Starting in the Middle: Design and Implementation in the Information Age

Edward Sobiesk and William K. Suchan
Department of Electrical Engineering and Computer Science
United States Military Academy

Innovation in Engineering Education

Abstract

In order to create more realistic and complex learning experiences in a mid-level Information Technology course, we made a shift in pedagogical focus from building new systems to extending existing systems. By starting course projects "in the middle" instead of from scratch, we were able to heighten student interest in course material as well as achieve higher levels of learning. In this paper, we discuss how we applied this concept over a series of projects, and how those changes led to improved student motivation and learning. We also describe challenges with this paradigm and provide tips on how to avoid potential pitfalls.

Introduction

Building upon the work of others is commonplace in industry and at higher levels of academia, but students in many engineering disciplines frequently find that their initial design and implementation experiences have them build relatively trivial products from scratch. In order to create more realistic and complex learning experiences, as well as to increase the interest of our students, we wanted them to be able to build reasonably sophisticated systems, but we found it difficult for them to do this when they started with a blank slate. Our solution was to shift the pedagogical focus from building new systems to extending existing systems.

For course projects, we give our students working systems as their starting point. As a result of this simple paradigm shift, we immediately realized many benefits, not the least of which is that our students are learning more and are doing better work. The concept of implementing useful systems instead of "toy" implementations provided increased motivation and allowed for a multifaceted, realistic experience that minimizes much of the overhead involved in initial system standup.

From an assessment standpoint, we are now able to evaluate student comprehension at higher levels. Because students must first understand the existing system in order to expand it, we also are able to assess the same lower level knowledge that we were able to assess in the past. The difference, however, is that we can now assess components of higher level comprehension that we could never have reached under the "build from scratch" paradigm.

Our students now leave our course with the confidence and ability to do things that we feel they are more likely to face in the future, namely, to enhance or extend existing systems. In the Information Age, it is no longer necessary to re-invent the wheel. The globally networked world offers countless opportunities for collaboration and for building upon the work of others. The future success of our students will rely on their ability to modify existing solutions to match evolving needs. Our students now have much better experiences upon which to draw from to help them accomplish this.

In this paper we discuss the successes of this approach, as well as some challenges and potential pitfalls. Our intent is for others to build upon our work and to enjoy the same level of student improvement that we are currently experiencing, while avoiding possible obstacles.

Course Description

The course described in this paper is called IT305 – *Theory and Practice of Military IT Systems*. It is an intermediate level, semester-long Information Technology (IT) course taken by almost 800 students each year as part of the Bachelor of Science curriculum at the United States Military Academy. In the course, students learn about IT principles and practices for acquiring, communicating, managing, and defending information, and how the Army accomplishes these tasks to achieve Information Dominance. Students study the IT concepts and techniques that will facilitate their professional success and, additionally, cover key aspects of the military IT infrastructure. A major intent of the course is to give students the technological competence and confidence to enable life-long learning in the IT domain. IT305 is a hands-on learning experience – students complete numerous in-class exercises and labs as well as four team projects. The philosophy behind the evolution of these four projects is the primary topic of this paper.

Concept Overview

Several years ago, a student in an introductory computing course described the experience by remarking, "They take easy problems and make them really, really hard." This is often the case in undergraduate classrooms. Students start projects from scratch and significant time is spent building a grandiose foundation upon which some very limited functionality is actually implemented. From the student's perspective, they feel like they must deal with the design and implementation environment for building a rocket, and in the end all that is accomplished is they light a match.

Our course, IT305, attempts to rectify this situation by having students begin with a working system, which helps us achieve our overall goal which is to increase learning. We believe that starting with a functional system allows for more complexity and functional accomplishment while minimizing "busy work" and project overhead. We are also convinced that it more realistically represents the environment our students will encounter upon graduation. In the next subsection, we describe in detail our implementation of this concept for each of our four projects.

"Starting in the Middle" Examples

Students in IT305 work in three-person teams. All teams must complete four projects, each worth ten percent of their course grade. The projects are (1) designing and implementing a web portal, (2) designing a medium sized network, (3) designing and implementing a relational database, and (4) designing and implementing a functional information system. Each of these projects is discussed in the following paragraphs.

The web portal project was originally developed as an electronic portfolio for teams to use in submitting their coursework for grading. Students were required to design and implement their portal, which included links to both their web portal design artifacts and links to their future projects (initially implemented as pages "under construction"). Unfortunately, much of the cadet work time was spent standing up basic HTML pages, which they had already learned how to do in a previous course. To combat this, we now give them a working web portal. They must keep the basic content of the given portal, but must significantly redesign the presentation. Since implementation of this new paradigm, the increase in the depth of student learning has been amazing. With the legwork of standing up the basic web pages out of the way, the students now focus their time on higher level page design concepts and create thoughtful presentations of the material using a new, consistent look and feel. Mandated use of external cascading style sheets (CSS), they are given a valid CSS to start with, also provides much deeper insights into the power and appropriate use of CSS to manage the presentation of a web site. Their learning discovery process often leads them to find both open source and proprietary material on the web. This results in important discussions about intellectual property rights and Fair Use. The simple act of providing a working web site unlocked the door to higher level learning and inspired tremendous creativity and effort.

The network design project is unlike the other three projects – this project has no implementation phase. The sheer magnitude of the scenarios involved (multiple buildings, dozens of computers, etc.) make full implementation impossible. Like the first project, this one has also evolved over time in a way that enhances learning. As originally conceived, this project required students to create a network design using components they researched on the Internet. Not surprisingly, student focus was consumed with doing web searches to find appropriate equipment and then too hastily placing it into their design, without sufficient consideration of the impact of their selection. Now, students are given access to an online "store" stocked with a diverse assortment of all the routers, switches, cables, and other equipment they could possibly need. This caused the student focus to shift from finding equipment on the web to carefully selecting from the equipment they have to choose from – which is exactly the design aspect we want them to focus on. This project involves far more subtlety then the web portal, and students now have the time and inclination to fully assess the implications of their design decisions. Once again, removal of the mindless manual tasks from the problem allows students to concentrate on the desired design elements and to achieve a deeper conceptual understanding.

The relational database project is implemented using Microsoft Access. In previous iterations of the course, students would be presented with a somewhat limited scenario for which they would design and implement a small relational database of around five to seven entities. The database would have generally simple one-to-many relationships, and ample time was allocated for the

creation of large amounts of "test data" by the students. Under our new paradigm, students are tasked to design and implement extensions to an existing, well populated, more sophisticated database. The result is that students are required to create more complex entities and relationships than previously possible, and a clear understanding of the structure of the existing database is needed to accomplish these extensions. Minimal construction of test data is required, allowing for development of more challenging queries, both against the existing database and against the student extensions. Overall, student learning has increased significantly, as evidenced by both the project submissions and by the complexity of the subsequent exam questions that are asked of the students. Test questions that in previous years would have been far too advanced are now handled well by the vast majority of students. Although not currently evaluated, we also teach students how to import large amounts of material from Excel spreadsheets. Anecdotally, students appear to feel that most of their time on this project is spent on learning and implementing concepts vice busy work, which is a marked improvement over previous years.

The final project of the course is designing and implementing an information system, consisting of web-based forms and reports that interface with a Microsoft Access database through use of Microsoft FrontPage generated ASP code. This project teaches students how to create distributed access to a database in a web environment. Previously, the course gave students the complete database to be used and students created web forms and reports from scratch, as well as PHP code that provided the interface between the web pages and the database. We continue to give students the complete database to start with, but we now have them use the FrontPage GUI interface to create the forms and reports, while also using FrontPage to generate the ASP interface code. For examinations, we even have them use FrontPage to generate a small database. As a side note, we also found that by assigning a charismatic theme for the Information System site, we increased student enthusiasm and work efforts.

Success Summary

Looking across all four projects, it has been our experience that when starting with a functional system, students are more excited about the projects, achieve greater learning at multiple levels, and are better prepared to face a world where they will build upon the work of others.

We make a serious attempt to leverage the power of enthusiasm. A student who is convinced that material is both relevant and fun will go far beyond the bounds expected for both effort and learning. Our students spend significant amounts of time completing projects not because they have to, but because they are fired up to improve, create, and innovate. We believe that removing many of the stand up costs and dealing from the start with a functional system does much to facilitate this motivation.

Starting with a functional system allows greater learning to occur at multiple and higher levels of Bloom's Taxonomy [1]. In previous iterations of the course, the projects achieved the Knowledge, Comprehension and Application levels. We feel our new projects still achieve these, but we also now achieve the levels of Analysis and Synthesis. Students must first analyze and understand the original system before they can extend it. They then modify, redesign, and integrate their ideas with the original to create a new, enhanced system.

We sincerely believe that the Analysis and Synthesis levels are exactly the skills our students will require when they graduate and begin to function in the "real world." The Information Age is making starting from scratch less and less common. Exploiting and extending current systems are the skills of the future. To build upon the work of others requires the skill of understanding the work of others. Our methodology not only builds this skill but also creates the self confidence that is so crucial to life long learning.

Challenges and Potential Pitfalls

The concept of getting students to do more by starting them with a working solution may at first seem counterintuitive. It is. Upon closer inspection, it may seem that this approach is obvious and relatively simple to implement. It is not. Some of the issues to be considered before implementing these concepts include the amount of instructor overhead involved, the need for continual assessment and refinement, and the difficulty in finding the optimal balance between design and implementation.

In order for students to learn the right lessons, triggers and hooks must be built in to the solutions they are given. Then, in conjunction with the problem scenario, the students are forced to bump into the desired learning points. Significant effort is required up front in order to make these projects a success. In addition to developing the problem, the student's starting solution must also be developed. After that, it helps to have another person independently work through the problem, trying to anticipate which direction students will go with it. This is an iterative process that requires the instructor to constantly evaluate the success of each project in order to make it even better for the next semester.

In creating the starting solution, the question that must be answered is "How much is enough?" While we find that giving a working solution has many positive results, giving too much of a working solution can have negative implications. The balance between design and implementation is delicate, and giving too much of the implementation can obviate the need for much of either, stripping the project of potential learning points. Additionally, providing too complex of a working solution can lead to the same paradox found with software reuse [2], where the time to figure out what is provided for free costs more than it is worth.

Conclusion

Our primary goal for the course is to inspire student learning. We are completely convinced that a positively motivated, enthusiastic student will learn more and voluntarily work longer and harder. We believe that having students extend functional systems has led to great increases in inspired learning which has been evidenced by the increased quality of project submissions. It is clear to us that starting projects in the "middle" will help us achieve our desired outcomes.

References

[1] Bloom, B.S. (Ed.). *Taxonomy of Educational Objectives Handbook 1: Cognitive Domain*. New York: Longmans, Green, and Company, 1956.

[2] Jensen, R.W. "An Economic Analysis of Software Reuse." *Crosstalk: The Journal of Defense Software Engineering*. December 2004.

Biographies

Edward Sobiesk has a Ph.D. in Computer and Information Sciences from the University of Minnesota. He is currently an Assistant Professor and Course Director for the course IT305-*Theory and Practice of Military IT Systems* in the Department of Electrical Engineering and Computer Science at the United States Military Academy. He can be reached at edward.sobiesk@us.army.mil.

Will Suchan has a Ph.D. in Computer Science from Arizona State University. He is currently an Assistant Professor and the IT Core Program Director in the Department of Electrical Engineering and Computer Science at the United States Military Academy. He can be reached at william.suchan@usma.edu.