A Faculty/Staff/Student Team for Collaboration in Developing Mobile Applications in the Software Engineering Course

Arturo I Concepcion[1], Sunny Lin[2], Tiffany Chiang[2], Patrick O'Connor[2],
Michael Korcha[1] and Darnell Shumpert[1]
[1]*School of Computer Science & Engineering, California State University San Bernardino, San Bernardino, U.S.A.*
[2]*Division of Information Technology Services, California State University San Bernardino, San Bernardino, U.S.A.*

Keywords: Collaborative Learning, Project-based Learning, Immersive Learning, Mobile Applications.

Abstract: A faculty/staff/student team was used as a collaboration model to develop, maintain, and improve mobile applications built for use by students on the campus of California State University, San Bernardino. The team was developed in the software engineering class where the projects are mobile applications requested by real clients in the campus. After the class is over, the students continue the development through independent studies and senior projects. During this time, the students work with a team of student interns and staff from the Administrative Computing Services, Information Technology Division of the campus until the mobile applications are published in both Google Play and Apple App Store. Using the CSUSB Student Opinion of Teaching Effectiveness (SOTE) surveys of the software engineering classes of 2009 – 2013, we found that students learn tremendously through hands-on experience and actual interactions with real clients, and also found that the principles and concepts of software engineering are learned better.

1 INTRODUCTION

A unique partnership was formed. The CSUSB faculty/staff/student team developed and maintained mobile app products for the campus for use by students. Currently these mobile app products are: CSUSB Mobile, Tour CSUSB, CSUSB RecSports, and CSUSB Library. CSUSB Mobile provides information and service to students for accessing class schedules, financial status, grades, payments, Blackboard, etc.; Tour CSUSB provides a virtual tour of the campus, which is used for recruiting students; CSUSB RecSports provides schedules and information regarding physical fitness and training at the fitness center; and CSUSB Library provides information and service to students using the libraries at both San Bernardino and Palm Desert campuses. Figure 1 shows the Web site https://mobileapps.csusb.edu. The mobile apps run on both Android and iOS platforms. The mobile apps were built by students for students.

The CSUSB faculty/staff/student team consists of faculty and students from the School of Computer Science & Engineering, College of Natural Sciences, and the staff and student interns at the Administrative Computing Services, Information Technology (ACS/IT) division.

There were two news releases regarding the publication and availability of "CSUSB Mobile" in our campus: "University adds more mobile apps to help make navigating CSUSB easier" which appeared on 28 Oct 2011 http://news.csusb.edu/2011/10/university-adds-more-mobile-apps-to-help-make-navigating-csusb-easier/ and "CSUSB mobile app updated with improved performance, graphics" which appeared on 31 Oct 2012 http://news.csusb.edu/?p=18592. The first article is announcing the availability of the mobile app services on campus while the second article is about re-engineering the mobile app to improve graphics, significantly improve performance, and minimize bandwidth usage. The second version uses a hybrid approach, where approximately 90 percent of the content is installed on the device and the rest resides on a secured server.

The goals of the faculty/staff/student team at CSUSB are two-fold:

-- To create a model in the software engineering course so that students would learn the principles and practices of software development through the mobile app projects in class.

-- To establish in the ACS/IT division the capability to support and manage the production of mobile app

I Concepcion A., Lin S., Chiang T., O'Connor P., Korcha M. and Shumpert D..
A Faculty/Staff/Student Team for Collaboration in Developing Mobile Applications in the Software Engineering Course.
DOI: 10.5220/0004839102600267
In *Proceedings of the 6th International Conference on Computer Supported Education* (CSEDU-2014), pages 260-267
ISBN: 978-989-758-020-8

projects from either clients on campus or external to the campus.

Figure1: CSUSB Mobile Applications.

To achieve the goals above, the following are the objectives:

-- The students should know how to elicit user requirements.

-- The students should know how to write the software requirements specification (SRS) document.

-- The students should know how to write the software project management plan (SPMP) and the corresponding software quality assurance plan (SQAP).

-- The students should be able to develop and manage repositories and development server environments in support of software development.

-- The students should know the different mobile technologies and programming languages used in the development of mobile applications.

-- ACS/IT staff personnel provide support to a team of student interns who are working on existing and currently developing mobile app projects.

-- ACS/IT provide financial support to student interns who are working on existing and currently developing mobile application projects.

-- ACS/IT upper management provide management and support to the faculty/staff/student team for mobile application projects.

The faculty/staff/student team supports project-based learning (PBL), which is described by (Markham, 2011) as integrating learning and knowing. Because of the ubiquity of mobile devices and the seemingly endless new applications that can be developed, students become motivated and eager to complete the project.

2 SOFTWARE ENGINEERING COURSE

The software engineering course is an upper-division requirement for majors in the School of Computer Science & Engineering, CSUSB. The course is run as a mock software engineering (Concepcion, 2005) where the class is organized into management teams and programming teams and each student plays a role in the software process as: project managers, team leads, software engineers, quality assurance, system administrators and technical writers. One of the major software project used in the class is AlgorithmA (Algorithm Animation) project (Concepcion, 1998, Concepcion, 1999, Concepcion, 2000, Concepcion, 2005, James, 2008). This software project has been maintained for over 20 years!

In Winter 2011, some of the staff at the Administrative Computing Services requested to include in the software projects, a mobile application project, in the software engineering class. which was deemed important because ACS was thinking of developing mobile apps for use by students in campus. So one of the software projects to be done in that quarter term was a mobile app project. The software project will be used as a prototype from which we developed the CSUSB Mobile app. From this prototype, the CSUSB Mobile was created. The following year, Winter 2012, all software projects were mobile app products. Three of them were published: CSUSB Library, CSUSB RecSports, and Tour CSUSB. This winter 2013, we prototyped five more mobile app products: Sodexo CSUSB Dining, Coyote Radio, RAFFMA Museum, Student Advising, and Slidewinder, our first iOS mobile game.

2.1 Software Requirements Specification

The software life-cycle starts with a document called Software Requirements Specification (SRS), which contain the software requirements that the client needs. The elicitation process is obtained through interview, meetings, and e-mail communications between the client and the development team. After about 2 weeks of the elicitation process, the team writes the SRS following the IEEE Std. 830-1998 format for SRS. The SRS is completed when the client approves the document. All students are required to write the SRS and then the best written SRS is selected to be the SRS for the specific mobile app product.

2.2 Software Project Management Plan

While the SRS is being finalized and completed, the management team writes the Software Project Management Plan (SPMP). This document contains the organization of the development team, the resources, the milestones schedule, management process, software process, and the risk management. Following the IEEE Std 1058-1998 format for SPMP, the team submits this document to the instructor for approval. The SPMP becomes the definition of the software process that the development team will follow to implement the mobile app product. Figure 2 shows the project organization of the software engineering class. And Figure 3 shows the list of clients for the project organization shown in Figure 2.

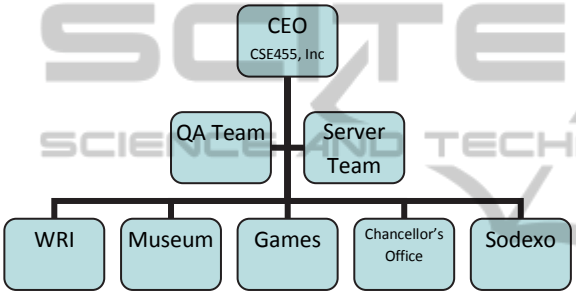

Figure 2: Project Organization for software engineering.

Besides the management team and the development team, there are two support teams: the server team and the quality assurance team. The server team configures the development and production servers. They also set-up the repositories where all the source code and documentation will be stored. The QA team is in-charge of software reviews and testing of source code.

Following the Agile Software Methodology, we had two iterations of the mobile app product in the quarter term. The first iteration takes about 4 weeks and the second iteration takes the last three weeks of the term.

The document also includes risk management. This is the identification of the software product risks and the software product risks. The management team must identify these risks and when it occurs they should have a plan on how to handle the risk.

Table 1: List of clients for project organization in Figure 2.

Client (CSE 455 Mon / Wed)
Water Resource Institute – Tree/plant identification for 2nd grade students (Boykin Witherspoon)
Museum – Virtual tour of Egyptian collection (Paige Taylor).
Games – SlideWinder (Jack Price).
Chancellor's Office – How to get to college gamification (Robin Wade).
Sodexo – Campus dining/Cafeteria, catering (Emily Orquiza).

To promote motivation among the student, Outstanding Software Engineer awards are given to each development team including the support teams. Each team selects their own outstanding software engineer using the criteria: significance of the contribution the product and involvement in improving the programming skills of the individual members of the team.

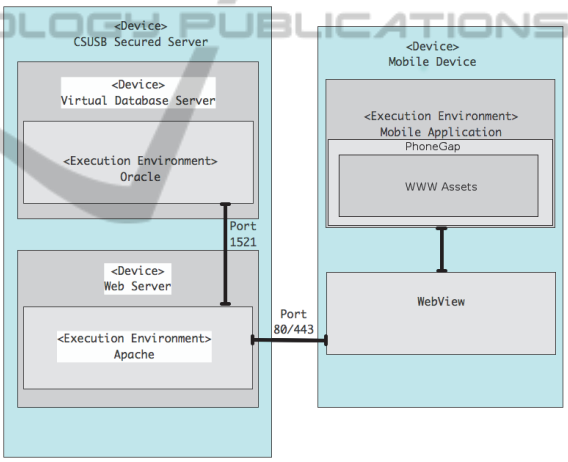

Figure 3: Mobile app development diagram.

2.3 Software Design

Several team members undertake this phase of the software life-cycle, the software design. This phase consists of two parts: the architecture and the detailed design. The Unified Modeling Language (UML) is used as the design notation for this phase. The architecture begins from the deployment diagram found in the SRS. This diagram shows the major hardware and software components of the system that will be needed to run the mobile application. The hardware components usually consists of the secured server and mobile devices while the software components usually consists of the PhoneGap application, user-interface, and the

database system. The detailed design consists of the detailed class diagrams or component diagrams of the software components of the architecture. The detailed design could also consist of important algorithms that are used in solving problems in the implementation. Shown in Figure 4 is the UML deployment diagram of a typical mobile app product.

2.4 Implementation and Coding

The most common implementation and coding used is Extreme Programming (XP). This method allows the production of code through two steps: coding and testing. The team of two students, given the requirements of what this software component will do, will code and test their increments of code until the requirements are fulfilled. Once the different components are completed and tested, they are integrated into larger and larger components. Integration testing are done to ensure they work and free of faults/bugs. Finally, system testing is done to ensure that the mobile app works with other systems, such as database systems and Web sites.

Some students used the SCRUM method, where sprints of important features are implemented in a period of one week. The features are chosen from a priority list which was specified by the client. Other students used the conventional Waterfall model and had mini-Waterfall model for each iteration of the product. All source code, design and documents are stored in the repository set up by the server team.

2.5 Software Reviews and Testing

The class follows the non-executable testing and executable testing methods. In the non-executable testing, the class reviews the SRS through a series of use case diagrams and sequence diagrams where a scenario of how the software will be used is applied to determine any inconsistent or ambiguous requirements are present. The student then use the same techniques above for SRS in reviewing the software design. All of these testing methods and techniques are found in the document Software Quality Assurance Plan (SQAP). It is written using the IEEE Std. 730-1998 format.

The quality assurance team performs the alpha testing of the mobile app product. Here the testing is done if all the functions listed in the SRS are working properly. If a fault/bug is found, the mobile app is given back to the development team to fix the fault/bug. This process is repeated until no more fault/bug is found. The development team is not credited the completion of this phase of the software-cycle, until the development team pass the alpha testing set by the QA team. The mobile app product is then given to the client for the final testing, the beta testing. The client and all the users in their office are given a copy of the mobile app product and tested for actual use in their environments. Any fault/bug are reported back to the development team and fixed. This process is repeated until no more faults/bugs are reported.

2.6 Demo and Presentation

The demo and presentation were done on the finals day of the quarter term. This is done at the end of the second iteration of the mobile application following the iterative software process as defined in the SPMP. All the clients of the mobile apps, invited representatives of the local software companies, and guests of the students are invited to the demo and presentation.

The demo and presentation is in a form of an exhibition/booth style where each team sets a location in the labs and there they present and demo their mobile app product to any visitor of their "booth." They either have a mobile device or an emulator running to show the product. They may also have powerpoint presentation in their "booth." The students remove their "software engineering hat" and put on a "sales person hat" selling their products to the visitors.

Local software companies that have visited us in the past are: Esri, Optivus Proton Therapy, iMedRIS, Epic Management, Kelly Space, and Surado Solutions. We also have the announcement of the Most Outstanding Management Team awarded on the day of the demo and presentation.

3 EVALUATION RESULTS

PBL shifts the emphasis from the core curriculum of software engineering to the application of the principles and concepts to actual projects. (Markham 2011) also said that the student's drive, passion, creativity, empathy, and resilience cannot be taught but are exhibited by the student when doing the project. To measure the effectiveness of the PBL techniques through the faculty/staff/student team, we used the CSUSB faculty teaching evaluation survey, Student Opinion of Teaching Effectiveness (SOTE).

3.1 Student Opinion of Teaching Effectiveness

SOTE is used to evaluate the teaching effectiveness of faculty at CSUSB and is used to determine the effectiveness of PBL in the software engineering course. It has 5 questions:

Q1: Rate your interest in the subject matter of this course before you took the class.

Q2: How many class sessions did you attend?

Q3: Why did you take this course?

Q4: How would you rate the overall quality of instruction in the course?

Q5: How would you rate your professor's specific contributions to your learning in this course?

The last two questions (Q4 and Q5) were used to rate the student's evaluation of the course and the instructor.

3.2 Results

The last two questions are evaluated as follows: lowest score of 1 for unsatisfactory, 2 for poor, 3 for fair, 4 for good, 5 for very good, and a highest score of 6 for excellent. For each question, the total number of students, the average score, and the median score are printed. The mobile apps project started in 2011 and has been used as the software project in software engineering since then. See Table 2 for the comparison of results of the SOTE from 2009 – 2013.

Table 2: Results of SOTE

Year	No.	Av4	Md4	Av5	Md5
2008	34	3.6	3.5	3.6	4
2009	29	3.6	3	3.5	3
2010	33	4.7	5	4.5	5
2011	47	4.8	5	4.8	5
2012	62	3.8	4	3.7	4
2013_1	36	4.5	5	4.4	4.5
2013_2	15	5	5	5.1	5

where Av4 and Md4 are the average and median scores for Q4, resp., and Av5 and Md5 are the average and median scores for Q5, resp.

As can be seen, the number of students taking the course is increasing. Currently the enrolment in 2014 has again increased to about 80 students. Due to increasing enrolment, the class was split into two sections starting in 2013. This is the reason why there was a decline in the results in 2012. When the software engineering class was split into two sections, the results came back up. In summary, the results from 2011 – 2013 improved as compared to 2008 – 2010.

What is not shown in the results is the investigative framework that Blumenthal mentioned when doing PBL: How the students determined the requirements from the client by writing the SRS; how the students made plans on completing the project on time by writing the SPMP; how the students made the architecture design of the project using UML diagrams; how the students acquired the necessary skills in learning mobile technologies and languages through the help and guidance of the faculty/staff/student team; how they worked in teams; how they make presentations to the clients for demonstrations and getting feedback from them to build the next prototype. These were all performed and done by the students in the course of software engineering, which contribute to their work readiness and future careers in management when they graduate (Jollands 2012, Tynjala 2009).

4 UPPER MANAGEMENT AND ADMINISTRATION SUPPORT

The faculty/staff/student team could not have been successful without the support of the CSUSB upper management and administration. Support also came from the College of Natural Sciences and the School of Computer Science & Engineering.

4.1 Information Technology Division

The IT Division has been in existence since 1992. Its mission is to foster the evolution and development of information technology resource management and to encourage the integration and utilization of new and existing campus computing, communications, and media tools and applications. In addition IT also supports the teaching/learning process, research, scholarship and creative activities, academic/administrative services, and local regional public outreach. IT takes pride in that it is guided by the following principles:

-- Be responsive to the changing information

resource technology needs of a highly diverse student, faculty, and staff community.
-- Offer support and leadership through collaborative efforts with faculty, student, and staff.
-- In a participative manner, perpetuate information resource technology integration as a part of the academic and administrative fabric covering all programs.
-- Aggressively respond to the tactical objectives set forth by the campus strategic plan.
-- Advance CSUSB as one of the foremost teaching/learning environments in higher education by applying, as appropriate, technology solutions.

The Administrative Computing Service office of IT is the entity directly involved in the faculty/staff/student team. This is the office that provided the staff and supports the student interns for mobile app products. One of the missions of IT is to support teaching/learning process for students and so the faculty/staff/student team is directly relevant to this mission. The mobile app products that were created are relevant to being responsive to the changing information resource technology needs of a highly diverse student, faculty, and staff community.

Currently the campus is focused on delivering the core administrative system (PeopleSoft) and LMS to desktops via both wired and wireless connections with good reliability and maximum speed. In the past two years, the campus saw the increasing number of mobile devices connecting to campus networks and retrieving information in the same volume as desktops and laptops. ACS/IT recognized the opportunity to use mobile devices and partnered with the School of Computer Science & Engineering to form the faculty/staff/student team.

During the development of mobile app products with the faculty/staff/student team, there were sensitive information (such as enrolled classes, grades, student account information, and making payments) that must be kept secured while students and student interns are working on the mobile app products. So for security reasons, staff in the ACS developed service call libraries for student developers, which encapsulated the database connection configurations, SQL queries, and query result sets. Developers just need to reference the service calls and will receive the proper result for the mobile app needs without direct interaction with the student information systems, HR systems, or finance systems. In doing so, ACS is able to protect private and confidential information from the student developers but at the same time make the information available at the developer's finger tips.

Two presentation/demonstration were made at the Administrative Council meetings: 01 August 2011 and 08 October 2012. The former president, Dr. Al Karnig, and current president Dr. Tomas Morales and the members of the Administrative Council were all impressed by the students' presentation and all supported the faculty/staff/student team and the mobile app products that were presented and demonstrated. Another presentation/demonstration was made to the Philanthropic Foundation Board of Directors on 08 December 2011 and again all were impressed by the work and the professional way the students presented the mobile app product. President Karnig and other upper administration officers were present at that meeting.

4.2 Evolution of Faculty/Staff/Student Team

The mobile app products started in the undergraduate software engineering class (CSE 455) taught by Dr. Concepcion where prototypes were developed first and then continued development after the class is over. Then through students' independent studies and senior projects and student interns working at the ACS/IT, the development is continued. The software engineering class is offered only in winter term, and in the following spring and summer, the faculty/staff/student team continued to work on the mobile app products until it is published usually in the fall term.

We started this cycle of development in winter 2011 when Sunny Lin and Tiffany Chiang from ACS/IT suggested to Dr. Concepcion to undertake a mobile app project in my software engineering class because ACS/IT had been wanting to start a capability in the campus to develop their own mobile app products. The winter 2011 software engineering class created the first prototype of CSUSB Mobile and after several former students of CSE 455 using independent studies continued working on the product in spring and summer 2011, CSUSB Mobile v. 1 was published in Fall 2011. The ACS/IT in summer 2011, created two student intern positions to maintain and continue the development of CSUSB Mobile.

The next winter 2012 software engineering class, the students created again prototypes of three new mobile app projects: CSUSB Library, CSUSB RecSports, and Tour CSUSB. Again several students continued the development via independent studies

and senior projects, and in Winter 2012, these three new mobile apps were published in both the Apple App Store and the Google Play market. At this time, ACS/IT obtained a Student Vital Technology Initiative grant to support two additional student intern positions, bringing the total number of student interns at four. ACS/IT staff provided the support for student's use of the university servers and databases.

A supporting class is CSE 322 (Web Programming) which is taught by Dr. David A. Turner. In this class, the students learn PHP, JavaScript, CSS, and PhoneGap. These are programming languages and mobile technologies that are essential to mobile app projects. Although CSE 322 is not a pre-requisite to CSE 455, students who have taken this course before taking CSE 455 form part of a core of students who will be the major developers and programmers for the mobile app projects in CSE 455.

In winter 2013, Dr. Concepcion taught again the software engineering class and this time we had six new clients: Water Resource Institute, Student Advising, Coyote Radio, Museum, Sodexo, and Chancellor's Office "How to get to college." The whole process is repeated. The faculty/staff/student team is an appropriate model to use in sustaining the mobile app projects for a very long time.

5 QUALITY, PERFORMANCE, AND PRODUCTIVITY MEASUREMENTS

Tools such as the browser add-ons Firebug and YSlow! for Mozilla Firefox and the built-in Chrome Development tools were used to test load times, quality of code, number of external HTTP calls and DNS look-ups, and verify caching is in place.

Improvement was based on comparisons between major and minor versions. Page load speed, page size, reduction in HTTP calls, and quality of code were major factors in determining improvements throughout the application updates. Weekly meetings were held to discuss progress, improvements, and new features to implement.

The following are specifically measured: number of HTTP requests, gzip compressions, CSS and java script minification, number of URL redirects, avoiding invalid links, page load speed, and page size.

The measurement is determined by demand of Web traffic. Especially for high visited pages to understand how efficient we can deliver contents to end users. The categories of information we are collecting includes Web object name and path, method status, type size, latency, and over all timeline for delivery. The Firefox add-on YSlow! gives grade levels on these measurements and these grades are used as a basis for quality and performance.

6 CONCLUSIONS

In conclusion, the faculty/staff/student team is a very appropriate model of collaboration between the ACS/IT Division and the School of Computer Science & Engineering/College of Natural Sciences in creating the capability of the campus to produce mobile app products not only for the students in campus but also external entities that would require such services. The ultimate result is the greatest learning experience by the students. We have also shown that we can produce good quality mobile app products following the principles of software engineering.

6.1 Student Learning and Education

(Blumenfeld, 1991) stated that investigation by students is responsible for sustaining the doing and supporting the learning. Students in software engineering are encouraged to investigate and find answers that are not available in textbooks or in the classroom lectures, such as finding what the clients really want on the mobile app, how to design and plan the entire project so it delivered by the finals day of the term, studying new technologies and languages, searching answers on the different chat and Web sites, and other activities that are part of the development process but not taught in class. The SOTE scores reflect their perception on whether they are learning and how the instructor is teaching the course materials but they have actually learned when they deliver the completed mobile app by the finals day to the client.

The faculty/staff/student provided the sustaining and supporting the learning for PBL. This team provided the expertise and consultancy for students in software engineering for guidance and directions in the development of the mobile app products since about 90% of the students in class have never programmed in JavaScript, HTML5, CSS, Objective-C, XCode, and PhoneGap, and they needed the initial help and training in the first few weeks of the class to learn how to use these mobile technologies and languages.

Then after the software engineering class is over, the faculty/staff/student team continue the development of the prototype mobile app products until they are published in both Google Play and App Store later on. The team is assisted by several former software engineering students enrolled in independent studies or senior projects.

Although these were good benefits to the students' learning software engineering, there is also a weakness, which was described by (Lee 2012). This is called social loafing where some students may not perform as expected and thus create dissatisfaction within the team. Since the grade is given as a team grade, these students who did not perform also get the same grade. Last 2013, we have experimented on having the student project manager or the team lead, with the assistance and approval of the instructor, to "fire" the student who is not performing his/her task. The student who is fired is given another task by the instructor, which may or may not be related to the project being developed by the team.

6.2 Customer Satisfaction and Future Projects

As of this writing, CSUSB Mobile averages a solid 1.5K visitors per day, peaking at 2.5K unique visitors recently. Both CSUSB Library and CSUSB RecSports were announced as available for students use. The Tour CSUSB is going to be used as a recruiting tool both here in California and abroad. A second version is underway to be produced in four other languages: Korean, Japanese, Chinese, and Spanish. Our clients are very satisfied and are looking to upgrading their current versions. Five new mobile app products will be added this fall 2013: Sodexo Dining, RAFFMA Museum, Coyote Radio, Student Advising, and Slidewinder (an iOS mobile game).

Norco College, in partnership with CSUSB, obtained a Title V (Habilidades Unidos) grant for 5 years which began in Fall 2011. The goal of the grant is to establish a 2+2 pipeline bachelor's program from Norco to CSUSB in commercial music, graphics art, and mobile and game development. The first program is with the Department of Music, the second is with the Department of Art, and the third is with the School of Computer Science & Engineering. Both Dr. Concepcion and Dr. Turner are the faculty responsible for this part of the grant. The grant supports the faculty/staff/student team with equipment and computers needed for the mobile app products.

CSUSB benefits from this by having mobile app products available to campus students and we are sharing the development framework and software developed to all other CSU campuses. We are currently having communications with the Chancellor's Office, on the Distressed Students project. The Arrowhead Credit Union, a local company, has contacted us to build a charitable foundation mobile app for them. Sodexo, a world-wide company on food services, has agreed to develop a pioneer mobile app for their food service at our campus. Another local company, Innovative Economy Crowd, is consulting with us on an mobile engineering applications.

REFERENCES

Blumenfeld, P. C., 1991. Motivating project-based learning: sustaining the doing, supporting the learning, In *Educational Psychologist, 26.*

Concepcion, A., Cummins, L., Moran, E., Do, M., 1999. Algorithma 98: an algorithm animation project, In *ACM SIGCSE 1999, 30th ACM Special Interest Group in Computer Science Education.*

Concepcion, A., Leach, N., Knight, A., 2000. Algorithma 99: an experiment in reusability and component-based software engineering, In *ACM SIGCSE 2000, 31th ACM Special Interest Group in Computer Science Education.*

Concepcion, A., Bernstein, M., Fitzgerald, K., Macdonell, J., 2005. A ten-week mock software company, In *ACM SIGCSE 2005, 36th ACM Special Interest Group in Computer Science Education.*

Jollands, M., Jolly, L., Molyneaux, T., 2012. Project-based learning as a contributing factor to graduates' work readiness, In *European Journal of Engineering Education, 37(2).*

James, W., Lucas, P., O'Connor, J., Concepcion, 2008. A., Re-engineering the AlgorithmA project for long-term maintenance, In *OOPSLA 2008, 23rd ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications.*

Lee, H-J., Lim, C., 2012. Peer evaluation in blended team project-based learning: what do students find important?, In *Journal Of Educational Technology & Society, 15(4).*

Markham, T., 2011. Project-based learning, In *Teacher Librarian, 39(2).*

Tynjala, P., Pirhonen, M., Vartiainen, T., Helle, L., 2009. Educating IT project managers through project-based learning: a working life perspective, In *Communications of the Association for Information Systems, 24(16).*