Carla Viviana Coleman Cordova and Nilanjan Banerjee *University of Maryland Baltimore County*

Editors: Nilanjan Banerjee and Sami Rollins

Illustration, istockphoto.com

INTERDISCIPLINARY TEACHING STRATEGIES for Designing and Building Effective Smartphone Applications

Working together, we've created an interdisciplinary project that teaches both designers and developers the life cycle and project management of mobile application development. Working with professional clients, visual arts and computer science students, we used teamwork to tackle real-world projects with deadlines, milestones, and budget constraints. Throughout the semester, students from the Advanced Interface Design and Mobile Programming classes collaborated to apply their design and programming experience to develop smartphone applications for clients. This paper presents the insights gathered from teaching this interdisciplinary course.

Mobile applications are replacing desktop applications, and are a medium for data dissemination, user feedback, and communication in our daily lives [1, 2, 3, 4, 5]. Millions of mobile applications provide a wide range of services, such as social, education, and entertainment ones. However, while a large set of mobile applications is available on the Google Play Store [5] or Apple App Store, only a small fraction of them are adopted and become popular because they are either poorly designed or poorly implemented. The worldwide revenue of applications in 2015 is $45.37 billion and this is expected to increase to $58.21 billion in 2016 [6]. This revenue could increase even more if the design and development of mobile applications becomes more user-friendly. Specifically, mobile application developers focus on either the design of the user interface (i.e., make it engaging and aesthetically pleasing) or the efficiency of the software implementation. This fragmented model of developing mobile applications often leads to unusable or faulty applications. The applications that do succeed, however, are built through careful symbiotic design of the underlying software and the visual appeal of user interfaces. Convincing examples of such applications are the inbuilt apps on iOS that are designed with the proper consideration of artistic elements and underlying software implementation.

FRAGMENTATION IN COURSES

The culture of such fragmented design originates in college classes. For instance, classes in a visual arts department only focus on designing visually appealing interfaces without consideration of the efficiency and overhead of the associated implementation, whereas computer science classes focus on the implementation aspects without a thorough understanding of the design of the user interface. In this paper, we present an interdisciplinary teaching project that proposes to end this problem and start finding solutions by combining two complementary and popular classes at UMBC (University of Maryland Baltimore County). ART 434: Advanced Interface Design and CMSC 628: Introduction to Mobile Computing Programming. The former is a capstone class of the BFA Graphic Design degree and is provided by the Visual Arts Department, whereas the latter is a combined undergraduate and postgraduate class in the Computer Science Department. ART 434, taught by Prof. Cordova, concentrates on the visual experience of the interface in mobile and desktop applications, while CMSC 628, taught by Prof. Banerjee, provides the tools necessary to design and implement mobile applications. Students in CMSC 628 have prior programming experience in an object-oriented programming language, such as Java, but no prior smartphone programming experience. As part of this project, specific mobile development topics, such as user interface design and implementation, accessing/displaying sensor and location data, and mobile visual design were co-taught by both instructors. Teams comprising computer science and visual arts students design and build mobile applications for local clients in Baltimore and Washington, DC. The instructors established three companies that are willing to work with the students on their projects: Educational Development Solutions Group (EDSG), International Development Operations Systems (IDOS), and Ellipse Gallery.

To understand the shortcomings of the two courses, we studied two student projects, one from ART 434 and the other from CMSC 628. The ART 434 project was a note-taking application, whereas the CMSC 628 project was a SmartFridge application that recommends when a specific food item has to be replenished. The projects illustrated the fundamental limitation of developing smartphone applications without a combined understanding of designing aesthetically pleasing usable interfaces and efficient smartphone software. The project in ART 434 only focused on developing visually appealing interfaces. However, the user interface comprised a large number of features and requires efficient software implementation in order to be responsive and useful. For example, the handwriting recognition user interface requires an underlying machine-learning algorithm that can interpret the text. Another interesting feature of the user interface is the large number of well-placed elements. While the placement was aesthetically pleasing, rendering all the controls efficiently requires a proper understanding of the user interface hierarchy in Android. For example,

FIGURE 1. The figure illustrates the flyers for a Poster/Demonstration session for both classes in Spring 2016 and 2017. The project titles and the description is also illustrated. The figure shows the variety of projects that the students developed as part of this class.

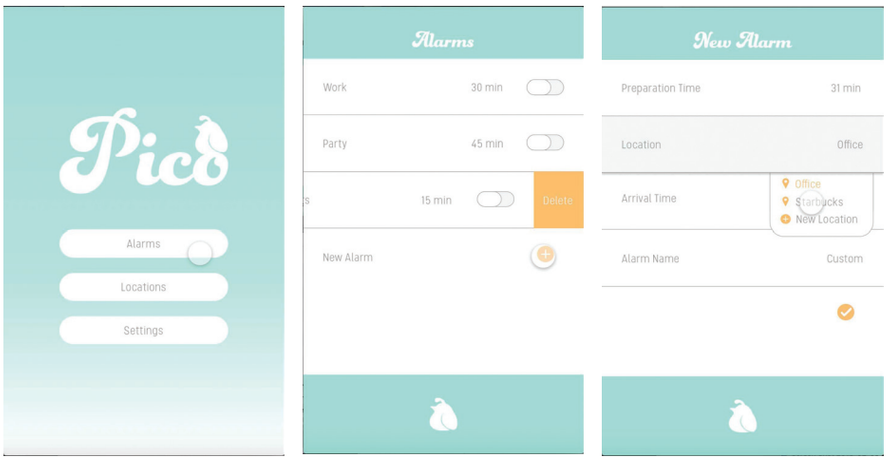

FIGURE 2. The Pico app, a simple UI designed with usability and ease of use in mind.

different views were arranged in a hierarchy called *ViewGroup*, which is rendered on a smartphone screen using a specific layout. A large number of complicated, un-optimized views, which may be required for building custom views, can slow down the app and make it unusable.

On the other hand, the SmartFridge was a fully functional application that is optimized for performance but the input interface was not very usable. For instance, the application used buttons that were closely spaced and not intuitive as an input interface for the app. Moreover, the app improperly used screen space, a constrained resource on a smartphone. It is important, therefore, to ensure that mobile applications are developed through the symbiotic mix of the artistic and computer science elements. This project aims to cross-pollinate the two aforementioned classes so that visual and computer science students can interact and eventually develop more practical and usable smartphone applications.

GOALS OF THE PROJECT

The ultimate goal is to develop a combined course that teaches the concepts of advanced user interaction design for smartphone applications and the underlying software systems implementation. In the long-term, such a course will be offered to Visual Art and CMSC/CMPE majors. Towards that aim, the project had a dual goal. First, it will be used to determine which components of the ART 434 course can be taught to CMSC/CMPE majors and which components of CMSC 628 can be taught to ART 434 students. Second, students from both classes will have real design and programming experience in their projects by working with actual clients, and thus their applications will be in use by the end of the semester. This partnership will benefit both courses because we will give lectures in both classes, allowing design students in ART 434 the opportunity to learn more about the programming background of designing interfaces for mobile apps and CMSC 628 students the opportunity to understand the design principles governing usable interfaces. Hence, this project is based on a team effort. Overall, students will gain the practical and authentic experience of designing and developing a fully operational mobile app from scratch, as well as learn about interface design and how to evaluate implementation efficiency.

In addition, we invited three speakers throughout the semester: a Microsoft developer, an interface designer from Washington, DC, and a human-centered design professional. All these guest speakers provided useful information because they are professionals in the field.

The combined course was taught in the Spring of 2016 and Spring of 2017. During both incarnations of the course, there were 40 students from the Computer Science Department taking CMSC 628 and 20 students from Visual Arts taking ART 434.

EVALUATION METRICS

Collaborative teaching experience of both classes: We collected qualitative and quantitative data from students and clients from spring 2016 to spring 2017, using questionnaires distributed during each semester, and video and photographic documentation of the collective experience [7]. This analysis helped us become more effective in future teaching collaborations. The questionnaires focus on two aspects. First, they analyze whether joint lectures and speakers help and inspire students to design and develop better apps. Second, they ascertain whether the topics covered in the lectures are useful.

Usability and management: We used the following setup. In the Spring 2016 class, we used three metrics to test users and clients: (a) Usability of the app; (b) Efficiency of the implementation of the app; and (c) Efficiency of group management. These were evaluated by the students, instructors, and the graduate research assistant. As well, these metrics were inferred from the questionnaires to the students and clients in each group, which they use to evaluate their working relationships with other members in the group and provide feedback on the contribution of each member. The data from the questionnaire remained confidential.

Analyzing client/student relationships: Students and clients were evaluated at the beginning, middle, and end of the semester. The evaluations were each geared separately and kept confidential.

We next describe the insights gained from the data collected from the students during the two incarnations of the course in Spring 2016 and Spring 2017.

INSIGHTS GAINED FROM THE COMBINED COURSE

INSIGHT 1: Combined lectures between Visual Arts and Computer Science on developing mobile apps is not useful. In the first incarnation of the combined course taught to Visual Arts and Computer Science students, we decided to offer two joint lectures. The first joint lecture was taught by Prof. Banerjee on the basic aspects of programming user interfaces in Android and the second joint lecture was taught by Prof. Cordova on the underlying

design principles for building aesthetically pleasing user interfaces for mobile applications. Specifically, the first major module of CMSC 628 is an introduction to the elements of user interface design in Android. Prof. Banerjee covers the underlying Model-View-Component implementation of user interfaces on Android, with concrete examples on how to implement custom views, custom layouts, and custom user interfaces using Java and XML. Prof. Cordova also spends the first portion of ART 434 teaching wireframing and prototyping techniques. The joint lectures are designed to answer two specific questions: (1) How can we design the best user interface possible? and (2) How can we implement the user interface while being cognizant of efficiency and performance issues on a mobile platform like Android? We also offered combined lectures for other components of the course that deal with accessing data from sensors (CMSC 628) and building user interfaces where the user can interact with sensors (ART 434), designing and building proper web services and, finally, designing efficient and usable smartphone games.

We quickly learned that such an approach is impractical. There are two reasons for this. First, the Computer Science and Visual Arts students do not have the required prerequisites to understand the material in the lectures. For instance, the Android implementation lecture was taught in Java using Android Studio, and the Visual Arts students had no prior programming experience. Even a drag-drop based methodology for code generation and development was not successful because the students lacked the understanding of using an integrated development environment like Android Studio. Similarly, the design of aesthetically pleasing user interfaces is predicated on understanding the basic art principles of developing UIs, which Computer Science students lacked. In a nutshell, developing joint class lectures for students from the two departments was unsuccessful. We believe that such joint classes can be successful only if applied to lower-level introductory classes, such as the basics of programming and basics of UI development.

FIGURE 3. The Poster/Demo session in the Spring of 2016.

INSIGHT 2: The optimal point of collaboration between Visual Arts and Computer Science is in the group projects. Another point of collaboration between the two classes were the group projects. We tried this approach in the Spring of 2016 and Spring 2017. Groups of six students were formed for the final projects, four from Computer Science and two from Visual Arts. The goal was to have two designers and four engineers developing apps. The applications developed by the students were showcased in a Poster/Demonstration session at the end of the semester. Figure 1 illustrates the flyers for both sessions and the projects that were demonstrated. The event was very popular with more than 150 attendees. The students showcased their apps in the session. Figures 2 and 4 illustrate screenshots from a small subset of the developed apps. The apps were developed with both design components and functionality in mind. We describe two apps from the perspective of software design and art next.

Pico (location-based alarms): The Pico app, illustrated in Figure 2, was developed to provide location-based alarms. The user had the flexibility to program alarms at specific times and for specific locations. The design goal for the app was a clean user interface, easy-to-use controls, and intuitive layout. The functionality included using location-based services and timers. In spite of its simplicity, the audience enjoyed using the app.

WatchA (an app for rating and maintaining a library of Netflix programs watched): Figure 4 illustrates WatchA, an app that can be used by Netflix users to rate programs they have watched, and add programs that they want to watch on their to-watch list. The UI for the app was more complicated with a lot of information on the series/movie in a small amount of space. The layout of the UI, illustrated in Figure 4, shows the care with which the layout has been designed. The functionality uses the Netflix API for download of information on the series/movies available and their ratings. The app was popular during the demonstration session and clearly illustrates the design and implementation working together to implement more effective apps.

INSIGHT 3: The key takeaway for students through this collaboration is understanding the practical limitations of design (Visual Arts) and feasibility of implementation (Computer Science).

THIS PAPER PRESENTS INSIGHTS FROM A JOINT COURSE TAUGHT BETWEEN VISUAL ARTS AND COMPUTER SCIENCE ON SMARTPHONE DEVELOPMENT AT UMBC. THROUGH QUALITATIVE DATA COLLECTED FROM STUDENTS FROM BOTH CLASSES, WE EXTRACT SEVERAL INSIGHTS FROM THE JOINT COURSE

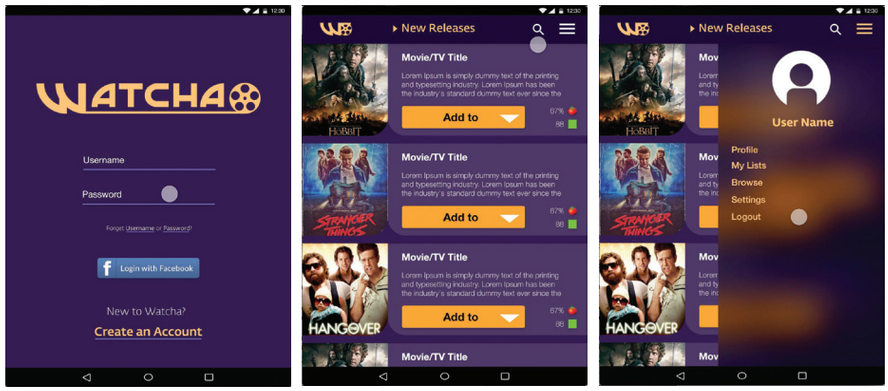

FIGURE 4. The figure illustrates the WatchA app. It shows a more complicated UI, good color and layout choice, and layout of widget.

Our conversations with the students and more formal feedback provided at the end of the year illustrates that the key takeaway from the project collaboration was finding the limitations of practical utility of designs made for smartphone apps and the limitations of present platforms to develop UIs. For instance, the Visual Arts students developed their UIs using Adobe InDesign suite. The assets developed as part of the Adobe suite were not directly importable or implementable using Android Studio. This includes challenges with placing widgets, importing custom fonts, using custom layouts, and colors. In many instances, students had to go through several iterations of designing the UI, trying to implement it, and then pivoting to design a more practically useful UI. We believe that this learning experience will translate to the real industrial world, where smartphone applications are developed using this iterative process and rapid prototyping.

INSIGHT 4: A major roadblock in combining classes in Visual Arts and Computer Science is the disconnect in tools used by both majors in application development. One of the major roadblocks we encountered was the disconnect between the tools used by the design students and the tools used by the Computer Science students. For example, the UI developed in Adobe InDesign or Experience Design, part of the Adobe Creative Cloud, cannot be directly imported into Android studio, making the process of codesigning the app a manual process. One of the major challenges, therefore, is to ensure that there are tools that can seamlessly integrate the design tools with the development tools for smartphone app development.

Other Impact on Students: This cross-disciplinary class affects students tremendously because they interacted with other departments and were exposed to a variety of different views, while learning how to deal with a real client. Most students did not have a strong philosophy, and therefore we had planned various methods to facilitate teamwork. For example, the invited speakers shared their professional experiences and emphasized the importance of teamwork in this type of project. In addition, the project promoted student maturity by helping them learn to work to a real schedule and strict deadlines. This project also helped develop a supportive student community, and after this collaboration, they were able to freelance by working with each other. This collaboration also helped students gain the confidence to reveal a finished project, since they will show their final pieces to prospective clients or employers via their design portfolios.

CONCLUSIONS AND FUTURE WORK

This paper presents insights from a joint course taught between Visual Arts and Computer Science on smartphone development at UMBC. Through qualitative data collected from students from both classes, we extract several insights from the joint course. Specifically, we find that joint lectures are not useful and the best point of collaboration between the two groups is in semester projects. Moreover, we find that the tools used by Visual Arts for designing UIs for apps cannot be directly used with development platforms, such as Android Studio. As future work, we are planning to explore if such an interdisciplinary collaboration can be applied to teaching more recent technology, such as Virtual Reality and Augmented Reality. ■

Carla Viviana Coleman Cordova is an educator, researcher, graphic designer, and artist. Her work has been featured in the books *Indie publishing* (2008) and *Graphic Design: The New Basics* (2008), both published by Princeton Architectural Press. She owns, edits, and maintains the website www.webtypography.org. Her motion design work has been aired on *Telemundo* and *BET* television stations. She has done web design for the *NIH* and her print work has been published by Princeton Architectural Press, *The Source* magazine, John Hopkins University, and others.

Nilanjan Banerjee is an associate professor in the Computer Science and Electrical Engineering department at the University of Maryland, Baltimore County. He is an expert in the area of low-power embedded sensor design and mobile and embedded systems. He is also committed to applying mobile systems concepts to improve Computer Science education.

REFERENCES

[1] Behar, J. et al., "A review of current sleep screening applications for smartphones." *Physiological Measurement*, 2013. 34(7): p. R29.
[2] Schobel, J., et al. "Using vital sensors in mobile healthcare business applications: Challenges, examples, lessons learned." Paper presented at the International Conference on Web Information Systems and Technologies, 2013.
[3] Scully, C.G., et al., "Physiological parameter monitoring from optical recordings with a mobile phone." *IEEE Transactions on Biomedical Engineering*, 2012. 59(2): pp. 303-306.
[4] Seo, H., "Bowling online: Smartphones, mobile messengers, and mobile social games for Korean teen girls." Texas ScholarWorks, University of Texas at Austin, 2014.
[5] Nickinson, P., "Android market now has more than a quarter-million applications." *Android Central*. https://www.androidcentral.com/android-market-now-has-more-quarter-million-applications
[6] "Worldwide mobile app revenues from 2011 to 2017 (in billion U.S. dollars)," last modified 2015, 2015, http://www.statista.com/statistics/269025/worldwide
[7] "Unit 4: Assessment of collaborative learning project outcomes," last modified 2010, http://net.educause.edu/ir/library/pdf/eli80084.pdf