120th ASEE Annual Conference & Exposition
FRANKLY, WE DO GIVE A D*MN
June 23-26, 2013

Paper ID #7559

Designed-for-Motivation based Learning for Large Multidisciplinary Team One Semester Hands-on Network based Course Case Study

Dr. Emil H Salib, James Madison University

Professor in the Integrated Science & Technology Department at James Madison University. Current Teaching - Wire-line & Wireless Networking & Security and Cross Platform Mobile Application Development. Current Research - Mobile IPv6 and Design for Motivation Curriculum.

Joshua Alfred Erney, James Madison University

Joshua Erney is a Senior Integrated Science and Technology major with a concentration in Telecommunications and a minor in Computer Science at James Madison University.

Mr. Matthew Edwin Schumaker

Designed-for-Motivation based Learning for *Large Multidisciplinary* Team *One Semester* Hands-on Network based Course Case Study

Dr. Emil H Salib, Joshu A. Erney, Matthew E. Schumaker
Integrated Science & Technology Department, James Madison University, Harrisonburg, VA
salibeh@jmu.edu, erneyja@dukes.jmu.edu,schumame@dukes.jmu.edu

1. Introduction

A key goal of the Integrated Science and Technology (ISAT) program at James Madison University [1] is to prepare students to be professionally well equipped when entering the workplace or enroll in graduate programs. This is accomplished by developing students' ability to become problem solvers who are able to investigate local, national, and global issues not only from a science perspective but also from technology, engineering and social context perspectives. During their Junior and Senior years, the program provides students with a unique hands-on research, design and prototyping experiences in the form of Senior Capstone Projects. The intent of a capstone is for students to utilize competencies developed in the first three years of the curriculum in the solution of a problem of their choice.

The literature on capstone project experiences is fairly robust particularly in terms of the diversity of the approaches explored. A team at Ohio Northern University (ONU) asserted in [2] that students who get involved in extracurricular design activities instead of the mandatory senior Capstone Projects tend to be highly motivated, gain the Engineering and Technology experience they need and have better chances in finding jobs upon graduation. There is also a growing trend to encourage students to take a more active role in their own education where the instructor is a facilitator of learning. In this model [3], the emphasis is more on learning and less on teaching, and it requires instructors to incorporate more active and student-centered learning methods into their courses. These methods include collaborative, cooperative, problem-based, and project-based learning.

However, most if not all of these Project-based cases tend to be limited in their *scope* and also in terms of the *team size* (typically 3 to 4 students per project). There is no doubt that these projects provide the students involved with the opportunity to hone their learned skills. However, it does not engage the students in what is considered these days the norm in the real world, that is, being a member of a large diverse and multidisciplinary team, particularly in the domain of network-based application development. Typically, the large team size translates into a large end-to-end solution to a complex problem defined by or presented to the students. Projects of large scope and large team size present a unique set of challenges that are typically considered the domain of real-world enterprises and rarely are addressed in an academic classroom environment. For examples, the probability that students would fail to achieve their goal is typically high due to (1) the breadth of technologies that must be used to achieve the solution, (2) the lack of knowledge and experience of the students in using many of the technologies, (3) the lack of experience on the part of students at managing a complex project, and (4) the inexperience of all parties involved at running/taking a course in this fashion.

In this paper, a number of students along with the course advisor present a unique experience and approach where 14 students, with diverse backgrounds and interests in science, technology and engineering at the College of Integrated Science and Engineering (CISE) were challenged to define, design, implement, deliver and assess an end-to-end solution, in a semester hands-on project-based course. This course is the main focus of the case study presented in this paper. The course advisor embraced the "Expectancy-Value" model of motivation described in [4, 5]. As described later, the class students were motivated to the extent they *valued* the outcome of the project and *expected* that their own action can help bring about that outcome.

This paper is organized as follows. In the first section, we provide a definition of the case study undertaken by the 14 students and their course advisor with specific emphasis on the case study's assumptions, success criteria and requirements. In the next section, we provide in some details the approach designed and applied to meet the requirements of the case study. The results and analyses of the case study from both the students' technical and learning achievements perspectives are discussed and detailed in the next two sections. The final section covers conclusions, key learning and recommendations.

2. Assumptions, Success Criteria and Requirements of the Case Study

The basic premise of this one-semester course case study was to foster highly motivational learning environment. This case study was performed under the following set of assumptions:

- ***Course***: required ISAT/Telecom-Networking & Security Sector course;
- ***Size of the class***: 14 students;
- ***Students diversity***: (2 females and 12 males); 8 juniors & 6 seniors with major and/or minor in ISAT/Telecom, ISAT/IKM (Information Knowledge Management), Computer Science (CS), Computer and Information Systems (CIS);
- ***Duration of the course***: one semester (15 working weeks);
- ***Frequency and duration of class meeting***: once a week for 100 minutes;
- ***Out of class time***: 3 to 4 hours a week;
- ***The role of the course professor (to be referred to from this point on as the course advisor)***: A coach and customer advocate, an assessment and feedback provider, advisor (more like a group manager who does not have complete in depth of how-to technical knowledge of all the components of the project);
- ***Facilities***: 24x7 accessible Instrumentation and Measurement Lab, Networking and Security Lab;
- ***Resources***: official advisor (1 faculty), 2 TAs (Consultants), unofficial advisor (1 faculty), Lab support (1 electronics technician);
- ***Budget***: $500 dedicated for hardware components (such as sensors, Arduino microcontroller, smart mobile devices, etc.). Rely heavily on the use of open source software, such as, Linux, Python, Java, Android, PhoneGap (Cordova), Web Technologies (HTML, JavaScript, etc.), MySQL, Apache Web Server;
- ***Expected deliverables***: Working prototype of the end-to-end solution, project report, and presentation.

In designing this project-based case study where a whole class of 14 students is engaged in a large one integrated project, it was a must to identify a set of success criteria. These were identified following the Expectancy-Value model [4, 5] as:

- the successful identification of a topic, an idea, an opportunity or a problem the entire class would consider of high *value*, and
- the successful design of an approach that consistently enables enforces and supports their *expectations* that the expected deliverables can be achieved.

In addition, one more success criterion was identified that is directly relevant to this course, that is,

- the successful delivery of a working network based application prototype of the end-to-end solution to be proposed, designed, implemented and delivered by the students.

To achieve the first criterion the entire class was trusted to conduct a number of brainstorming sessions outside and during class time. This took place on the sixth week of the semester following the completion of an introductory presentation and four weeks of technology and tools demos and exercises to be described in some details in the next section. Initially, the students made recommendations of a number of projects they thought were practical and realistic. Eventually, the class chose a problem that was very personal, that the students were dealing with themselves, and in a way that has made the whole class get behind each other. The entire class identified with this opportunity and rallied behind the high value of developing an end-to-end solution to such daily and personal problem. The opportunity as stated by the students in [6],

> "*We've experienced problems with finding open rooms to study in and with rooms being overcrowded to the point of every computer being taken. Sometimes the classrooms are occupied by large project groups, classes, or lots of individuals making it hard to get work done. It would be valuable to have access to a service that allows you access to the number of people in a given classroom or computer lab without physically visiting the classroom or the lab.*"

To realize the second and third success criteria, the course was designed by the course advisor to meet the following requirements:

- to introduce the students to basic networking and computing technologies and tools;
- to have an organizational structure that would support modular and integrated solution design;
- to form high performance and diverse team(s) that are effective and efficient;
- to foster project-based environment where the students are responsible for their own learning;
- to ensure adequate accessibility and availability of resources for the students (human resources, facilities, hardware, software, and budget); and
- to adopt an assessment process that is effective in providing timely feedback to the students and enabling them to apply adjustments as necessary.

3. Course Design to Meet the Case Study Requirements

This class was designed to meet the above listed requirements and to provide the students with a real-world experience of working on a large team in a setting designed for motivation. In the first week of the semester, the students were introduced to the concept of end-to-end integrated solution in the context of network-based applications and services. The course advisor, for simplicity's sake, explained that an end-to-end solution could be implemented by dividing the large team into sub-teams along the lines of the most basic and generic network-based application architecture model, that is, back-end physical, central network and front-end user experience computing components as shown in Figure 1. From this point on, these will be referred to as back-end, central and front-end components.

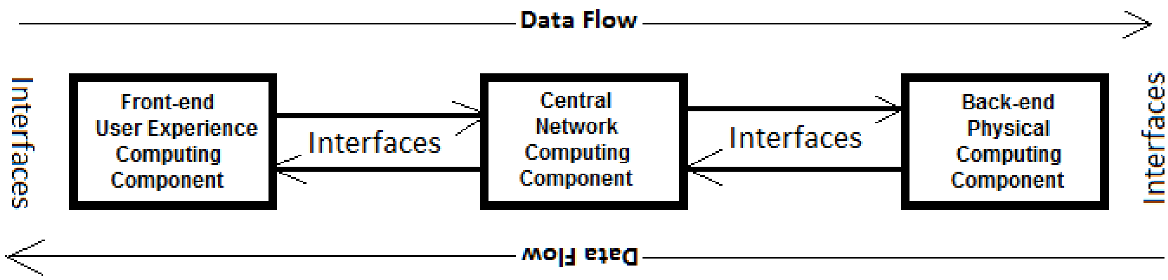

Figure 1. Simple Network-based Application Architecture Model

The back-end component of the solution represents the data collection, monitoring and/or control functions and/or elements. While the front-end component represents user experience and associated user interfaces, such as, graphical user interface (GUI) or command line interface via which a user would enter, submit or retrieve data. The central component, in the most generic terms and open ended fashion, could represent a typical network cloud or even just a single physical connection that enables the front-end and back-end components to communicate with each other. Since the course is part of a networking and security curriculum, the course advisor introduced to the students, a number of possible examples of the central component such as data storage that could be accessed by both the back-end and front-end components for the purpose of submitting, storing and retrieving data over a variety of different physical and logical networking channels and protocols.

In addition, the course advisor also talked the students into the needs for yet another sub-team that would be responsible for managing the integration of the end-to-end solution. That sub-team would make sure that whatever functionality developed by each of the three sub-teams for their respective component would be seamlessly integrated to support one or more of the end-to-end solution functions. It was most difficult for the students to appreciate the need for an integration sub-team as their concept of integration is synonymous with the idea and technologies of plug-and-play. Although, some of them were introduced in prior classes to the concept of Application Programming Interfaces (APIs), they had very limited experience in how it may be applied in the context of an end-to-end solution where they as a *large* team is responsible for all its components. It is worthwhile highlighting that the majority of the efforts spent by the sub-teams, and that most of the value gained by the students were through the interaction with each other towards a common goal, that is, *an end-to-end integrated solution*.

The class was introduced to two senior Computer Science majors who would serve as technical consultants. That is, they were available to assist the students in troubleshooting issues but not to implement or develop codes or solutions. If a sub-team needs a consultant's help in the development of code or circuitry, then the students are required to negotiate such a request with the course advisor.

To improve the chance of creating high performance sub-teams, the students were asked, during the second week, to fill out a skills inventory form as shown in Table 1.

Sub-team	Role	You know	You would like to learn
Back-end	Physical Computing		
Front-end	User Experience Computing		
Central	Network Computing		
Integration	Project Management		

Table 1. Skills Inventory Form

To ensure that the students have a reasonable and common exposure to hands-on examples of the three major computing components, they were introduced to the following four basic hands-on demos and exercises over the next four weeks:

- ***Physical Computing:*** Arduino an Open-source Microcontroller Development Platform;
- ***User Experience Computing I:*** Python for Android (Mobile Application Development);
- ***User Experience Computing II:*** Django based Web Server/Application and
- ***Network Computing:*** Storage, Network Infrastructure, and Protocols.

These demos and exercises provided the students with basic hands-on knowledge of few current technologies and tools available for implementing hands-on network-based applications. The consultants were very instrumental in the design and development of these demos and exercises. Unfortunately, the course advisor did not offer a corresponding exercise or example in the area of system integration and/or management.

After the four weeks, the students shared with the course advisor and consultants their proposed project (see section 2). In addition, the consultants reviewed the filled in Skills Inventory forms with the students and helped the students update their input. Based on the updated forms and the input of the consultants, the students were systematically placed into one of four sub-teams (the number in parenthesis represents the number of students): back-end (4), central (4), front-end (4), and integration (2). See Figure 2 for the initial organizational chart.

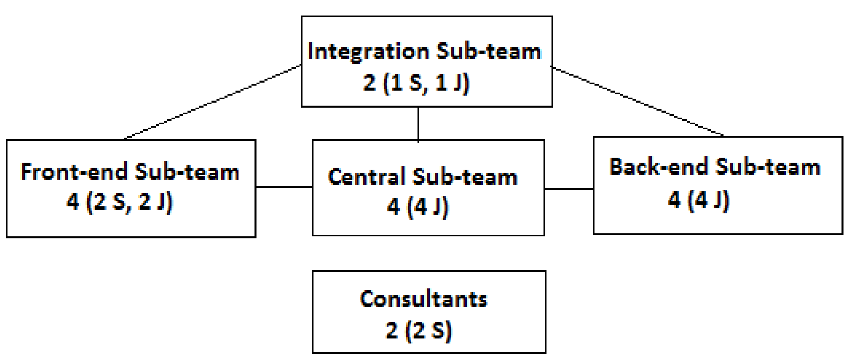

Figure 2. Class Organizational Chart (S: Senior, J: Junior)

Project status meetings were held weekly. These status meetings were supposed to be managed entirely by the integration sub-team. Each of the other three sub-teams was to submit a status report to the integration sub-team prior to the weekly status meeting. Included in the status report was a running action items register. The integration sub-team started each meeting with an overall assessment of the project highlighting progress made and issues encountered by each of the sub-teams. In addition, the integration sub-team presented the latest view of the integrated solution architecture and data flow along with the latest set of end-to-end solution features and functions. Following the *brief* overall assessment of the project, each of the three computing components sub-teams presented their status in details and performed demos of functionality as per the commitment and ownership tracked in their action items register. Towards the end of the semester, the sub-teams focused on integrated demos among the three computing components. During the status reporting, the class students provided their feedback and insight to the presenters. At the end of each status meeting, the course advisor requested that each sub-team summarizes the new, updated and closed action items. Following the meeting, and within 24 hours, each sub-team submitted an updated copy of their weekly status report including updates and additions discussed and agreed on and committed to in the form of updating the action items register.

The breakdown of the class grade was 20% on the weekly status, 10% on the project dry run and 70% on the final project demo, presentation, and report. After our weekly status meeting, each student was provided with a brief written feedback and grade taking into account the student's individual contribution and the overall accomplishment of his/her sub-team. The weekly status grade was used to ensure that those who are typically accustomed to hiding behind the group contributions are identified and well informed of where they stand. The dry run demo and presentation was used to help the students hone their presentations skills and the logistics of how best to deliver an effective demo of their prototype. The grade for the dry run was based on the contributions and performance of the individual sub-teams; each student in a sub-team received the same feedback and grade. The grade for final presentation, demo and report was based on the quality, completeness and delivery of the integrated solution demo, the quality of the presentation and delivery and the quality and completeness of the project report. At the end, the students were placed by the instructor into one of three categories: consistently demonstrated excellent contributions to the success of the project (19%), consistently demonstrated strong contributions to the success of the project (54%) and consistently demonstrated weak contributions to the success of the project (27%). The number between parentheses is the percentage of the students fell in the corresponding category.

During the week preceding the semester finals week, the entire class performed a dry run of their end-to-end solution where the students had the opportunity to practice their presentation and tune up their demo. This was the first time the entire class participated as ONE unit in delivering an end-to-end demo of the integrated solution. Although, they had been partially successful in the following pairwise integrated demos (1) the back-end and the central components, and (2) the front-end and the central components, they realized during the dry run how much they have under-estimated what could go wrong in integrating all three components. During the finals week, the entire class presented their working prototype to the course advisor, consultants, unofficial advisor and lab technician. At the end of their final presentation, the students shared their key learning and feedback regarding their experiences in such a non-traditional undergraduate course.

4. Case Study Results & Analyses: Students Technical Achievement Perspective

In order to describe the system that was implemented, it seems logical to go in the direction that the collected data flows. Starting with the back-end component an Arduino UNO microcontroller [7] was hooked up to a computer via a USB cable as depicted in Figure 3. Connected to the Arduino microcontroller were two infrared LED emitters (VSLY5850) and two infrared receivers (LTR-4206E). Between each emitter/receiver pair was an infrared beam emitted by the LED. The beams were set up so that they were about a foot apart, and parallel to each other. On either side of the door frame were an LED emitter/receiver pair, this setup prevented interference between the beams, as the LEDs were identical and were transmitting on the same carrier frequency. The Arduino monitored and compared the receivers' voltage level to a user configured threshold. When the threshold was crossed, Arduino registered the time and compared that time with the time associated with the other receiver. Depending on which beam was broken first, someone was either entering or exiting the room. The Arduino would post a '1' or a '0' into the database server if someone entered or left the room, respectively.

Due to a number of issues with the Arduino Ethernet Shield [8], the back-end sub-team in collaboration with the lead member (who will be referred to as "the glue" later on) of the front-end sub-team decided to replace a direct connection from the Arduino to the Desktop running the web and database server with a computer running a simple Python script. The Desktop was configured to run a number of php request handler files, web server and database server software components. The Python script is designed to map the '1' or '0' string sent on the serial port from the Arduino board into an HTTP GET request to the php request handler on the Desktop which was designed to increment or decrement the value of the Head Count data element for the respective room in the database.

The database that the primary central sub-team (see next section) created was implemented using MySQL on the Desktop which is a machine running Linux Ubuntu server. Another version of the database was implemented on Windows Server 2008 by the secondary central sub-team. The database included a table that contained LabName (the primary key), the PeopleNumber in a room, and the Schedule. Once a specific lab was selected by the user, the front-end component was able to retrieve the number of people recorded in the MySQL DB on a near real time basis using an autonomously generated HTTP GET request to a php request handler script running on the Desktop.

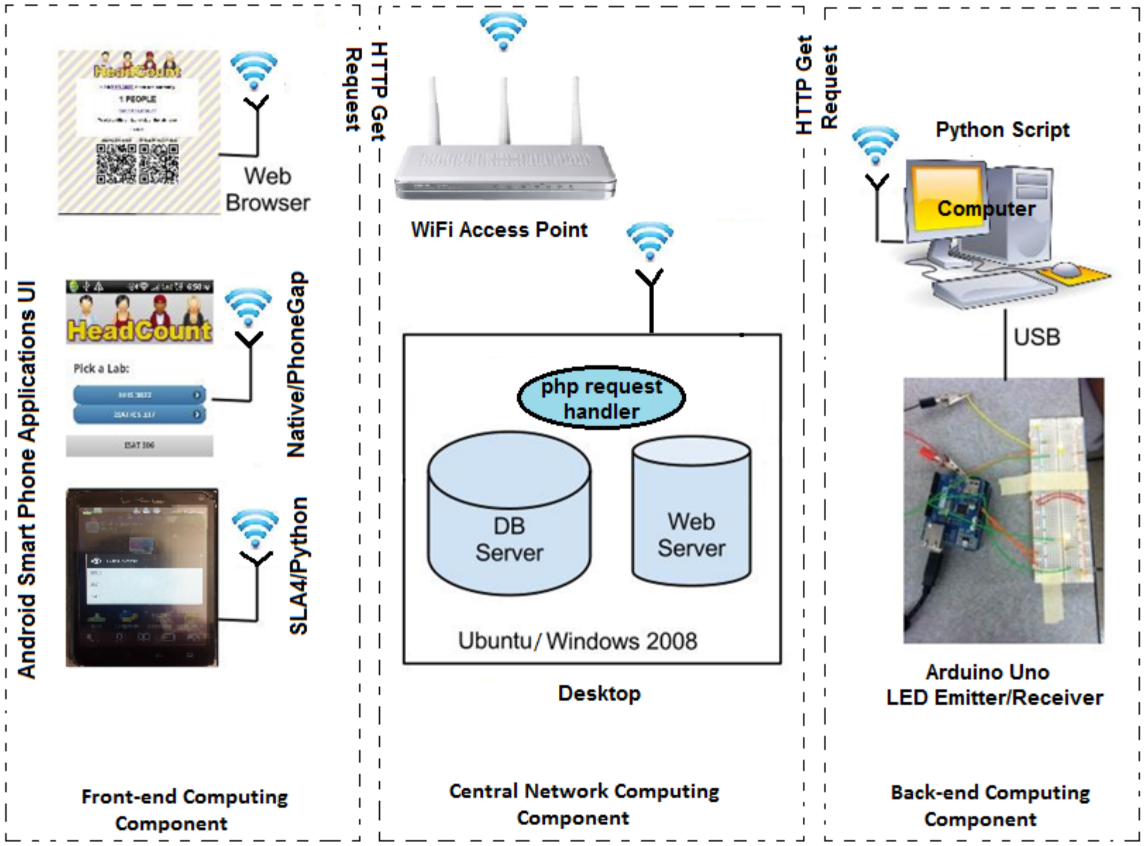

Figure 3. HeadCount Application Network Architecture

The front-end sub-team created three versions of the front-end user interface: Web Application (see Figure 4a), Android Mobile App using SL4A/Python and Android Mobile App using both native and cross-platform development environment, known as PhoneGap/Cordova [9] (see Figure 4b). From the mobile or web application, the user could select what room they are inquiring about, and see how many people are in that room at that time.

Figure 4a. Web-based Application UI

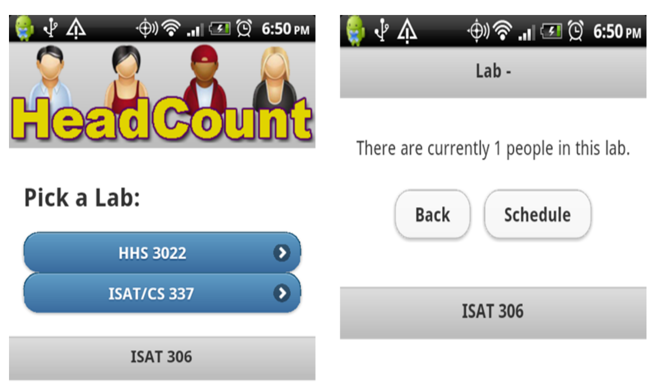

Figure 4b. Android Mobile Application UI

Overall, the class created the following sets of software code:

- ***"Processing" [10] Arduino Uno*** – determine if an object has crossed the beams in the "in" or the "out" directions;
- ***Python script*** – mapping serial data stream into internet protocol based stream;
- ***php request handler codes:***
  - ***request.php code*** – connect to MySQL database, parse request, query database to insert or retrieve data depending on the request type (lookup, plus one, minus one), respond to request with data or error;
  - ***getlabs.php code*** – required by both the web-based and PhoneGap-based applications;
  - ***jsonptest.php code*** – JSONP [11] and CORS [12] (with cross-domain configuration) enable a web page to request data from a server in a different domain than that served the initial page; this was required by the Web and PhoneGap-based Mobile application.
- ***index.html/css/js codes*** – for both web based application and PhoneGap based mobile app on Android;
- ***SL4A/Python [13] code*** – for Python-based Mobile App on Android which makes use of request.php

The above list of codes has been provided to demonstrate the breadth of knowledge gained by the students in the course of developing this project.

5. Case study Results & Analyses: Students "Real-World" Learning Perspective

Although the project idea the students selected appeared valuable and relevant, defining the scope of the project proved to be difficult, as the students were asked to implement a solution with little to no information about possible new skills they may need to acquire and apply. Students were asked to pick an end target, and commit to that, not knowing whether or not they're promising too much.

Another key issue was the lack of knowledge of how to practically map an end-to-end functionality into specific functionality for each of the three computing components of the end-to-end solution. As a matter of fact, most of the time, to the course advisor's dismay, the scope of what individual sub-team was willing to commit to was the governing factor of the end-to-end functionality the entire team was willing to pursue.

In the early stage of working on the project, the students' expectations and confidence for successfully accomplishing the target goal of their proposed project started to deteriorate. A solution to this issue came in the form of allowing the students to redefine the scope of their project halfway through the semester, that is, when they had a better grasp of the amount of work necessary to complete their tasks. This mid-way adjustment process was applied a number of times throughout the rest of the semester. These adjustments included changes to the organizational makeup of the class. See the final organizational chart in Figure 5.

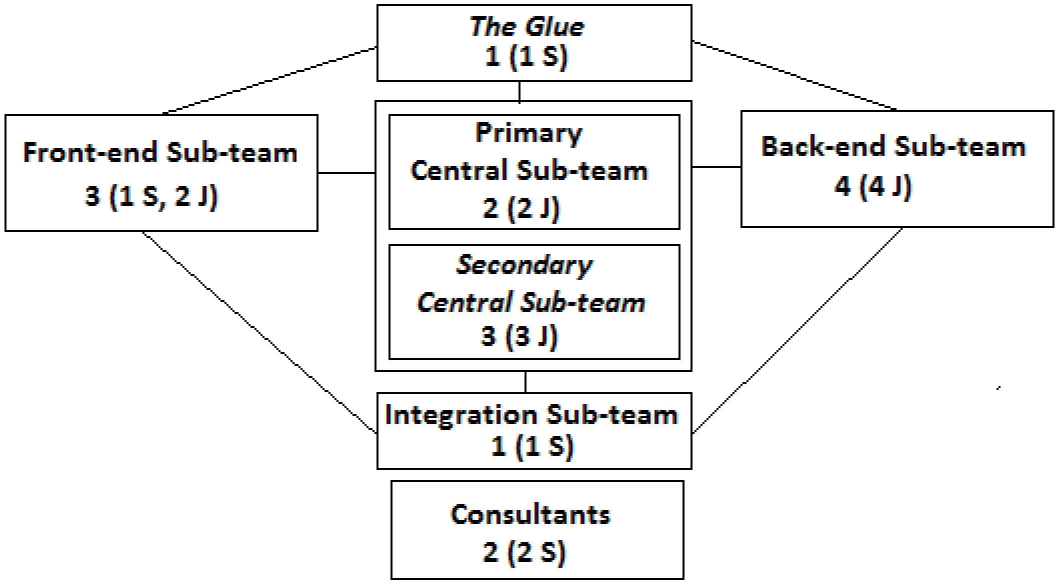

Figure 5. Final Organizational Chart (S: Senior, J: Junior)

The Back-end, Central and Front-end Sub-teams

Each sub-team struggled with many individual and unique challenges that they had to overcome to make this project work. However, it was felt that the learning came not only from learning about how databases function, how to create a mobile phone application or how to interface the microcontroller platform to the network, but also how to manage the *human resources* that were at the disposal of each sub-team in an efficient manner, and how to facilitate the interaction of the three sub-teams in a practical, timely and effective way.

Efficiently utilizing the human resources of each sub-team was difficult at first. For example, with the back-end sub-team, all members were almost constantly working as one unit on the same problem, as opposed to using a more divide-and-conquer approach that would have allowed the research of multiple problems at one time. This divide-and-conquer approach could have also supplied a more diverse set of solutions to a given problem. Another advantage of this approach is that the members of the group can learn from other members of the group. Students learned that it is not necessary for every member of the group to be deft at every aspect of their group's part of the project. It is more effective to divide the available resources to research different solutions to current problems.

Another issue the class ran into was that some team members did not have the necessary skill set to complete the tasks they committed to or assigned to by their sub-team. For instance, the members of the central sub-team that worked previously with databases had only worked with Microsoft Access databases, and had no experience creating a database for a website outside of integrating Access with Microsoft Visual Studio. An approach to resolving such an issue is to borrow members from other sub-teams who have the knowledge and experience necessary to complete the tasks at hands. This approach was applied in the central sub-team's case, where one student (was referred to as "the glue" in Figure 5) in the front-end sub-team, who had some experience setting up apache, MySQL, and php development environment, stepped in and guided the central sub-team towards successful completion of their committed task. However, it should be noted that the central sub-team was unwilling, for many weeks, to step up and admit that they were totally overwhelmed and had no clue as to what to be done. It also became clear from this experience that the weekly status report did not have the appropriate mechanism to encourage the students to (1) own up in a timely manner to the fact they were *ill-prepared* to tackle certain tasks and (2) to seek *immediate* help and assistance from the available resources. Unfortunately, this situation had left the central sub-team with bruised self-confidence that they were not able to recover from. They were not able to break out of being depended on “the glue” student for the rest of the semester.

Apart of the central sub-team, recognizing and utilizing outside resources was something that the other sub-teams became experts at throughout the semester. When pushed into a situation where some students possessing limited to no experience working with their objectives, students needed to quickly adapt and discover where they could find the information that they were looking for. This information came in many forms, from seeking help from other professors, the consultants, to online books, to online communities like “Stack-Overflow” web site. It is worth noting that the consultants were instructed by the course advisor not to perform any of the implementation tasks themselves. Students could go to these consultants for help, but if the students contracted them to develop code, then the students had to negotiate such a request with the course advisor. The course advisor had hinted about exchanging grade for consulting time in developing code, but that idea was never implemented. Moreover, it became clear very quickly that the consultants had no time or incentive to dedicate outside class time to develop code or perform any of the tasks themselves. Overall, many of the students felt that the consultants were definitely worthwhile and effective in their fundamental role of guiding and assisting.

The Integration Sub-team

The end-to-end solution integration was probably the most difficult task the students faced in completing this project. The primary task the integration sub-team, had on their hands, was to work with each of the sub-teams to make sure they were keeping integration with other sub-teams in mind and as part of the development of their functionality. The integration sub-team was supposed to get the sub-teams together and iron out the integration issues. This task required that the integration sub-team members be highly respected by and credible in the eyes of the members of the other sub-teams.

At the beginning of implementing the project, there were two integration team members. Their job each week was to make a compilation of the other sub-teams drawings, code, status reports, as well as to maintain a schedule and timeline for the project. Also, they were supposed to make Google docs where all the students would be able to add updates they had for their part of the project. Unfortunately, the selection of these individuals was based on the wrong assumption. One of them claimed that he had done some project management in prior projects; however, it was in a Manufacturing & Engineering course. It was also discovered later on that he selected the Integration sub-team assignment to hide his lack of hands-on expertise in most of the networking and computing development. The other team member was not any better off both from the technical and managerial perspectives. In summary, neither of the members of the integration sub-team fully understood the way the entire project functioned, or how each sub-team component was to be interconnected, and therefore collaboration and interactions among the other three sub-teams suffered significantly and reflected negatively on the entire project.

The real solution "glue" and those who had to be "separated"

This lack of leadership from the integration team resulted in other members stepping up to take the integration of the project into their own hands. These members bypassed the integration team to work with other sub-teams directly. This direct communication created a situation where some members were no longer bound to their sub-teams and was viewed as the "real" integration sub-team at least from a technical perspective. In specific case, there was one student, who was very skilled in his area as well as in other areas. This particular member had an abundant knowledge of MySQL, php, and in general, how different pieces of software can be made to communicate with each other. He ended up being "the glue" among all the sub-teams.
The students found that re-grouping (see Figure 5) allowed them to be more efficient, and created many situations where one student could learn from another student who was working on something completely different.

Towards the middle of the semester, it became clear that the project was heading for utter failure from an end-to-end integration perspective. At this point, with input from most of the class members, the class advisor intervened and split off two of the four members of the central sub-team to form their own parallel "secondary" central sub-team (see Figure 5). They were to focus on implementing the central network component using Windows 2008 Server as a potential alternative to the Linux based central network component. In addition, one of the integration team members was re-assigned to the secondary central sub-team to configure HAProxy TCP/HTTP load balancer [14] on Asus RT-16N Wi-Fi Access Point to balance the HTTP traffic across the web servers running on Linux and Windows 2008 servers. Unfortunately, these two new functionalities were not fully realized. However, the creation of that secondary sub-team and the stepping up of the "the glue" student in that the student was no longer bound to a specific sub-team appear to have motivated the class to achieve their goals and specifically motivated the primary central sub-team to complete their initially committed goal which was crucial in delivering an end-to-end integrated solution.

Remarkably, at the end of the semester, the team pulled it off and delivered a working prototype of the end-to-end solution! However, some of the students believe that the delivered integrated solution would have not been possible to realize, if it was not for the student known as "the glue."

6. Conclusions, Key Learning and Recommendations

Overall, the students met their final goals as defined in section 2. They defined a project that they could all relate to, they created a solution with the expectation that it could be achieved, and it was an end-to-end network solution as described in section 4. Also, the class satisfied the criteria of success given in section 2. The results described in sections 4 & 5 validated that the design approach of the course given in section 3 is well thought out and viable for undergraduate multi-disciplinary large team one-semester hands-on network based projects. However, there is a number of key learning that should be kept in mind and recommendations that should be considered for future offerings of this project-based course.

Although, the Expectancy-Value [4, 5] model appears to be a necessary ingredient for a successful large-team, one-semester hands-on end-to-end projects, it was not sufficient for the success of this project case study. One of the key contributing factors to effective expectation management of whether the team can achieve their set goals was the selection of effective integration sub-team members. Also, the organizational structure should be flexible enough to enable natural lead members to be identified and promoted to the role of technical integration sub-team members. However, these technical leads should be supported by at least one member for project, milestones, timeline and documentation management. See Figures 2 and 6 for the initial and final organizational charts. Another key factor that has not been addressed in this case study and should be added as a third dimension to the Expectancy-Value model is cost [15]. That is, the amount time and effort spent by the students in realizing their committed goals.

It appears that the four demos and exercises selected and administered by the course advisor were adequate for exposing the students to the current and up to date networking and web technologies. However, they missed the mark on preparing the teams early on for the specific tools that are required to get them started on the right track for their specific tasks. Therefore, the topics of the demos and exercises should be collaboratively and jointly decided upon by the entire class based on the project idea selected. In addition, the students should play an active role in delivering these exercises with the help of the course advisor, other participating co-advisors and the consultants. This approach will also help the students to describe in more specificity their skills inventory and in a timely manner identify those who may be potentially "the glue" of the project and those who may need to be "separated" for special training and tasks. In addition, the course advisor should offer an example of best practice in the area of system integration and/or management.

As we mentioned above, we are planning to improve the process and form of soliciting students' skills inventory. In addition, we believe that including, at the onset of the project, learning styles in higher education [16] assessment as part of the skills inventory exercise may help the instructor, the consultants and the entire class to identify individuals who are prime candidates for lead roles on the sub-teams and more critical those who may be excellent candidates for the system integration tasks.

It should be highlighted that some students took much more from the course than other students. Students with a greater intrinsic motivation seemed to be more willing to learn new things, and accomplish goals set by the team (including their contributions), enabling other members of the team to be as active in the process of learning, and the creation of the solution. Although, the peer-directed learning opportunities were not part of the original design of this case study, it should be included and nourished along the lines of what is described in [17].The integration sub-team, in this case study, seemed to have gained the least out of the project overall. They may have felt unmotivated because they had no hands-on tasks to do, but this didn't have to be the case. The members of the integration sub-team could work with other sub-teams, particularly, on implementation of interfaces between computing components. This should help with the integration sub-team's overall understanding of the project, and boost their intrinsic motivation, as they would feel they directly contribute to the project. The integration sub-team membership should be reserved to students who seem to be enthusiastic about the project, are internally motivated to complete the project, and have strong leadership and presentation/ report skills or those who are willing to develop these skills over the course of the project. The integration sub-team needs to be able to communicate effectively with the students individually and the entire team collectively. They should analyze every team member's class schedule to determine a suitable weekly meeting time out of class to work on the project. They should be able to foster spirit of collaboration and should be able to deal with complaints from the entire team and the course advisor, and take constructive criticism to heart.

The assessment process based on weekly deliverable of demo-able functionality and action items register played a major role in keeping the project on track and allowing the students to constantly be aware of where the project as a whole stands. These weekly status reviews also enabled the class to adapt and adjust from both resources and scope perspectives. It is highly recommended that the assessment process adhere as closely as possible to the *sprint* concept defined in the Scrum framework [18]. It is our intentions and plans to develop a quantitative and data-driven assessment that we will apply for future classes to improve the level of success predictability of the processes and approach presented and discussed in this paper. On a related topic, we believe that the grading process should be enhanced to include a step where each student meet with and present to the instructor and consultants the significance of their contributions to the success of the project.

Acknowledgment

The authors wish to acknowledge the ISAT 306 Section 1 students who participated in this course case study, ISAT Department for providing laboratory and funding support. In addition, we would like to thank the two TAs, John Catron and Hunter Grenfell, Dr. Benton (the unofficial advisor), Mr. Joe Rudmin (Lab Technician) and numerous online community sites. The entire class would like to recognize the unique and crucial contributions of "the glue" (Adam Maas) without his drive and willingness to help other sub-teams beyond his, the project would not have been as successful. In addition, the authors would like to acknowledge Sarah Osorio's contributions to initial drafting of this paper, Dr. Morgan Benton for his extensive and valuable input and Dr. Anne Henriksen for her initial and final review of this paper.

Bibliography

1. Department of Integrated Science and Technology (ISAT), "Bachelor's Degree in Integrated Science and Technology (ISAT) at <Name of University>," CISAT Creative Services, March 9, 2011. <URL of University>, Accessed: December 2012.
2. Khorbotly, Sami, and Khalid Al-Olimat., "Engineering Student-Design Competition Teams: Capstone or Extracurricular?" IEEE Xplore (2010): 1-6. Print.
3. Goldberg, J.R., Active Learning in Capstone Design Courses [Senior Design], Pulse, IEEE Volume 3, Issue: 3, Publication Year: 2012 , Page(s): 54- 57
4. Center for Teaching and Learning, The University of Texas at Austin, Teaching Resources, Design Your Course, Design for Motivation, <http://ctl.utexas.edu/teaching-resources/design-your-course/design-for-motivation/>, Accessed: December 2012.
5. Schunk, D. H., Pintrich, P. R., & Meece, J. (2008). Motivation in education: Theory, research, and applications, 3/E. Columbia, OH: Pearson.
6. ISAT 306, Section 1: Classroom Availability Project Report, Spring 2012.
7. Arduino Uno, < http://arduino.cc/en/Main/ArduinoBoardUno>, Accessed: December 2012.
8. Arduino Ethernet Shield, < http://www.arduino.cc/en/Main/ArduinoEthernetShield>, Accessed: December 2012.

9. Easily create apps using the web technologies you know and love: HTML, CSS and JavaScript, <http://phonegap.com/>, Accessed: December 2012.
10. Processing, an open source programming language and environment, <http://processing.com>, Accessed: January 2013.
11. JSONP or "JSON with padding", http://en.wikipedia.org/wiki/JSONP, Accessed: January 2013.
12. CORS, Cross-origin resource sharing, http://en.wikipedia.org/wiki/Cross-origin_resource_sharing, Accessed: January 2013
13. SLA4/Python, android-scripting, Scripting Layer for Android brings scripting languages to Android, http://code.google.com/p/android-scripting/, Accessed: January 2013.
14. The Reliable, High Performance TCP/HTTP Load Balance, < http://haproxy.1wt.eu/>, Accessed: January 2013.
15. Anderson, Patricia N., "Cost Perception and the Expectancy-Value Model of Achievement Motivation", Paper presented at the Annual Meeting of the American Educational Research Association (New Orleans, LA, April 24-28, 2000),
16. R.M. Felder and R. Brent, "Understanding Student Differences, "An exploration of differences in student learning styles, approaches to learning (deep, surface, and strategic), and levels of intellectual development, with recommended teaching practices to address all three categories," J. Engr. Education, 94(1), 57-72 (2005).
17. Andrew McNally, "Work in Progress - Peer-Directed Learning in a Project Based Model," Frontiers in Education Conference (FIE), 2011.
18. The official scrum rulebook, http://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum_Guide.pdf13>, Accessed : December 2012