

A TWO-SEMESTER CAPSTONE SEQUENCE – PREPARING STUDENTS FOR THE REAL WORLD

Billy Farmer, Department of Computer Science, Texas Christian University, 2800 S. University Dr., Ft. Worth, TX 76129, 817-257-7166, b.farmer@tcu.edu

James Comer, Department of Computer Science, Texas Christian University, 2800 S. University Dr., Ft. Worth, TX 76129, 817-257-7166, j.comer@tcu.edu

Tom Nute, Department of Computer Science, Texas Christian University, 2800 S. University Dr., Ft. Worth, TX 76129, 817-257-7166, t.nute@tcu.edu

Lisa Ball, Department of Computer Science, Texas Christian University, 2800 S. University Dr., Ft. Worth, TX 76129, 817-257-7166, l.ball@tcu.edu

ABSTRACT

A senior capstone sequence is a vital concluding element required of most students majoring in a technical discipline. Much of the course work in the major of a technical field is designed as a building process. It is assumed that students entering a major know little about that discipline except that it looks interesting. Based upon this assumption, the faculty undertakes to teach the basic skills, principles, and underlying theories of the discipline to them.

The goals and characteristics of the Computer Science senior capstone sequence, at Texas Christian University (TCU), including how it is administered will be discussed here along with two of the projects developed to fulfill this requirement.

INTRODUCTION

TCU Computer Science and Computer Information Technology students are asked to complete a variety of assignments as they progress through the requisite course work required for these majors. Ideally, each assignment is designed to help build a growing foundation of basic skills and an understanding of the theoretical basis of the discipline. While the immediate intent of the faculty is to impart an adequate understanding of the subject matter that is covered by the various courses, the underlying goal is to prepare students for further course work in the major and eventual employment in their chosen field.

Recognizing that the development of software to meet the complex needs of today's industries is a very different activity from writing programs to fulfill coursework requirements, the Department has long required students to satisfy a research component. The faculty's intent is to provide an environment that is designed to integrate learning from their various courses in the major with information derived from courses outside the major. Over time, this requirement has matured into a two-semester capstone course sequence that is designed to provide TCU students with an experience that approximates the scale, intricacy, and difficulty of the projects that await them in industry. It was felt that a single semester course was inappropriate since it did not allow the students adequate time to apply the skills learned in their various classes.

At present, the senior capstone sequence is a graduation requirement for all Computer Science and Computer Information Technology students at TCU. The two courses that comprise this sequence for Computer Science majors are *Software Engineering* and *Senior Design Project*. Likewise, Computer

Information Technology students are required to complete a two-course sequence consisting of *Issues in Project Management* and *Senior Development Project*. Each of these course sequences is an essential culmination to their respective majors. The faculty feels that without the senior capstone sequence, the process of enabling students to continually attain to higher levels of maturity in the discipline falls short of the end goal of preparing students to solve the problems they will be faced with when they are employed in the real world.

THE CAPSTONE SEQUENCE

Students must demonstrate, in the capstone sequence, that they have achieved the academic goals established by the Department along with their ability to apply these principles to the implementation of complex, industrial-strength software applications. Projects are designed to encourage students to integrate learning from their various courses in the major with information derived from courses outside the major. As such, projects have been developed that have involved interdisciplinary partnerships. These partnerships have involved representatives from other university departments, industry, and government agencies.[1]

The present capstone sequence consists of a first course that is designed to equip students with the tools needed for the application of sound software engineering principles and practices. In this class, students are allowed to define their own projects or to select from projects that have been proposed and are supervised by the department's faculty. The second course emphasizes team collaboration — the objective being for students to design and implement the project chosen in the first course.

CoSc 40943 - Software Engineering or CITE 40643 – Issues in Project Management

The first of the two courses in the sequence is structured to develop the knowledge, skills and experience needed to design, develop, implement and maintain large, complex software systems in a team environment. It includes a study of software development life cycles, with emphasis on the analysis and design of software systems. A significant team project is required necessitating students to use and enhance their technical, writing and presentation skills and to learn how to function in a technical team environment. The project is completed in CoSc 40993, Senior Design Project or CITE 40993, Senior Development Project. [2]

CoSc 40993 - Senior Design Project or CITE 40993 - Senior Development Project

This is the second course of the Department's two-course capstone requirement. For most students, the project is a continuation of the one begun in the Software Engineering or Issues in Project Management course from the prior semester.

Emphasis is placed primarily on milestone deliverables, presentations (internal and external to the department), technical papers, ability to work in a team, initiative, and cooperation with other groups (e.g. customers, collaborating university faculty/students). [1]

COMPUTER SCIENCE AND PSYCHOLOGY

An ongoing, collaborative project exists between the TCU Psychology and Computer Science Departments. To date the project has resulted in the development of two separate software applications, *Under Construction* and *Downward Spiral*. This collaboration began with the Psychology Department's desire to enhance the value and efficacy of some of their products.

Using their research, conducted on the rehabilitation of substance abusers, the TCU Psychology Department has developed a form-based activity and a board game with the intent of improving the overall effectiveness of an abuser's treatment. During the 2003-2004 academic year, the Computer Science Department was asked to assist in a project to computerize both products. This effort provided an easier method of packaging and distribution of these products as well as improved opportunities to further the Psychology Department's efforts of research in the area of substance abuse treatment. [3]

Under Construction Project

The collaborative effort began with Computer Science students working independently on each of three separate applications that have ultimately come to comprise the *Under Construction* application. These were the *Tower of Strengths* form, the *Parts of You* form, and the *Building Blocks* form. During development it was determined that the computerized version would be more effective if it was developed as a single integrated application consisting of multiple parts. Today, these three forms, along with a fourth, *Putting it Together*, comprise an integrated application with four activities. These activities must be completed in a specific sequence. Their intent is to help participants identify their strengths and weaknesses, which can be built upon to develop new strengths while using their current ones to solve certain problems and situations they face in life.



It quickly became clear that a single semester would not provide adequate time for students to design a quality product. A decision was made to develop *Under Construction* as an integrated application. Additionally, the Department agreed to develop a second application based on the *Downward Spiral* board game. The Computer Science Department concluded that these two activities warranted the establishment of a team to work on these applications as a single Senior Design/Development Project. It was believed that the learning opportunities associated with managing the design and development of these two applications simultaneously would provide the Department's students with experiences worthy of the six semester hours of credit that they earn from this sequence. [4,5]

Downward Spiral Project

Downward Spiral began as the result of the Psychology Department's research in the incorporation of interactive games into the treatment of drug abuse patients in rehabilitation centers. The purpose of the project was to test the effectiveness of the game in encouraging teamwork and friendly competition amongst players while emphasizing the importance of making good life decisions.

The premise of the game is to illuminate the “downward spiral” that life tends to take when individuals continually abuse drugs, violate the law, and make poor decisions in general. In order to improve the effectiveness of the game, the scenarios depicted in the game are taken from real life accounts of drug abuse patients that were researched and interviewed by the Psychology Department staff involved in the project. The game was originally designed as a turn-based board game that plays out much like the popular game of LIFE™ by Milton Bradley. [6] Each player must follow a spiral shaped path around the board until they either win or lose the game.



Along the way, players face consequences occurring as the result of bad decisions that get progressively worse as they continue along the downward spiral of their “life” in the game. These consequences include health problems, impaired judgment, family crises, financial misfortunes, and many others. Upon nearing the end of their downward spiral, players may face the ultimate consequence of the death of their player – thus losing the game, or they may discover the err of their ways and take the road to recovery, thereby winning the game.

Under Construction provided some interesting problems for students to work on, such as the development of a user interface that meets specific customer requirements, data storage of player choices, background processing of previous choices of a player used to limit available choices in current play, etc. *Downward Spiral* added the element of animation, far more background processing requirements, client/server communication and processing, and much more. The combined requirements of these two applications have provided challenging learning experiences for two groups of seniors. However, more remains to be done in order to improve existing functionality and to incorporate a variety of new features. The value of this project can be readily seen in the comments of the students who worked on the application during the 2005-2006 academic year. [7,8,9,10]

STUDENT RESPONSES

Reflection can be a valuable activity in the learning process. This activity gives students the important opportunity for self-evaluation of their performance and, perhaps more importantly, a chance to understand how they have benefited from the experience. At TCU, Computer Science and Computer Information Technology students are required to include a “lessons learned” section in the documentation portion of the final deliverable for their respective senior project. Students involved in this process during the 2005-2006 academic year expressed the following comments about their experiences [10]:

“... I did learn the importance of a content management system and a way to track changes so I do not have to merge files, and manually check to see where people are working to make sure they do not step on each other’s toes. Overall I think the most important idea to take from this class is that everything works better if we all work as a team.”

~ Michael Pyle

“Downward Spiral taught me a lot of things. I have not been confident using networking capabilities of programs including sockets, streams, and threads. After having written the server I feel I have [a] far better grasp in what is required to have these pieces of the program function. The real difficulty though was not writing the server itself, but interfacing it with the chat client. Travis and I had set up the slash delimited commands for the communication between the chat server and chat client together, but after implementation, that did not all work the way either of us necessarily expected when put together. This collaboration in program projects is the true challenge of programming, not the languages and concepts. Furthermore, after inheriting Downward Spiral from the previous year’s group, I learned a new appreciation for commenting code. Before I never had to comment code for the benefit of others, but now after looking at code that probably could’ve used more comments for the benefit of our team and creating code that needs comments for the benefit of future teams, I learned the true value of commenting. Overall, this project has been a lot of fun and an interesting learning opportunity for a glimpse as to what a software engineering project could be like in the real world.”

~ Steve Calder

“Aside from learning a little about the C# programming language and about Microsoft’s Visual Studio, I walk away from this course with a healthy respect for software development as a whole. I believe I have a much greater understanding of the different facets of project management and working with a team in general, not just where software design is concerned. In particular, I believe I have much more respect for the concept of documentation. More than likely, when we all enter into industry and start our separate careers, we will be assigned to a project that has already undergone considerable development and we will have to efficiently pick up where previous developers left off. Without adequate documentation on the part of our predecessors, the task of completing a project would be daunting. So with this in mind, I and my teammates have strived to be as thorough as possible in our documentation and in the intuitiveness of the code itself in order to better provide a stable development environment for future senior design students. For the future, I will strive to carry my understanding of proper documentation concepts and of industry standards and expectations with me into my career in order to ensure quality work on my part.”

~ Jake Thompson

“I’ve learned that trying to fix bugs in programs when there is good documentation and lots of comments in the code is fine but trying to fix bugs in parts of an application that has very confusing documentation and no comments in the code is a nightmare. I’ve also learned that Microsoft Access does not work well with .NET when used extensively. No matter how much planning and designing you do, there will inevitably be something that gets changed when the application gets developed. When you pick up a previous project you may dislike the technologies used to develop the software but you have to live with it, and it’s very unlikely that you’ll be scrapping large portions of what’s already there. It’s also very difficult to keep projects on schedule...”

~ Tyler Reainthong

“Over the course of this project, I’ve come to realize there is much more to writing software than simply programming. I now better recognize the importance of proper documenting, not only for others that

might work on the project, but also for yourself to make sense of what you have done when you need to revisit it later. I've learned that the client will keep coming to you with a few more small requests and that you can't always give in and attend to those requests. I've also learned a lot about group dynamics, and how to rely on others, in addition to learning a new programming language."

~ Bara Gunnarsdottir

"This senior design project has taught me many invaluable lessons in the realm of not only code design, but teamwork and communication as well. Obviously, this project helped my development as a software engineer, serving as the first time I was asked to write "production-level" code and a program with a significant amount of length to it. It was definitely a shock right out of the gate to learn that I'd be working on a project that couldn't be completed in a few days or a week, but one that could possibly take the entire year and still not be finished. Hence, I learned a lot about how to scale down a large project and focus on getting individual parts working instead of trying to get the entire program to work in one fell swoop. As well, I learned a great deal about interacting with many different people on a single large assignment, and what it takes to come together to produce a worthy program. Due to the large size of our design team (seven programmers), we were forced to try and split responsibilities up in such a way that we weren't stepping on each other's toes or causing conflicts within other parts of our code. This meant that communication was crucial amongst teammates, and we needed to find a way to consistently see eye-to-eye with each other when it came to describing what had been done and what needed to be done. Finally, this project also brought along the challenges associated with having seven people with seven different sets of strengths and weaknesses. This meant that I needed to be confident about what I was best at, and find a way to apply it to our project..."

~ Adam Weiser

"For my experience(s) of being in Software Engineering/Senior Design, it can be summed up in one word: Adjustment(s)..."

~ Travis Garnett

SUMMARY

A good measure of the importance of the capstone sequence in providing students with the preparation they deserve from their academic endeavors can be seen in their own words about their experiences. These experiences should give students a clearer understanding of the value of the precepts that they were taught in the course work leading to and culminating in the capstone sequence. Clearly, students should leave the university with a deserved sense that they are prepared for the challenges that await them in whatever new endeavor they may undertake, graduate studies, or industry. The authors believe that the Department's capstone sequence is meeting those goals.

REFERENCES

- [1] "A Student Research Symposium at Texas Christian University – How it has Benefited the Computer Science Department", James Comer, Tom Nute, Billy Farmer, Lisa Burnell, and Robert Roggio, Proceedings of the 2006 International Conference on Frontiers in Education: Computer Science and Computer Engineering, Las Vegas, Nevada, FEC4163.pdf, June 26 –29, 2006. CD-ROM: ISBN #1-932415-99-8).
- [2] COSC 40943: Software Engineering, Dept. of Computer Science, Texas Christian University, <http://www.cs.tcu.edu/people/professors/ball/40943wp.html>.

- [3] “Simian Games”, Texas Christian University 2005-2006 Senior Design Team, <http://frogware.mytcu.com/index.html>.
- [4] “Simian Games”, Texas Christian University 2005-2006 Senior Design Team, http://frogware.mytcu.com/under_construction.html.
- [5] “Under Construction - Motivating individuals to build with strengths” Department of Psychology, Texas Christian University, <http://www.psy.tcu.edu/acr/tower.htm>.
- [6] “The Game of Life”, Milton Bradley, http://www.hasbro.com/default.cfm?page=ci_history_life.
- [7] “Simian Games”, Texas Christian University 2005-2006 Senior Design Team, http://frogware.mytcu.com/downward_spiral.html
- [8] “The Downward Spiral Game”, Department of Psychology, Texas Christian University, <http://www.psy.tcu.edu/acr/spiral.htm>.
- [9] “Preliminary Analysis of the Automation of *Downward Spiral*, a Drug Abuse Treatment Program”, Tom Nute, Billy Farmer, James Comer, Michael Czuchry, Proceedings of the Fifth Hawaii International Conference on Social Sciences, Honolulu, Hawaii, May 31 – June 3, ISBN #1539-7300.
- [10] “Downward Spiral – a Comprehensive Project Overview (P6.doc)”, Senior Design Team, Texas Christian University, Spring, 2006, <http://frogware.mytcu.com/project/P6/P6.doc>