeded with their own separate projects and only when the issue of deploying technology in the building came up did they find the opportunity to collaborate. The initial notion that the two studios would work on a common project semester-long has proved to be impracticable. Instead, a strategy for success may be that inter-disciplinary work should proceed more episodically. When there is a specific issue to be solved, consultants and collaborators are brought in to work together. The teams would work together when needed and diverge to work on their own projects otherwise.

As to the interactive kiosk, it proved to be a useful tool for soliciting feedback with some caveats. Given the novelty of the technology, many students treated the interactive screen as a display screen only: they would watch the content, but would not interact with it. Our observations of how the screen was used point to the need to place it in a public area, but semi-removed from a heavily traveled area so that users feel comfortable standing in front of the screen and interacting with it. Having some monitoring of the screen (either placing it in a monitored area or a publicly visible area) may reduce the chance of abuse and sabotage. The decline in user feedback after only a few weeks points to the need to continuously update the content. We did notice that animated content (e.g. YouTube and Flash videos) attracted more interest than static content, but also contributed to the perception that the screen is for viewing only. Regardless, we strongly believe that the deployment of situated displays has the potential to contribute significantly to the range and amount of feedback students receive and thus should have a positive impact on their creative problem solving skills.

Acknowledgements

This material is based in part upon work supported by the National Science Foundation IIS - ITR - CreativeIT Division. Grant Number 0714158. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Carbone, A. and Sheard, J.: 2002, A Studio-based Teaching and Learning Model, in IT: What Do First Year Students Think?, Proceedings of the 7th Annual Conference on Innovation and Technology in Computer Science Education, ACM Press, New York, pp. 213-217.

Castells, M.: 2001, The Internet Galaxy: Reflections on the Internet, Business, and Society, Oxford University Press, USA.

Churchill, E. F., Nelson, L., Denoue, L., and Girgensohn, A.: 2003, The Plasma Poster Network: Posting Multimedia Content in Public Places, Proceedings of the 9th IFIP TC13 International Conference on Human-Computer Interaction, IOS Press, Amsterdam, The Netherlands, pp. 599-606.

Cuff, D.: 2003, Architecture: The Story of Practice, MIT Press, Cambridge, MA.

Horgen, T. H., Joroff, M., Porter, W., and Schön, D. A.: 1999, Excellence by Design: Transforming Workplace and Work Practice, John Wiley & Sons, New York.

Jabi, W.: 1996, An Outline of the Requirements for a Computer-Supported Collaborative Design System, Open House International, 21(1), pp. 22-30.

Lawson, B.: 1980, How Designers Think, Architectural Press, London, UK.

Persson, P.: 2001, Social Ubiquitous Computing, CHI 2001 Workshop on Building the Ubiquitous Computing User Experience, PARC, Palo Alto, California, USA, http://www.parc.com/csl/projects/ubicomp-workshop/positionpapers/persson.pdf, accessed on July 19, 2008.

Rosson, M. B. and Carroll, J. M.: 2001, Usability Engineering: Scenario-based Development of Human-computer Interaction, Morgan Kaufmann, San Francisco, USA.

Schön, D. A.: 1987, Educating the Reflective Practitioner, Jossey-Bass, San Francisco.