04C0101

Project Number: IQP-KAL-0301 - 51

OfCourse:
ANALYSIS OF A COURSE MANAGEMENT SYSTEM FOR INDIVIDUAL INSTRUCTORS

An Interactive Qualifying Project Report

submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

by

D. Seth Hunter

Matthew K. Mursko

Steven H. Willis

Jesse D. King

Date: 26 February 2004

Professor Karen A. Lemone, Major Advisor

Abstract

The purpose of this project was to perform research and conduct testing in support of the development of a course management system named OfCourse. OfCourse was designed for individual instructors and operates on the philosophy of inserting the course management tools into the course content. Research topics included course management systems, distance learning, teaching theory, intelligent tutoring systems, and website hosting options. The field and laboratory testing that was conducted resulted in a comprehensive analysis and evaluation of OfCourse.

Contents

Executive Summary

OfCourse is a software package developed at Worcester Polytechnic Institute which provides a multitude of learning, communication, and administrative tools designed to complement an instructional website. The core philosophy behind the development of this software is the idea that tools should be placed into the content of the course, as opposed to the more common methodology where the course content is placed into the tools. The target audience consists of individual course instructors not affiliated with large universities or corporations, or those affiliated with such an organization who wish to offer a course to non-affiliated students. Instructors in this audience often have no access to existing course management system products, of which many have expensive license agreements and require more setup and maintenance than can be managed by a single person. While numerous such systems are also available under free license agreements, they also require extensive resources beyond those typically available to an average independent course instructor. OfCourse attempts to fill this void by being small, easy-to-use and instructor-oriented. The ultimate vision for OfCourse is a compact disc that can be placed into an instructor's computer and will proceed to place its tools into the course website, where ever it may be hosted, and provide the instructor with a simple and easy-to-use setup procedure.

The primary goal of our project was to perform research and testing in support of

the development of OfCourse. Specifically, our group was commissioned to perform field and laboratory tests in order to form a coherent evaluation of the software. In addition to this, we set out to extensively research what is necessary for such a course management system. OfCourse was being designed to be easily installed at some form of web hosting company. As such, we also researched companies which provide hosting for websites, in order to determine the level of service available from such a organizations, and the limitations on the developer groups.

The first step taken toward these goals was background research into teaching theory, human-computer interaction, intelligent tutoring systems, and course management systems (also called learning management systems). Our group then proceeded to survey a group of students who were likely to have used other systems such as Blackboard. Prior to actually testing the OfCourse package, we performed an online survey of various types of companies that provide website hosting services. Finally, we began performing actual field and laboratory tests of development versions of OfCourse.

Our group accomplished four successful tests of the OfCourse software package, allowing us to analyze the system and draw numerous conclusions which were utilized by the developers. We were also successful in determining a category of web hosting companies that were superior in terms of being able to host a course management system. Our background research served the important purpose of not only providing a basis for many of our evaluations, but also for providing ideas for future work that could be done in regard to OfCourse.

Unfortunately, due to time limitations, there were two goals which we were unable to accomplish. First, we had originally hoped to be able to evaluate the entire OfCourse package, including the setup scripts. However, at the time of our last laboratory test, the setup scripts and a number of key tools, such as grading

and quizzes, were either inoperable or unimplemented. Second, we had hoped to have had more time to perform in-depth research into existing course management systems, which would have given us significantly more data to compare to our results.

Working concurrently with our group were two MQP groups responsible for the actual implementation of OfCourse. These groups each developed a subset of the tools, and then combined them to form the complete system. Both groups relied on our work to make a few key design decisions, especially in relation to what web programming languages to use. They also looked to us to test the product and find major problems and inconsistencies, and to make suggestions on desirable improvements.

Acknowledgements

The authors of this project gratefully acknowledge the assistance provided from the following persons:

- Professor Karen A. Lemone, Project Advisor - for providing guidance, recommendations, and feedback over the course of the project.
- Mr. David Toth - for teaching the RoBoTL test course using OfCourse and providing an insightful evaluation of OfCourse.
- Ms. Leena Razzaq - for conducting a formal laboratory test on OfCourse.
- Mr. Jeffrey Renard - for performing an evaluation of the OfCourse courseware.
- Mr. Michael Bleyhl - for preparing a presentation on course management systems in the corporate environment, which was presented at WPI.
- Wei Peng - for allowing us to examine and attempt to install old OfCourse software.
- Meryl Gray, Nicole Hearn, Stephen Hunter, Christopher Maskwa, Michael Medeiros, Andrew Mursko, and Alan Willard - for volunteering to enroll in the RoBoTL test course, and for answering survey questions about course management systems and OfCourse.

Chapter 1

Introduction

This project deals with online, web-based, course management systems. These systems can range widely in their scope. Some are meant as just an online supplement for classes that meet regularly, while others are meant for courses in which the instructor may never meet the students face to face.

We were specifically looking towards a solution for individual instructors who would like to teach a course online, yet do not have the resources a university or corporation might have. This individual would not be able to afford the current offerings of course management software; nor would they have the servers on which to host these systems. There do exist numerous free course management systems, however these systems require resources far beyond those of a typical individual instructor. The requirements include a relatively expensive and difficult-to-manage server machine and a consistent "always on" internet connection.

1.1 Approach

There are several different approaches one could have taken with such a project. One could look at the problem from the student's side and try to figure out what is

best for the student while creating a course management system. Others may look at it through the eyes of the instructor, or from the view of the system maintainer. It would be possible to completely develop a system from one of these points of view in order to maximize the efficiency from that approach.

Ignoring any of these approaches entirely would probably not be beneficial, but for this project, the aim was to help build a system which would make the instructor's life as easy as possible. As technology advances, it can sometimes create more work for the instructor, instead of less. We wanted to ease this effect if at all possible. With less time and energy focused on the technology, instructors would be able to devote their attention towards the material being taught and the needs of the students.

Furthermore, we wanted to help build and evaluate a course management system constructed around the idea of dropping the necessary course management tools into the content of the course. Current course management systems all seem to let the tools dictate the layout and interface of the system, with the actual content of the course partitioned among these rigid tools. The goal of our project was to provide necessary background research, testing and evaluation to aid two MQP groups in their development of a course management system which satisfies the above mentioned goals: is built for an individual instructor, is instructor friendly, places the tools within the content, and can be installed somewhere that won't put an undue burden on the instructor.

1.2 Relating Science to Technology

Some people pursue science purely for the theoretical, because they have a thirst for knowledge. This pursuit can be highly satisfying on its own, but this is not

necessarily all that society can obtain from the gathering of knowledge. Hopefully, the theoretical information gained can one day be applied to societal issues and used to solve a problem.

We believe that by helping to develop quality course management systems, we will be helping to develop quality teaching environments in which knowledge can be transferred with greater ease. A well designed and implemented teaching system can help encourage better teaching methods, just as poor tools can inhibit the transfer of knowledge.

1.3 Goals

The goals of this project included producing a body of research detailing what would be necessary and feasible for a course management system as described above. There are two MQP groups who are concurrently developing a course management system called OfCourse. They will be relying on this research.

We also planned to carry out and analyze usability tests of OfCourse while it is in development. We planned to install and run a number of test courses with volunteers. This included surveys to be administered and the results analyzed. We also wanted to look into out how students currently use WPI's course management system, provided by Blackboard [1].

Our findings were presented to the groups developing OfCourse continually throughout the duration of the project. All of our results are detailed in this report and its appendices.

[1] `http://www.blackboard.com`

Chapter 2

Background

For this project, it was necessary to research topics relating to teaching, learning, and technology. We also wanted to look into scenarios where the instructor and students were not necessarily in the same location. Our background research was split into five major sections: Teaching Theory, Human-Computer Interaction, Distance Learning, Intelligent Tutoring Systems, and Course Management Systems. Each section discusses the state of the art and current challenges in the respective fields.

In "Teaching Theory" overall teaching principles are discussed; these are principles which can be applied to any course. "Human-Computer Interaction" details the science of studying how humans interact with computer interfaces, and how to create more efficient interfaces. Since a large attraction to course management systems is their ability to teach at a distance, a section on "Distance Learning" was included. "Intelligent Tutoring Systems" are software tools which can be used to assess a student's comprehension and thought process. And finally "Course Management Systems" discusses standard software packages used to post course materials, and communicate online.

2.1 Teaching Theory

Building a successful course can be thought of as the successful combination of science and creativity on the instructor's part. There are several variables which can change the effectiveness of a course. Class size, quality and availability of teaching assistants, previous student knowledge, differences in learning, and the gap between the strongest and weakest student all contribute towards the relative ease or difficulty of creating and teaching a course. These factors can lead to an environment where learning flourishes, or one where students and instructors can become very frustrated with the outcome. [12]

2.1.1 Organizational Problems and Solutions

Some of the problems that can occur in any type of class are student anonymity, frustration over unfair or inconsistent grading, loss of motivation, and organizational overhead becoming the forefront of the instructor's attention, as opposed to content. These factors can all lead to student alienation and, in certain extreme cases, class attrition. These problems occur most often in large, introductory, college lecture based courses, though any type of course can fall victim to these problems.

There are various techniques an instructor can apply to reduce the occurrence of these problems, and in some cases make having even a large class an advantage. [12] Many of these techniques are to be used in class, or are policies and procedures which can be applied when dealing with grading, teaching assistants, sections of students, and communication methods. The ideas and techniques can similarly be applied to any type of class, not just large introductory classes. It is interesting to note that distance learning courses present many of the same problems that are indicative of large courses.

Holding one-on-one meetings between the instructor and each student part way into the course can foster a closer relationship between instructors and students. Instructors can obtain a more accurate feeling for where their students are, and meetings can bring to light issues the students may not have otherwise mentioned. Experiencing out-of-class "face time" with the students in this manner is a technique which can effectively lift students from the feeling that their only roll is showing up in class and taking tests. In fact, non-classroom interaction is so useful that the University of California at Irvine sponsors a program called "take-a-professor-to-lunch". Small groups of students can invite professors to lunch on campus in order to get to know them better. The students receive a discount on their meals and the professors get a free meal. [12]

The internet can also be used in support of these techniques. A few examples include using mailing lists and discussion boards for student and teaching staff communication, and online posting of course material. In a large lecture, a student can feel intimidated asking the professor for help during class. Students stay silent in this manner and often suffer as a result. With a mailing list or discussion board, people are more likely to ask questions because it can still provide the feeling of being anonymous to those who might not ask questions during class. When there is a constructive discussion taking place, it can take down walls and bring the students closer to the instructor. These tools can foster a better sense of community and decrease student alienation.

The previous examples only demonstrate the implementation of class tools on the internet as *one part* of a solution for creating a successful course. It has recently become possible to make the online component of a course much larger. Likewise, it is possible, and even necessary, to apply the same course management principles specifically to the online component itself. As with any lecture style course, it will

not be effective merely because the course exists and someone is teaching it. In certain cases, applying these principles to the online component is a necessity, such as with distance learning courses where the instructor has no face time with the students.

2.1.2 Teaching Methodology Problems and Solutions

Instructors need to take more than just structural or administrative concerns into consideration. Understanding the way people learn is also of the utmost importance. One must realize that the base facts are not what students are being taught, but methods for understanding new concepts. It seems like most instructors know that rote memorization of facts is probably not the best method for transferring knowledge, but teaching goes beyond that. Understanding how students will absorb the new information and how it will combine with their previous understanding of the world is vital.

How People Learn details the results of an extensive research project sponsored by the National Academy of Sciences and U.S. Department of Education. The findings are very in depth and represent extremely interesting ideas, such as how to structure a course based on the structure of the information being taught.

> *Expert teachers know the structure of the knowledge in their disciplines. This knowledge provides them with cognitive roadmaps to guide the assignments they give students, the assessments they use to gauge student progress, and the questions they ask in the give-and-take of classroom life.* [4, p.230]

A large portion of their research and findings deal with the transfer of knowledge and how to deal with preconceptions. "A key aspect of the new ways of teaching science is to focus on helping students overcome deeply rooted misconceptions that

interfere with learning." [4, p.229] This idea is perfectly expressed in the following analogy by J. Minstrell in *How People Learn*:

> *Students' initial ideas ... are like strands of yarn, some unconnected, some loosely interwoven. The act of instruction can be viewed as helping the students unravel individual strands of belief, label them, and then weave them into a fabric of more complete understanding. An important point is that later understanding can be constructed, to a considerable extent, from earlier beliefs. Sometimes new strands of belief are introduced, but rarely is an earlier belief pulled out and replaced. Rather than denying the relevancy of a belief, teachers might do better by helping students differentiate their present ideas from and integrate them into conceptual beliefs more like those of scientists.* [4, p.169]

Relation to Online Courses

If one wanted to use these ideas while developing an online course, then some of these techniques could be applied in a very straight-forward manner. For example, if an instructor wants a closer community feeling, then he must make sure to have a discussion board as part of the online course. Some techniques can not be applied directly to online components. For example, large introductory courses are often split up into sections with teaching assistants responsible for a good portion of the teaching and evaluation of their sections. When moving this type of class to a web-assisted (or even web-only) course, there are several questions one needs to ask: How exactly would an instructor go about grouping people into sections who might not even be available to meet together? Would separate sections even make sense online? And, do the answers depend on exactly how much of the course is done on the web? In these cases, the way people learn online must be studied to find effective techniques for teaching a course online. The results must then be programmed into course management software while continually being monitored for feedback and new ideas.

2.2 Human-Computer Interaction

The field of human-computer interaction studies the ways in which humans interact with computers. Two of the most important topics in human-computer interaction are usability and accessibility. Usability and accessibility are especially important to consider when developing software. Usability is the degree of ease which users experience when using a piece of software. Key topics in usability are ease of navigation, availability of adequate help, and accuracy of verbiage in software. Accessibility is the degree of ease in which software can be accessed by users. Accessibility not only considers the typical user, but also considers, and attempts to optimize, the ease of access for disabled users. Being a relatively mature subject in computer science, human-computer interaction provides suggestions for improving and metrics for evaluating the usability and accessibility of a piece of software.

2.2.1 Usability

Usability is an important aspect of any new type of technology. Usability is used to measure the level of difficulty that a customer is willing to go through to learn to use new technology. When referring to online learning, Michael Feldstein points out that "If they don't find it immediately useful then they walk away." [8] This demonstrates the high priority usability testing must be given. Feldstein also mentions that when companies claim usability testing is too expensive, they tend to not understand how expensive it is to not carry out usability testing.

Usability testing has been effective in other areas of technology, such as e-Commerce. This testing can also be applied to online learning. Vendors need to be wary of new technology such as Flash because it may alienate a student instead of bringing them in. [16] They also need to consider the human-computer interaction

aspect of online learning software. Simulations or games which help students to learn material are more engaging, but creating that kind of software is very expensive and will very likely not be covered by a school's budget. [16]

2.2.2 Accessibility

Accessibility is very important to consider when producing software. The software must be available and easy-to-use for end users in order for them to take full advantage of its capabilities. Many modern software packages are integrated with the internet in some manner. For this reason, accessibility on the internet and the accessibility of software are intricately related. There are three main issues when dealing with the accessibility of the internet. First, the statistics of people who have access to the internet are important to consider. Second, accessibility for inexperienced users and users with disabilities must be analyzed. Finally, the computer technology standards enumerated in the web accessibility initiative must be adhered to when designing software.

Internet Accessibility

Use of the internet is limited to a narrow percentage of the population. The 2003 C.I.A. World Fact Book estimates the world population is 6,302,309,691 people, with a mere 604,111,719 internet users. This means that, of the total world population, only 9.5% of all people have access to the internet. In the United States, the total population is 290,342,554 people, with 165,750,000 people having access to the internet, or 57% internet user penetration. [7] Even in a technologically advanced first world country such as the United States, only approximately half of the population has access to the internet. Only a narrow portion of the population can access the internet, and therefore only this narrow portion can take full advantage of the inter-

net and the software which makes use of it. Although design and implementation of software cannot remedy this situation, it is worth noting the internet penetration of various populations when considering the audience the software will reach.

Disability and Inexperience Related Accessibility

The accessibility of the internet and course management systems for disabled users must be improved. Disabled users make up a surprisingly large portion of the online population. "At least 10% of the population in most countries has disabilities; visual, auditory, physical, speech, cognitive, and neurological disabilities can all affect access to the web." [5]. If disabled users are not taken into consideration, software which uses the internet can be a hindrance for people with both physical disabilities and learning disabilities. "Accessible online programs offer disabled persons an avenue to pursue educational options where none might have existed before. By making information more accessible to all, everyone benefits." [13]

A study conducted at the University of Toronto's Adaptive Technology Resource Centre assessed the accessibility of course management systems for blind, low vision, mobility impaired, and learning disabled users. There are many technologies available for those users with physical disabilities. Blind and low vision users rely on screen readers for interaction. Unlabeled frames, nested tables, and popup windows cause confusion to both the screen reader and the user. Some technologies which are unusable to the vision impaired are chat rooms, white boards, and other Java-based tools. Mobility impaired users were unrestricted in their use of courseware. Users with learning disabilities experienced navigational problems. They suffered from lack of consistent layouts and lack of availability of robust instructions. Users with learning disabilities also had problems related to their specific learning disability. This problem can be addressed by the course provider for the specific user. "The

major obstacles to accessibility are complexities in page layouts, inconsistencies in item labeling, a lack of instructions for task completion and the absence of consistent and clear functions related to items within courseware platforms." [13]

Inexperienced web users may also have problems accessing internet based software. They may become confused by inconsistencies in labeling and navigation. Inexperienced users may also find help documents inadequate. The writers of these documents may overlook details which need explanation to the inexperienced user. "It is also imperative to provide adequate and comprehensive instructions for the use of the actual courseware, as well as dedicated institutional support for both instructors and end users." [13] Problems relating to disabled and inexperienced users can all be effectively remedied by carefully evaluating the design of the software and writing well rounded instructions for using the software.

Web Accessibility Initiative

Adhering to computer technology standards can greatly increase the accessibility of websites and applications. The World Wide Web Consortium [1] supports a web accessibility initiative. This initiative provides an excellent definition of web accessibility.

> *Web accessibility includes web sites and applications that people with disabilities can perceive, understand, navigate, and interact with; web browsers and media players that can be used effectively by people with disabilities, and that work well with assistive technologies that some people with disabilities use to access the web; web authoring tools, and evolving web technologies that support production of accessible web content and web sites, and that can be used effectively by people with disabilities.* [5]

The initiative also provides methods to achieve accessibility. These methods include using descriptive text for links, using "alt" attributes for media tags, providing

[1] `http://www.w3c.org`

captions for media elements, and providing alternative content for applets, scripts, and other plugins the user may not have. Also provided are tips for layouts such as making page structure consistent, using cascading style sheets wherever possible, and using frames and tables with meaningful names. The initiative offers official guidelines for designing pages with high accessibility and tools to validate webpage accessibility based on these guidelines. [6]

According to the web accessibility initiative, there are many benefits to adhering to accessibility standards. Supporting disabled and non-disabled users, users with lower literacy levels, and low-bandwidth users will result in an increase in audience. Efficiency will be gained by having less of a need for maintenance and increased hits from search engines as a result of ease of parsing. [1] Addressing accessibility issues of software should be highly prioritized in the design and iterative evaluation processes.

2.3 Distance Learning

Distance learning is sometimes implemented when students cannot physically make it to the classroom environment. These students either cannot find the time, or they cannot travel to the school they wish to attend. Teachers who offer distance learning give students opportunities they may not otherwise have had. Using the internet, teachers can reach more of these students by setting up courses online. Some of the tools currently available may not be suitable for all aspects of distance learning. A good way to see if these tools are being used effectively is to carry out evaluations on the courses that are being offered.

With internet capabilities increasing, distance learning could replace the classroom. However, distance learning cannot completely replace classical teaching meth-

ods, because students desire a certain level of interaction with the instructor. Students may not feel like an active participant of the course. Some of the students who use distance learning live far away from the college or university offering the course, but some live close enough that they can take advantage of visiting the professor every once in a while. The students who do not have this advantage may use video conferencing or online chat room tools.

When considering the types of tools that are available to use in distance learning, there are two primary divisions of these tools: synchronous and asynchronous. Synchronous tools allow the student and instructor to communicate in real time; some examples are chat rooms, video conferencing, cooperative text editors and interactive white boards. Asynchronous tools are analogous to mail boxes, where you can leave a message in it until someone, either the student or the teacher, comes along and picks it up. Some examples of asynchronous tools are e-mail, discussion boards, material repositories, and homework turn-in tools. Both sets of tools have their benefits and detriments.

The advantage of synchronous tools is you do not have to wait long for responses. With asynchronous tools, if the response is not prompt enough, then an inspired student may loose interest in an important question they had asked. The benefit of asynchronous tools is students and teachers do not have to be restricted by time schedules that do not line up, making asynchronous tools more flexible than synchronous tools.

When one has set up a distance learning course, it is suggested that they should monitor how students are using the tools for that course. If students are not using the tools, then it makes it more difficult for them to learn the material. One way to see if tools are being used is to check how many times a student logs on to the course webpage, or which documents of course material they download. Another

way to test if students are applying themselves is to keep track of how often they e-mail the professor. If a student or a teacher is experiencing difficulties with the tools, it is advisable to ensure that the user interface facilitates access to the full functionalities provided by the tool.

Teachers and students can take advantage of distance learning in many different ways. They can use different tools to communicate needs and express concerns without having to be in the same location. With the ability of having everything accessible from a home computer, distance learning is making things easier for those who need it.

2.4 Intelligent Tutoring Systems

An intelligent tutoring system (ITS) is software designed to aid in the educational process. For a given topic domain, the ITS poses a problem in the given domain and then requests a solution to the problem from the student. Depending on the degree of correctness of the student response, the system will prompt the student with follow-up questions which are based on answers to previous questions. For this reason, an ITS is also referred to as an adaptive tutoring system (ATS) or a system to facilitate intelligent computer-aided instruction (ICAI). The follow-up questions attempt to model the way in which the student was thinking in order to solve the problem. The system then determines what, if anything, was erroneous in the student's solution path, and then the system will attempt to correct what was wrong. The system is programmed with common mistakes which students make with similar problems. Then, the ITS shows the student what was wrong with the solution and presents them with hints on how to avoid making the same mistake in the future. By using follow-up questions, the system will gauge the degree to which

the student has successfully learned the topic. Research into intelligent tutoring systems is attractive to those involved in both the computer science field and the educational field. A large amount of research has been done recently in this subject matter because the consequences of improvements in ITS technology will benefit the education of all.

The typical model of an ITS is made up of four parts. These are the domain model, the student model, the teaching model, and the learning environment. [9] The domain model consists of the correct knowledge about the given subject matter. This is used by the system to solve the problem. The student model consists of the knowledge programmed into the system which simulates the student thought process. The teaching model is a collection of "misknowledge", or common mistakes which are made by students on similar problems in the topic domain. The learning environment is also referred to as the didactic strategy, and is the method in which the teacher would like to educate the student. All of these domains taken together provide the basis for an ITS. Specifics about the educational benefit, implementation strategies, and evaluation criterion of an ITS will be further discussed. [21]

2.4.1 Applications

The importance of intelligent tutoring systems technology lies in its potential educational applications in the fields of academia and industry. Within both fields, this technology can be used as a traditional classroom-style instructor, as a passive assistant to aid and answer questions for students performing virtual experiments or simulations, or as the actual controlling engine for a simulation or game. The following are examples of specific implementations of intelligent tutoring systems to illustrate the variety and significance of their potential applications.

A more traditional application of this technology is demonstrated via a Java-

based dialog system used to ask and answer questions in a one-on-one environment with a student. This implementation aims to simulate a teacher-student conversation in which the teacher presents the student with material, asks them questions about the material to gauge understanding, and allows the student to also ask questions in return. [18]

This same implementation was also able to be used in a slightly different application, where it simply responded to student inquiries. The example given for this application was as a laboratory aid which answers questions from the student as they performed experiments in a chemistry lab. [18]

Another somewhat similar example with a different approach is an ITS which teaches students to argue law cases. To do this, the ITS acts as a peer to the student, and responds to the student's arguments as though it were an opposing lawyer instead of a traditional instructor. This allows the student to practice and refine their deliberation skills and tactics with a virtual "opponent". [2]

Straying even further from the traditional ITS usage, it is also possible to use such technology to control a game simulation designed to indirectly instruct a student through trial and error. The ITS technology adapts the environment of the simulation to simultaneously mimic reality and encourage the student to discover the appropriate actions. The example application of this implementation style was as a business simulation game to train students in the art of business management. [20]

Together, this diverse set of examples illustrates the broad range of important applications of intelligent tutoring systems. With proper implementation, an ITS can be used to simulate nearly any sort of adaptive learning environment, from one-on-one interactive to full-fledged environment simulations.

2.4.2 Implementation

As is true with most computer technology in the information age, the research and technology behind intelligent tutoring systems is rapidly advancing. Despite this, very few options for commercial intelligent tutoring systems exist. This fact can be attributed to two main causes. First, there is a lack of adequate artificial intelligence (AI) techniques. AI is the backbone of intelligent tutoring systems, or intelligent computer-aided instruction. "ICAI is still very much a research topic, with today's systems suffering from the same basic limitations as more general AI systems (i.e., lack of common sense reasoning). Yet ICAI is a natural test bed for AI techniques since it requires reasoning from a rich knowledge representation, modeling the user, and communication of information structures." [17] Second, the costs of developing such a system would be very expensive due to the limitations in the AI field. This capital investment, taken with the fact that these types of systems have not yet proven to be overwhelmingly effective, will prove to be a significant risk to whomever successfully undertakes the development of a commercial intelligent tutor.

Although there are few commercial options to study, many computer science and education research departments develop their own intelligent tutoring systems. One of the first intelligent tutoring systems, Scholar, developed in 1970 by Jaime Carbonell, used a mixed initiative approach to tutoring students in geography. The mixed initiative approach is similar to the Socratic teaching method. "The Greek philosopher Socrates believed that education could not be attained through passive exercises such as reading or listening, but instead came from actual problem solving. [Carbonell's] technique involved posing problems to the student, each one carefully crafted to require the student to use new knowledge, to point out gaps in the student's knowledge, or to entrap the student into discovering his own misconceptions." [17]

One of the most difficult aspects of modeling the intelligent tutor is how to represent the knowledge base. The knowledge base is how the knowledge of the system is organized, and in what method it is presented to the student. One popular method is the script, represented as a tree. The nodes of the tree symbolize events and processes, and the links between the nodes symbolize relationships between those events and processes. One other approach to representing the knowledge base is using the semantic network, which is the approach that Scholar uses. The semantic network is "a highly interconnected network of nodes and links representing objects and their properties: this requires a highly structured data base in which concepts and facts are connected along many dimensions." [17] Each element in the semantic network is an object with a name, a value, and tags. The values can be properties of the element or pointers to other elements. The tags provide semantics to the network, allowing the intelligent tutor to interpret the properties and values. These are just a few of the many approaches to modeling the knowledge base and implementing an intelligent tutoring system. Many who participate in intelligent tutor research develop their own novel ways of implementing the system, however a definitively best practice method has not been found yet. "While today's ICAI systems are carefully crafted, one of the primary motivations for further research is the potential for education systems that generate their pedagogical abilities automatically given only a base of domain knowledge. As AI and ICAI research continue to advance, computer-based tutors should become both more effective and more economical." [17]

2.4.3 Evaluation

"Intelligent tutoring systems are increasingly being employed in education and as a consequence the need for careful and systematic scrutiny of these systems has be-

come an important issue." [10] Conducting evaluations of these systems can provide incredibly useful feedback to be used in the iterative redesign process. "Most ITS researchers have concerned themselves only with envisioning the potential of ITSs and investigating the implementation issues involved in constructing actual components and systems, and have paid little attention to the process of evaluation." [10]

Although no formal evaluation methods for evaluating intelligent tutoring systems exist, the field of human-computer interaction provides us with general methods which we can use to analyze and evaluate software. There are many very general methods that can be used to evaluate all software. Some of the most notable include evaluations from expert inspections, benchmark evaluations from performance metrics, and evaluation of feedback and instruction quality. These evaluations can be applied to any piece of software, and should be applied to intelligent tutoring systems as well. There are also other general evaluation methods that more intimately lend themselves to intelligent tutoring systems. The diagnostic accuracy method of evaluation is important to use. An intelligent tutoring system must be able to recognize and interpret when a student has made an error. Diagnostic accuracy evaluations can determine when the system can and cannot successfully recognize and act on the student error. One additional evaluation that can be applied especially to intelligent tutoring systems is the proof of correctness evaluation. "Proof of correctness is a check whether the system fulfills the desired requirements or goals or whether there is a correspondence between its structure and behavior and its specifications. The method evaluates the internal components of the systems in hypothesis testing fashion and therefore is suitable for internal and experimental nature of evaluation." [10] These methods and many others can be applied to evaluate intelligent tutoring systems. The researchers and developers of intelligent tutoring systems have much to benefit from the evaluations of their systems, be-

cause, "despite the decade long existence of ITSs, the degree to which they have been successful is equivocal solely due to the lack of proper evaluation." [10]

2.5 Course Management Systems

Course management systems are a set of tools used to enhance the quality of education in distance learning courses and in formal classroom courses. These systems have the potential to reach an increasingly large audience. For this reason, computer scientists and education professionals alike have been pursing the study and development of such systems. The field of education has the opportunity to profit greatly from new technology such as course management systems. Course management systems offer many benefits to administrators, teachers, and students. Recently, The College of New Jersey researched and developed its own course management system, called Simple Online Courseware System. The research done prior to development of Simple Online Courseware System made it possible to decide which features would be included in the system. The ability to view course rosters, send course e-mail, browse course documents, browse a threaded discussion board, and submit assignments to a virtual drop box were among the features the initial release contained. [11] The ability to find a dependable source of all course related content and the facilities to communicate with others involved in the course are the primary features which course management systems offer to teachers and students to enhance the educational experience.

Considering the wealth of benefits of course management systems, it is surprising to know that not all educational institutions make use of them. There are two main reasons why course management systems are not universally adopted. First, the cost of a high quality commercial course management system is extremely high. Few low

cost options exist for the small institution or the individual instructor. Second, teaching a course with a course management system requires a large amount of work on the part of the instructor. Course management systems require at least a minimal level of computer literacy. Newer course management systems strive to minimize the effort that instructors must put forth in order to use the system to its full effectiveness. Many systems use familiar dialogs and layouts in order to aid novice computer users. Some systems reduce the time spent setting classes up through the recycling of material from previous classes. Old class material such as the course syllabus and course documents can be recalled from an archive by the system. As course management systems become easier to use and tools become more polished, course management systems will continue to successfully serve educational experiences.

2.5.1 Academic Uses

For years, schools have been putting together tools on the web for students to use in addition to their in class time with the professor. [11] It is possible to consider these tools as everything ranging from simple e-mail and posting of course documents, to taking tests online and streaming video. Several schools have gone through the process of deciding what tools they need and the best way to acquire them. Some decide that the best solution is to buy licenses to proprietary packages from companies such as Blackboard. Others decide to build in-house solutions for themselves, such as MIT's Stellar. [2] There is also the option to modify and use freely available course management systems (which were likely developed as an in-house solution at one school and then released as open source software).

However, when a school or professor implements a course management system,

[2] `http://stellar.mit.edu`

they need to make sure that it has the features they need and that it is conducive to learning. To do this, they first need to define exactly what features they need and why.

Features

There are several features which could be built into a course management system. Almost all course management systems let the teacher post course documents and important information for the students to see. Some also have the course roster available with contact information for students, and most importantly, the teacher and any assistants. Homework and project assignments can be posted along with links to helpful websites. Having the professor's schedule available for browsing on the course management system for office hours and scheduling can be very useful to students needing extra help. All of these are usually parts of a successful course management system.

While all of these tasks can be accomplished using a simple course website, a full course management system can be much more integrated with the learning process. For example, many students find it helpful to have a discussion board where they can post questions and receive answers from instructors and other students. It is this kind of interaction, which the students would probably never have without a course management system, that make these systems so unique. With the right features and the right design, a good course management system can not only aid in the learning process, but can be a driving force.

For a perfect example of a well thought out feature, there is the LabCourse Management System. LCMS has a very intriguing feature which allows for integration with LaTeX [3], a "high-quality typesetting system":

[3] `http://www.latex-project.org`

> *A substantial part of LCMS is the administration of lab course exercises. Esp. for computer science and mathematics, the use of mathematical formulas is rather important. For this reason, we have added a special support of mathematical formulas using LaTeX import. LCMS is able to convert LaTeX based exercise sheets semi automatically into HTML for publication over the WWW. The single exercises are stored within a MySQL database and can be searched by keywords, areas, and level of difficulty.* [14]

With LaTeX, one is able to format complex mathematical and chemical equations and draw diagrams where it might be impossible to otherwise. In this case, the creators of this system saw the need for a feature which would truly aid in learning, and developed an elegant tool to handle it.

Design

A course management system can have all the features in the world, but if it is not designed properly, it is not worth anything. If the course management system is not conducive to learning, students will not use it. If it is not easy to create courses, teachers will not use it.

The design needs to be carefully worked out with learning as the foremost thought. Once a preliminary system is developed, one needs to continually refine their system with usability studies, analysis, and evaluation of statistics. The development path needs to be a spiral development cycle where one designs, builds, releases, and then re-evaluates the design, and starts over again.

2.5.2 Corporate Uses

Course management systems have also proven to be quite invaluable in the corporate realm. Companies employ such software systems as a means of remotely training many employees in specific work skills or knowledge related to their tasks. This

is done mostly in large, diverse companies where the cost of bringing the intended group of employees together to the same physical location is too great. In these cases, bringing them together virtually saves time and resources, potentially expands the audience for a course at little to no extra cost, and leaves a more accessible record of course completion.

Corporate use of courseware differs from academic uses in a few key points. First, it is important to note that adults in a professional environment tend to learn differently than students in an academia environment. This difference is characterized by them tending to focus only on those bits of information that are specifically relevant to their work, whereas academic students are encouraged to learn the entire subject. Because of this, courses in professional environments should be structured as more of a reference source than an "A-Z learning program." Second, there is an increased need for software to provide tools which allow for more social interaction, since many corporate training sessions do not have regularly scheduled classes and/or laboratory sessions like academic courses do. [3]

2.5.3 Analysis and Evaluation Methods

Along with the rash of interest in and development of online course management software in recent years has come the realization that much of what has been created is of poor quality or design. Because of this, it has become crucial that individuals and organizations looking to use such software be able to evaluate various solutions before making a decision. Similarly, it has also become important for the creators of course management software to do extensive testing prior to releasing versions to the public.

These new needs have created their own fields of study: tools, models, standards and methods for testing and evaluating this online courseware. Examples would

include tools to analyze usage patterns, design models to ensure that the courseware fulfills the requirements of general course management theories, and basic standards of implementation to ensure compatibility and accessibility across a broad range of computer platforms.

Evaluation Tools

Despite the growing need for and interest in tools to evaluate courseware, there appears to be very little existing stand-alone software for this purpose. Instead, most tools of this nature are built into courseware software packages. The prime example of this, albeit little more than a data mining tool, is the feature in Blackboard that allows instructors to view usage patterns of students. This data can then be used to make rough evaluations of how useful each tool or set of tools is. It can also map common usage patterns to improve the course layout. Tools like this however, have the shortcoming of sometimes evaluating the course and its layout more than the quality of the tools of the course management system.

Logically, it follows that the most likely reason for this lack of independent evaluation tools is a lack of adherence to standards among different courseware vendors. When similar tools are implemented with entirely different structures, interfaces, and languages, it becomes nearly impossible to create a single stand-alone tool to fairly compare them.

Design Standards

When analyzing courseware systems, it is important to have some set of standards on which to base the evaluation. These standards are often written from a designer's or developer's standpoint, but ensuring conformance to them during testing is critical to many customers as it will indicate a certain level of quality of what is being

looked for (usability, accessibility, modularity, or portability).

The IMS Global Learning Consortium provides a large set of such standards. [4] These include standards for such features as vocabulary, disability accessibility, interoperability, meta-data, and content packaging. Each of these individual standards emphasizes conformance to certain ideals that are likely to be important to many customers.

The Aviation Industry CBT Committee [5] has also published a set of more specific requirements, designed to standardized software used in aviation training programs. The primary requirements of these standards include specifications of hardware and software platforms that must be supported, information interchange between systems, and user interface design.

Assessment Models

The use of assessment models is intended to help evaluators quantify various aspects of courseware equally and fairly. These may either lay out a standard evaluation process, or alternatively provide standard ways of evaluating specific features or usability aspects.

One such model, which suggests a method of courseware evaluation, was designed to eliminate stakeholder bias and focus evaluators' attention toward critical issues. This model consists of three main steps: Planning, Conducting, and Reporting of Results, each with a list of goals to be accomplished for that step. [15]

Another model, the Sharable Content Object Reference Model (SCORM), aims to define a common development model for courseware systems to be intelligent in their flow and for their modules to be easily exchangeable between SCORM-compatible systems. This model was developed as part of a project sponsored by

[4] http://www.imsglobal.org/
[5] http://www.aicc.org/

the US Department of Defense, and conformance to it should be considered a strong positive quality when evaluating a course management system. [6]

2.5.4 The Course Management System Niche

It is necessary to make a distinction between the relative roles of intelligent tutoring systems, distance learning, and course management systems. An intelligent tutoring system is a tool used to teach and reinforce a subject from a preprogrammed knowledge base. The system adapts to the individual student's knowledge of the subject and method of learning to teach effectively to that specific student. A teacher does not have to be present after the system is proven to be effective. Distance learning is the practice of education over the internet, usually asynchronously. Many teachers who teach courses on the internet use webpages as a means of displaying course content and communicating with the students. This leaves the feeling of little social interaction with the teacher and between the students. A course management system is a set of tools to facilitate the teaching of a course. Often, these tools have a classroom type metaphor to them, with tools such as announcement boards, assignment submission boxes, and grade books, for example. This metaphor effectively provides the illusion of a traditional classroom environment. Tools such as chat rooms and discussion boards provide students with more of a feeling of social interaction. An intelligent tutoring system, distance learning, and a course management system can all be used in a scenario together. An intelligent tutoring system could be one of the tools in the course management system being used to teach a distance learning course. Intelligent tutoring systems, distance learning, and course management systems are closely related entities which have their own distinct purposes.

[6] `http://www.adlnet.org/`

Chapter 3

Methodology

This chapter describes the approaches taken in order to achieve the goals of this project. One main goal of this project was to support the development of OfCourse with research. The approach we took to research background details and administer commuter surveys is discussed in this methodology. The other main goal of the project was to evaluate OfCourse. The procedures for administering and evaluating OfCourse via three test courses are described in this methodology. Also considered are administrative details, including the timeframe of the project, the materials and resources required, and the costs and budget of the project.

3.1 Research

A large portion of this project was devoted to research. We did background research on subjects related to course management systems. We read studies conducted about course management systems, and conducted a few surveys of our own. Other research we did included investigating other course management systems, different web hosts, and their restrictions.

3.1.1 Background

A large portion of research we did was background research. We studied basic theory of subjects such as teaching, human-computer interaction, distance learning, intelligent tutoring systems, and course management systems. We read journals and articles written about these subjects and summarized the main results, relating them to our topic. This information was included in our background and used to support the research and the evaluations which we conducted. We also researched both the student's and instructor's usability and needs in a course management system. We read articles in which studies were conducted on the importance of various tools in course management systems. The importance of features and tools was rated for both students and instructors. It was imperative to do background research on these subjects in order to have valuable recommendations to make to the groups which implemented OfCourse.

3.1.2 Commuter Surveys

We further studied the student's and instructor's needs by conducting surveys of our own. We surveyed students about their usage rates and experiences with tools in course management systems. We surveyed a group of approximately twelve commuter students from Worcester Polytechnic Institute about their use of myWPI, the course management system used at WPI powered by Blackboard. The results gleaned from this survey allowed us to understand the typical college student's perspective on using course management systems. We incorporated these findings into the suggestions given to the groups which implemented OfCourse.

3.1.3 Miscellaneous

Other background research we did involved investigating other course management systems that currently exist. We looked at various commercial products such as Blackboard, as well as some smaller in-house implementations. We considered which tools the various course management systems offered, as well as how widely used the systems were. This research, combined with the student and instructor research, provided us with the tools which we recommended be included in the final product.

Finally, we looked into web server hosts in order to find a place to install OfCourse. The topic of web hosts was researched because the final system needs to be installed and run on servers in order for instructors to teach with it. Besides the issues of cost and space, we were especially interested in which languages were most commonly available and the web host restrictions about installing software on servers.

3.2 Evaluation of OfCourse

The other main goal of this project was to evaluate OfCourse, the course management system that was designed and implemented. This goal was very time dependent on how fast the OfCourse implementation groups got a stable release for us to test. We received a preliminary version of OfCourse about six weeks into B term. Every few weeks thereafter, we received updates of the software. OfCourse was installed on Worcester Area Chapter of the Association for Computing Machinery's servers. In order to install the OfCourse software, there were special considerations which needed to be accounted for. These are discussed in the Administering OfCourse section below.

In order to obtain evaluations of OfCourse, we recruited volunteers to teach

two courses using the software. Jeffrey Renard volunteered to use OfCourse to supplement one of his high school classes. David Toth taught an introduction to computer programming course to volunteer students. Finally, we conducted a formal laboratory test on OfCourse with the help of Leena Razzaq. All of these tests provided the necessary feedback to obtain a comprehensive evaluation of OfCourse.

3.2.1 Administering OfCourse

In order to perform these tests, we needed to install the OfCourse system on a server somewhere. We were allowed space on Worcester Area Chapter of the Association for Computing Machinery's webserver for this purpose. They also provided database space and support. Eventually, the groups developing the OfCourse system would like to be able to make their software packages such that they could be uploaded and run from an inexpensive webhosting service. But, while still in development, it was necessary to host the project on servers for which we could easily get support.

The server we installed OfCourse on was running Debian Linux, although it should run just fine on any system that has support for Perl through CGI. We were given user accounts on the server so that any modification of OfCourse would not need the system administrator's attention. There was extensive knowledge of Linux and CGI in our group, so this situation worked out well.

We installed new versions of OfCourse while the courses were being taught, so we needed to deal with different install and upgrade procedures. After setting up a course, one of our team needed to stay on as a privileged user of the system in order to fix any problems which might have come up. Through these experiences, we became beta-testers of the system ourselves, which enabled us to directly give feedback to the developers as well as from the designated testers.

3.2.2 Surveys and Interviews

One technique which we used to evaluate OfCourse was surveying. For the RoBoTL test, we conducted surveys before, during, and after the student's experience using OfCourse. We asked them to identify which tools they deemed most important, strong and weak points of the system, and to compare and contrast OfCourse to other course management systems they had used in the past. In order to obtain the perspective of the instructor, we conducted interviews with the teachers of all our courses. The instructors informed us of which tools they considered the most useful and provided critiques of the system. Much useful information was gained from the interviewing of the instructors and the surveying of the students.

3.2.3 Audio Engineering Principles II

The first test of OfCourse was conducted to study the ease of use of OfCourse, as well as to familiarize ourselves with the software. Specifically, we wanted to study the usage characteristics of students who used the system. Jeffrey Renard, a distance learning teacher at a high school in Springfield, Vermont, volunteered for this task. Being a distance learning teacher, Mr. Renard had previously used other course management systems, the most notable of which is Blackboard. Mr. Renard was willing to teach a small class in Audio Engineering Principles with three of his students using OfCourse. Unfortunately, due to various circumstances, Mr. Renard was unable to teach his course using the software. Instead, he agreed to do an evaluation of the software. Mr. Renard and his students took time to explore OfCourse and its tools. The result of this evaluation was feedback on the features which were good, recommendations for improvements, and ideas for future work.

3.2.4 RoBoTL

The second test of OfCourse was conducted to draw a comparison between OfCourse and other course management systems from the students perspective, as well as to determine the ease of use of OfCourse from the instructors perspective. The course began in late December and continued into January. WPI graduate student David Toth was the instructor. He taught a course on beginning programming to seven student volunteers recruited by this group, consisting of family, friends, and classmates. The content of the course was programming with RoBoTL, a tool developed at WPI by a previous project group. RoBoTL is a fun and simple programming language which is similar to real programming languages. RoBoTL existed as a Java applet which was simply plugged in to the course webpage pane of OfCourse. Surveys were administered to the students over the course of their experience with OfCourse. These surveys allowed us to receive feedback from the students. Throughout the process, Mr. Toth provided recommendations for improvement of OfCourse based on his experiences with the system.

3.2.5 Formal Laboratory Testing

The purpose of this evaluation was to analyze the usefulness and usability of the OfCourse system by performing a formal laboratory-based usability study focused purely on the instructor. The subject in this study was a graduate student from the Computer Science Department of Worcester Polytechnic Institute named Leena Razzaq. Ms. Razzaq was specifically chosen because of her interest in a previous experience with teaching small distance learning courses online. Her previous experience had consisted of instructing introductory Computer Science courses using a programming language called Tea, the product of her own research projects. While

she does possess computer knowledge far exceeding that of the average independent instructor that is the primary target user of OfCourse, we felt that, because of the incomplete state of the software, this was actually beneficial to the goals of our study. Knowing the software development process intimately, she was clearly able to abstract the software package in her mind to where she saw it being as a final product. This resulted in her being able to give us meaningful and applicable feedback beyond what a more common user would likely be able to give us.

Administrative Guidelines

The guidelines used for the development of the procedure for this study came primarily from research from the Background chapter, as well as from a common Human-Computer Interaction textbook by Schneiderman.[19] Important points taken from these resources were as follows. First, it is important to carry out such studies in a neutral environment without distractions. Following this, we conducted the test in a quiet computer laboratory on campus, at a date and time of Leena's choosing, so as to lower the risks of her being overly distracted or concerned about other deadlines and tasks associated with her graduate studies. Additionally, we also did our best to keep the entire evaluation as short as possible, so as not to lose her full attention. In total, the actual process took about 33 minutes from start to finish.

Next, it is absolutely crucial to any such study that the subject fully understand that they are not the one being tested; it is the software which is being analyzed and evaluated. The goal of this requirement is to ensure that the person testing the software is comfortable and does not hold back any reactions for fear of being embarrassed or appearing unintelligent. Because of this, the first thing we did when we sat down with her and get settled in was to read her the following line of text:

> "Please keep in mind that you are not being tested here; the course management system is. While we want you to try and find the tools necessary to accomplish the tasks, if at any time you do not understand something or need help, it is important to us that you ask a question or indicate your confusion without feeling embarrassed."

One major limitation of this sort of testing is the normal inability to record the tester's thought process as they navigate through the software and attempt to complete the specified tasks. While not a complete solution, we attempted to counter this problem somewhat by requesting that Leena do her best to "think out loud" as she tried to find tools, figure out how to use them, comprehend the meaning of different labels and icons, and thought of comments or ways in which things could be improved.

A final key point which is discussed throughout most descriptions of formal laboratory testing is ensuring that the tester is fully aware of what is expected of them, how the evaluation will proceed, and what the observers will be recording. To adhere to this custom, we informed Leena that we would be running through a list of about 14 short tasks, each designed to test a functional portion of the system, followed by a short interview and chance for her to openly comment and give any other feedback she might have. We also explained to her that while she was working on the tasks, we would be attempting to record all of her comments, questions, "out loud thinking," task results, and the level of difficult she seemed to have as she searched for tools and made use of them.

Division of Labor

As part of the evaluation process, each of our project's members took on a specific role to ensure that everything ran smoothly and organized. Seth acted as the leader of the test, being the primary person who interacted with Leena by informing her

of the tasks one at a time and answering questions that she came up with. Steven played the role of a virtual student who registered for Leena's simulated course and provided some online interaction by utilizing the discussion board, chat room, and polling tool. Matthew and Jesse acted as detached observers and did their best to record all relevant information and results as they occurred.

Tasks

The list of tasks that we developed for Leena to complete was intended to simulate the normal explorative process that an instructor using the software for the first time would be likely to go through, as well as simulate some common situations that an instructor would run into during the course of using the system. Each task was designed to allow us to evaluate specific components or subcomponents of the system which were functional, as well as generally evaluate the user interface and the tool hierarchy.

The specific tools which were ready at the time of testing and were able to be evaluated were the login and registration, discussion board, chat room, polls, registration list, and registration approval and denial. Additionally, the flow of tasks was designed to lead Leena through a natural progression of discovering the tools, building on knowledge learned from previous tasks. The precise list of tasks will be discussed, along with each task's results, in the Results and Analysis chapter below.

Interview

Following completion of the list of tasks, we proceeded to conduct a short interview with Leena. For the first part of this, we asked her to simply speak freely about any comments she had, ask any additional questions, and give a general evaluation of

the system. This first part of the interview was intended to get immediate feedback about any immediate impressions she had of the system before we directed her attention to specific questions. Next, we asked her targeted questions that we had about specific tasks that she had accomplished, problems that she encountered, and her feelings on certain design decisions. Finally, we ended the evaluation by once again asking her if she had any final comments or feedback in case our questions had reminded her of anything additional.

Caveats

There are some important factors which limited the extent of this study. The study itself had to be completed before the OfCourse software package was finished and fully functional. Because of this, a number of important features of the system are noticeably absent from our tests: homework, grading, and a setup utility. This is due to those features either being unfinished or not working at the time of testing. Despite this, the tools that were tested did result in extremely useful and important feedback.

3.3 Administrative

The information below outlines the progression of this project and specifies what resources were allotted and worked with by the project team.

3.3.1 Timeframe

The work for this project was primarily split up into three sections, roughly coinciding with the three terms over which the project took place. The first third, done during A term of 2003, consisted of background research: studies of existing course-

ware implementations, evaluation methods, and related technologies. The second third, done during B term of 2003, consisted of research that was supportive of the OfCourse MQP groups, such as hosting solutions, and testing and evaluation of their software product. The final third, completed during C term of 2004, involved analyzing data gathered by tests done on the courseware and composition of this project report.

The time-line of work was as follows:

A Term:
Week 1: Initial organization and search for background information sources.
Week 2: Perform background research.
Week 3: Prepare project introduction and background information.
Week 4: Prepare project methodology.
Week 5: Survey Internet Service Providers for potential use as host of OfCourse
Week 6: Prepare project proposal and initial round of surveys.
Week 7: Survey students who may or may not have used Blackboard at WPI. Gather myWPI usage data.

B Term:
Week 1: Revised project proposal. Explored possibilities of "lurker" tools.
Week 2: Finalized research of ISPs. Located test beds for MQP product. Reviewed Tea and RoBoTL languages.
Week 3: Created surveys. Explored potential use of old OfCourse system for testing.
Week 4: Revised surveys. Prepared for guest presenter Michael Bleyhl.
Week 5: Prepared for first test of OfCourse with Jeffrey Renard. Attempted to setup old OfCourse version.
Week 6: Presentation by Michael Bleyhl. Setup new OfCourse version. Converted surveys to HTML forms.
Week 7: Gathered students and launched second OfCourse test with David Toth using RoBoTL.

C Term:
Week 1: Revised Background section of report. Prepared suggestions for MQP groups. Researched distance learning.
Week 2: Revised Methodology section of report. Researched formal methods for HCI Usability testing.
Week 3: Prepared Results & Analysis section of report.
Week 4: Prepared Conclusions section of report. Performed formal laboratory test of OfCourse

Week 5: Prepared Abstract and Introduction sections of report.
Week 6: Revised Bibliography and prepared Appendices sections of report.
Week 7: Finalized project report.

3.3.2 Materials and Resources

The significant resources which were used to complete this project were access to various course management systems, student and instructor pools from which we selected samples to survey and interview, and access to ISP accounts from which we ran and tested the MQP groups' software.

3.3.3 Costs and Budget

There was no funding or budget available to this project, and accordingly we found no significant costs involved in our work.

Chapter 4

Results & Analysis

This chapter discusses the research and studies performed as part of this project, explains the results and information gathered, and performs and in depth analysis of the data. These results and analyses are the product of the work discussed in the preceding chapter, and are the basis for the conclusions that will be drawn in the following chapter.

4.1 Hosting OfCourse

The following section contains results from the internet and Web Hosting Service Provider study that we performed. This research of various providers was done to determine what type of web space provider is best suited to hosting a course management system, and what types of resources are most commonly available via these providers. This knowledge should be extremely beneficial to any developer targeting the individual instructor, because it not only allows the developer to more effectively choose which languages to program in, but also aids them in making suggestions in their documentation as to which providers will support their product. Additionally, it should aid instructors in narrowing down their search field when

looking for some place to host their website. This is crucial because a website needs to be stored and served from somewhere, and there are an overwhelming large number of possible hosts out there.

4.1.1 Local Internet Service Providers

The first category of internet service providers that we researched were those that tended to be very localized to individual regions, as opposed to those which had a presence nearly nation-wide. The distinctive quality of these providers is that being smaller and having a smaller target consumer base, they must provide more incentives for customers to choose them over a larger and better-known company. More often than not, one of these incentives tends to be more web space, sometimes with more web programming languages available.

On average, the typical local ISP offers somewhere in the range of 10-30 MB of storage space for webpages. However, these accounts tend to be restrictive, rarely allowing custom CGI scripts and often not providing web programming languages and databases such as PHP and SQL. Additionally, the Acceptable Use Policies of nearly all local ISPs precludes the use of the web space for commercial purposes without paying an, often large, extra fee. This would prevent individual instructors from profiting from any course which they taught on that webpage, which would deter many.

Interestingly, many local internet service providers also offer separate web hosting plans, similar to those companies discussed below which offer them exclusively. These plans, however, tend to be more expensive and offer fewer benefits – a side effect, most likely, of the fact that they do not concentrate exclusively on this sort of service.

4.1.2 National Internet Service Providers

The second major category of internet service providers that we surveyed consisted of those who target a more national customer base. These providers have the same primary goal as local companies – to provide internet access to their customers – but differ in many respects because of their much larger size and focus. Because these providers' services are available nation-wide, and therefore benefit from stronger brand-recognition than local providers, they tend to offer fewer additional features to their users, especially in the area of website hosting. The national internet service providers which were surveyed were America Online, Earthlink, and NetZero. These were chosen because they represent three of the most popular nation-wide providers available today.

America Online proved to be extremely hard to gather the desired information from, as numerous contacts with different customer service representatives provided many responses indicating a clear misunderstanding of our request. After many attempts, we were able to successfully gather the information we desired. In summary, America Online does not currently offer customers the ability to run custom CGI scripts[1], nor do they offer PHP or any sort of SQL database software. When combined, these restrictions rule out the ability for one to use this provider for hosting course management systems.

Earthlink's normal internet service plan, similar to America Online's, also did not offer the capabilities needed to host a course management system on the provided web space. They do, however, have a separate service plan that is strictly web-hosting. This plan does not fall under the "national ISP" category, so it will not be discussed in depth here, but in summary it provided nearly identical service to

[1]The customer service representative did note the possibility of custom CGI scripts being allowed in the near future, stating that it was under investigation within their Research & Development division.

those surveyed and discussed below in the Web Hosting subsection.

NetZero turned out to be the hardest of all to research, as their website provided no such information, e-mail responses were strictly automatic and unhelpful, and phone support was pay-by-the-minute. However, it became quite clear based on these less-than-helpful responses that NetZero actually fell into the extreme end of national internet service providers: they do not offer web space to their customers at all.

A final important point to note about all national ISPs surveyed is that their Acceptable Use Policies all restrict the use of their webspace to personal and non-profit purposes. This would prevent an individual instructor from charging students, which would deter many potential instructors. Additionally, providers that fall into this category tended to very rarely offer any sort of "business" plan, as seen more commonly in local providers, that would allow commercial use.

4.1.3 Web Hosting Service Providers

The third major category of providers that was researched as a potential host for course management software was web hosting companies. These providers offer services which tend to be much more precisely tuned to the goal of hosting a course management system than other service providers which just include web space as an additional feature. They do not provide internet access to the customer, and instead require that the user access their accounts via the internet. Their operation is based solely on groups of machines acting as web servers which allow the customer to log in remotely and modify their content. As a trade off for not providing internet access to the customer, they tend to offer more storage space, more bandwidth allowance, more web programming languages, and cheaper, more targeted pricing plans.

The findings presented below represent a generalization of data gathered specif-

ically from ten typical web hosting companies (see Table 4.1).

• `www.lypha.com`	Lypha Networks
• `www.hostrocket.com`	HostRocket
• `www.cedant.com`	Cedant Web Hosting
• `www.hostsave.net`	HostSave
• `www.gisol.com`	Global Internet Solutions
• `www.onsmart.net`	OnSmart Network
• `www.ipowerweb.com`	iPowerWeb
• `www.jumpline.com`	JumpLine
• `www.lunarpages.com`	Lunar Pages
• `webhosting.yahoo.com`	Yahoo!

Table 4.1: Typical Web Hosting Companies

Furthermore, a quick scan of many others produced nearly identical results. Therefore, it is believed that this information represents a general overview of all service providers of this type.

Language and Tool Availability

Unlike many internet service providers, web hosting companies tend to offer a relatively common subset of popular, modern web programming languages and tools. Among these, the most common and relevant to course management software were PHP, CGI, MySQL, SSL, Java, and FrontPage extensions.

Furthermore, many providers also offered a variety of other, less common languages and tools. Because of this, it should be possible for nearly any combination of tools and languages to be used in a course management solution. However, it would be advisable for a team developing a new course management system to create a proposal of specific technologies to be used before a targeted search is done. This would help ensure that there would indeed be a large number of providers capable of hosting the final product.

Policies & Restrictions

Nearly all web hosting providers offer tiered pricing plans, which offer various levels of bandwidth and storage space. The cheapest levels are usually around 5-10 GB per month of bandwidth and 100-250MB of storage. Because of this, such restrictions should not be of great concern to the developers of this software, since the platform itself rarely incurs significant overhead compared to the multimedia used by the instructors and the number of students enrolled. Variables such as these would be under the control of the instructor, and therefore it would be their decision of how much bandwidth and storage space to purchase.

Access to accounts was most commonly restricted to FTP or other similar method of file transfer. Very few allowed the user direct SSH access to their servers. For this type of application however, FTP should be sufficient in most cases since the instructor should have little need to run programs directly on the server. The exception to this would be when some sort of setup script is provided that runs from within the server's shell instead of from the customer's computer.

Also of significant relevance to this topic is that web hosting providers are designed for commercial use, and would therefore not limit the instructor's ability to charge students for use of their online course material, should they choose to.

Finally, nearly all of these providers allow their customers to create and utilize their own CGI-bin, allowing them to run their own CGI scripts and programs. This is significant, because many local and national ISPs do not allow this, instead merely allowing access to a limited number of pre-written scripts which would almost certainly be of little use to a course management system. Since this a critical requirement for nearly all course management system implementations, it alone can often be the deciding factor when choosing a provider.

Pricing

The pricing plans for these providers all appeared to be quite reasonable for the targeted audience of individual instructors teaching to an audience of paying students over the World Wide Web. Differences in pricing plans within providers is mostly dependent upon storage space and bandwidth, while differences among providers is mostly based on quality of service and number of tools and languages provided. Basic, bottom-level plans are in the range of $8-$15 per month, with higher-level plans ranging anywhere from $25-$100 per month depending on just how much storage and bandwidth the customer needs.

4.2 Commuter Survey Results

Researching another aspect of course management systems, we administered a survey. The original survey which was administered is located in Appendix B. This survey was performed on Professor Lemone's orientation group, who were easy for us to reach. The survey contained many questions about myWPI, WPI's course management system provided by Blackboard. Unfortunately, most of the students were freshmen and had not yet been in a class which used myWPI.

From the twenty commuter students, a dozen showed up to one of the social activities. At this activity, we passed out a survey that was created to assist us in understanding what students liked about myWPI, and generated the results seen in Appendix B.2. The surveys were rather informal, and some questions left room for interpretation, which we tried to clarify while we waited for them to fill out the surveys.

Most students were discovered to not have had used myWPI for their classes. The two or three who said that their teachers used myWPI, they rated the effectiveness

of the use by the teacher at approximately 7.8 out of 10. Out of those students, they rated the impact of myWPI on their ability to learn low at 3.7. A few students said that it would have been better if more teachers used myWPI. The students who responded rated the ease of use of myWPI to be about 8.1. Other responses from the students stated that the discussion board is the most important feature to them. One student mentioned that he liked the *Grades* section of the system. A few students who had used other course management software packages also rated them well. There were a few suggestions on how the usefulness of myWPI could be improved, such as use of virtual classrooms and having more online help. General comments suggested that the overall usability and usefulness of myWPI was fairly good.

4.2.1 Analysis of Surveys

Many students do not use the course management system for their courses. Since the students who took our survey, very few said that their teachers used myWPI; most answered with an N/A or blank. If the professors were to utilize myWPI more, then the students would be inclined to make more use of myWPI. For example, if the teacher only uploads assignments and does not make topics in the discussion board, then students would not use the discussion board. This indicates that myWPI is generally not being used very much or very effectively at this school with the general undergraduate student body. The more myWPI is used, the more potential educational benefit it can show.

4.2.2 Student Opinions

Some comments were revealing as to what the students liked and disliked about myWPI. One student said, "I like how [myWPI] is organized, but I do not like the

design." The structure has an organized layout, but navigating that structure seems difficult. Some tools that the students want to find are buried under hierarchical categories, which are not obvious to some users.

The myWPI system has good communication tools that help make students feel like they are closer together. For example, students of an orientation group can discuss with each other times for getting together for social events. A helpful communication tool mentioned in the surveys was the *Groups* section. This section has information on students who have been established as a group. The *Groups* section is a webpage that can be visited by members of the group, who are placed together by an advisor or instructor. Here, group members can post shared files, e-mail other group members, and even have a group discussion board. Many of the students surveyed said that they found the discussion boards to be helpful. Since students do not see each other in the classroom, they can communicate using these helpful tools.

The communication tools also can be integrated with a class. The discussion board can be used to help students with problems they have while outside of the classroom. The teacher, TA, or other students can help a student that is having trouble with a problem from class. One student wrote that he really liked the assignment and grades features of myWPI, which are also helpful tools for a class. These features allowed him to easily get the homework needed and to keep track of his progress through his course.

4.2.3 Error of Surveys

Unfortunately, the survey sample was very small (twelve people). Statistically speaking, data analysis should be completed on larger samples (twenty or more). Too few people in a sample give a poor representation of the target group as a whole; it

creates a high chance that some people would not be represented.

Another concern with the validity of the survey was that the students were not chosen at random. If students are not randomly selected from the target group, the result is a poor representation of that group. When students are selected from a subgroup, they may create bias because they belong to a uniform group. For example, the group we selected was composed of first year commuter students, and thus was not a random sample of the target group, WPI undergraduate students. This survey would have been more accurate if we used random students from different classes and varied backgrounds.

Some of the questions from the survey were just variations of previous questions. Also, some questions were poorly worded, which could be seen when students asked about some of the questions. The surveys could have been simplified if some of these ambiguities and redundancies were removed. Questions with minor differences do not necessarily obtain more information, and questions that are not clear do not help the accuracy of results.

4.3 OfCourse Evaluation

The testing and analysis of OfCourse was at the very center of the goals for our project. OfCourse was developed with the primary goal of providing an easy, cost-effective solution for individual instructors who are unaffiliated with a large university or corporation, and therefore unable to afford or manage many of the large course management systems which are currently available. Each of the three tests below were conducted independent of one another. The first test was with a high school teacher and the students in his class, *Audio Engineering Principles II*. The next test was performed with a WPI graduate student teaching a introductory pro-

gramming language called *RoBoTL* to a group of volunteers. Finally we performed a formal laboratory test of the software package with another WPI graduate student. Also included in our evaluation are the views we gained of OfCourse through our own continual and extensive use of the course management system.

4.3.1 Audio Engineering Principles II

We gained a good deal of information from our first test of OfCourse with high school students. First of all, it let us become more acquainted with OfCourse ourselves. We were able to learn about OfCourse's internals as well as its user interface.

Unfortunately, there were a few problems with getting this course off the ground. To begin with, we had originally planned on using an older version of OfCourse with these students so we could compare their experiences with those of students using the newly developed version. After some initial work in this direction, we eventually found out from the old system's maintainer that the old version of OfCourse had gained some hard-coded "hooks" into its currently installed location that would make it all but impossible to install anywhere else.

It was decided that we would see what the other project teams had developed so as for the new version of OfCourse. Fortunately, they had something we could start using almost immediately. Even so, we had already lost some time due to the false start, and it would take a little bit of time to work out the last few major bugs in the new OfCourse. We eventually got a working version of OfCourse up and running (see Figure 4.1)

There were still further problems relating to the different position of our winter break from that of the high school instructor and students we wanted to use the system. Eventually, these logistical problems worked themselves out and we were able to get some feedback from the instructor.

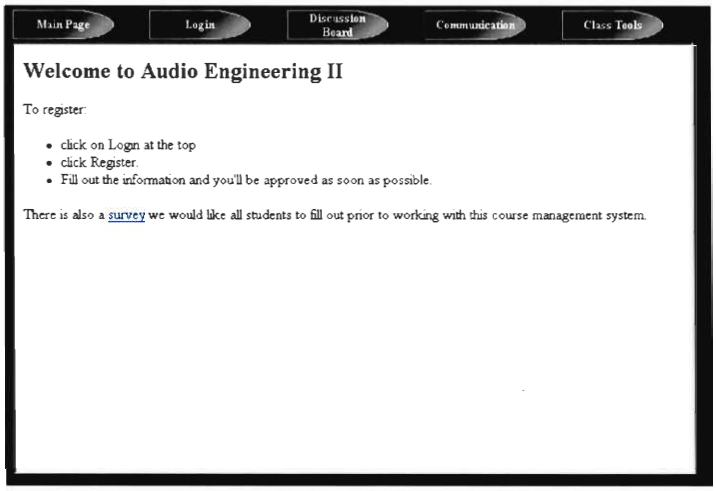


Figure 4.1: The main page for Audio Engineering Principles II in OfCourse.

Feedback

The instructor, Jeffrey Renard, noted that the discussion board was the strong part of the system. He commented that it would be helpful to add a file attachment to messages posted to the board. Related to this idea is that of a file upload tool. OfCourse does have such a tool, but this was part of the system we were unable to test at this point.

The other discussion oriented tool he commented on was the chat room. Unfortunately, Mr. Renard experienced significantly high latency, and therefore found it difficult to participate in real-time synchronous discussions. However, he did find that the logging capability of the chat room was a very nice touch. This would enable any student to go back through discussions for help on a specific topic. An additional feature he thought would be helpful in a system like OfCourse is an interactive white board utility integrated with the chat room.

Mr. Renard thought the gradebook was a nice item, but felt that it needed more work to truly be useful. The ability to to sort the items by various criteria, such as category would be helpful. There is also currently no way to perform a sum or

average on the grades. There is also no method available to weight different assignments. Basically, he felt that the current system was relatively good at recording grades, but did little else; more functionality is needed.

Analysis

Mr. Renard made some very good points about the current interface and functionality of the system. In his view, the grade book needs to be developed further; an instructor should not need to do the extra work of figuring out the grades when the system was built to make such administrative tasks easier.

The chat room also needs a great deal of work. While the only problem found with the chat room was the lag time, it is such an important problem that it needs a great deal of attention. Based on our research, holding sessions with students in a chat room type environment is almost a necessity for a completely online course where the students and instructor have no other real-time interaction. If there was a discussion going on with several people in the current chat room with its current lag time, our tests and research indicate that there would probably be quite a bit of confusion and frustration amongst the users. If the lag time becomes too much of a burden, then the chat room's usefulness will become a detriment and symptoms such as student alienation as described in Section 2.1.1.

The other reference to the chat room was for an additional feature, the white board. This would enable instructors and students to more easily and naturally express ideas and concepts that might not readily translate into plain text. Many people can understand a problem, theory, or idea better when they can visualize it. With the white board addition to the chat room, a professor could be discussing an idea with students via text discussion while at the same time drawing a diagram further illustrating the idea.

As for the proposed file attachment addition to the discussion board, we can see potential in this idea. Although we were unable to test the file upload tool, which is a feature in OfCourse, it would be interesting to somehow combine these two ideas such that every file attached to a discussion board message automatically goes into the file upload area.

Mr. Renard also seemed a bit confused by the overall structure of OfCourse. The current version of OfCourse was built so that one installation would support a single course at a time. It is logical to assume that an instructor may have more than one course he would want to teach. As such, in his review Mr. Renard made a couple references to the problems which would come about when there are several users with several available courses. These issues were possibly not dealt with by the development teams since only one course would be dealt with for any installation.

Mr. Renard noted that the student registration process was decentralized, and in some cases could be good for an instructor who wants tight control over their classes. Each course has its own area to approve student registrations for enrollment. This fits with the overall theory for OfCourse, that the tools are placed inside the course. In this manner, the registration tools are placed in each course individually. However, he did note that with many students and courses, a centralized system would be more useful.

4.3.2 RoBoTL

The second course which was used to test OfCourse was taught by WPI graduate student David Toth, and consisted of seven volunteer students, recruited by the members of this project group. These volunteer students were family, friends, and classmates of the members of this project group. The content of the course was beginning programming with RoBoTL, a tool developed at WPI by a previous project

group. RoBoTL is a fun and simple programming language which is similar to real programming languages. RoBoTL is intended for those who have had no programming experience and for those who want to learn the basics of object-oriented programming. Examples of code and tutorials are provided to aid the student in the learning process. OfCourse was installed on the ACM's servers, then Mr. Toth created and posted homework assignments to be completed by the students. As the teacher, he moderated the course and was available to provide help to those students who needed it. The "content" for the course, the RoBoTL Java applet, was not located on the ACM's servers, but on another. Due to OfCourse's design, we were able to create a simple course webpage (see Figure 4.2) that just included a link to RoBoTL. This worked seamlessly within OfCourse's framework.

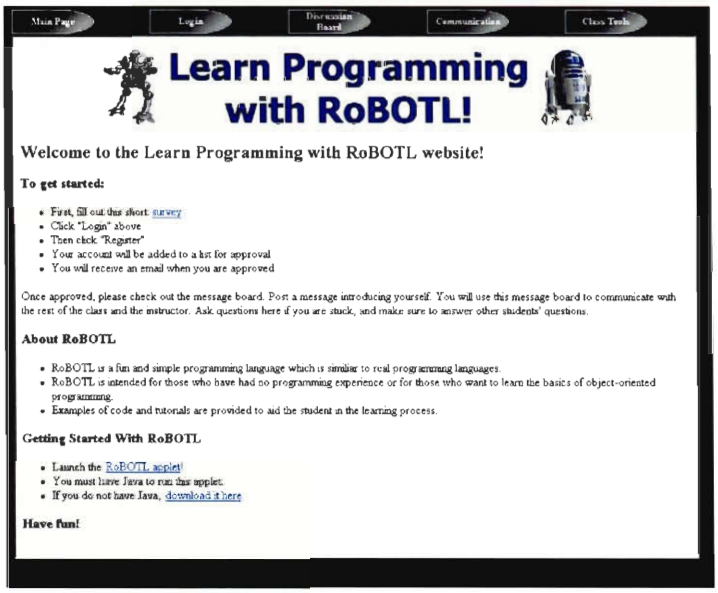


Figure 4.2: The main page for RoBoTL in OfCourse.

Surveys

Through this process, surveys were administered to the students before, during, and after their experience with the courseware. The actual surveys administered are located in Appendix C. We received five responses for the before-use survey, one response for the during-use survey, and no responses for the after-use survey. The results of these surveys are also located in Appendix C.

The before-use survey was designed to gauge the students previous experience with any other course management systems and their satisfaction with those systems. Three of five respondents stated they had previously used Blackboard. Those three respondents agreed or strongly agreed to the statements that their experiences with Blackboard were satisfactory; and that Blackboard aided with the learning of the material in the course. We also attempted to determine which tools that the students considered to be most important to include in a course management system, on a scale from 1 (least important) to 5 (most important). The lowest four tools, the discussion board, real-time chat room, file turn-in utility, and the online quiz tool had an importance of 80%. The course document repository had an importance of 84%, the announcement section had an importance of 92%, and having access to student grades had an importance of 100%. These before-use of OfCourse survey results show that the content aspect of a course management system is more important then the communication aspect. The majority of the students who volunteered to take this course had never taken a distance learning course. Their only experience with course management software was in conjunction with a course that met regularly. We suspect that after a period of time with no in-person contact with the teacher, the students will value communication tools higher.

The during-use survey was designed to gauge the effectiveness and usefulness of specific tools implemented in OfCourse. The survey we administered was for the

evaluation of the discussion board. We asked how many times a week the student used the discussion board, if it was helpful to learning the course material, and if it performed up to expectations. The survey also asked for any suggestions to improve upon the discussion board tool. One student responded to this survey, and used the discussion board zero times a week, and likewise felt neutral to its degree of helpfulness and performance. Nothing can be learned from this response because the student did not use the discussion board tool.

The after-use survey was designed to gauge the students overall impressions and satisfaction of OfCourse. It also asked for any feedback on ways to improve OfCourse, and asked to rate the individual tools. These after-use ratings for the tools can be compared to the ratings received on the surveys before the use of the system, and some interesting conclusions could be drawn from this comparison. Unfortunately, after posting reminders on the discussion board and sending an e-mail to the class list, no students responded to this survey, so nothing can be learned from the students after their use of OfCourse.

Feedback

In addition to providing us a forum to survey students about their use with OfCourse, Mr. Toth provided us some valuable feedback about the system. He is a graduate student in computer science at WPI, has taken a class in human-computer interaction and has done usability studies on other projects. Mr. Toth has the experience to make useful and insightful comments on ways to improve OfCourse. Mr. Toth had constructive comments about the navigation of OfCourse, the discussion board tool, and some general comments.

The ease of navigation of a system is important to the system's success. If the navigation of a course management system is confusing or hard, students and teach-

ers will not feel the use of such a system is a worthwhile expenditure of time. Mr. Toth felt the navigation of OfCourse was confusing. In OfCourse, there are multiple ways to get to different tools, and it can take a long time to get where you want to go. The discussion board is a main section, yet it is also under *Communication* Similarly, the grade book is under *Class Tools* and *Instructor Tools.* This leads to a feeling of frustration with the system. Menu layout is an important aspect of navigation. Mr. Toth felt that having a vertical menu would be more effective then the current horizontal menu. He suggested laying the items out in a hierarchical tree form, with categories such as communication, tools, instructor tools, and account. The communication menu would have items such as the discussion board, chat room, and class e-mail list. Tools would contain items such as homework drop box, check grade, and file upload. Instructor tools would have the ability to add and grade student assignments. Under the account tab, the user would be able to register, login, and logout. Mr. Toth suggested that having the ability to logout under the *Login* tool was misleading, and that it would be ideal to have *Logout* be hidden before you login, and likewise have *Login* become hidden after you login. Mr. Toth did not like the use of frames in the system. Different browsers can display and navigate through frames with unexpected results, and furthermore, it is difficult to link directly when using frames. As an alternative, Mr. Toth suggested using server side includes. Server side includes provide a similar functionality as frames, but all content is displayed in one single webpage pane.

The discussion board was the most extensively tested tool in this test of OfCourse. Mr. Toth thought it was very good in general, but had a few suggestions for changes. The discussion board features the ability to post anonymously. The notice that a message was posted anonymously appeared above the post on some occasions, and below it on other occasions. The notice should appear in the same lo-

cation consistently. Mr. Toth also felt the navigation through the discussion board was somewhat confusing (see Figure 4.3). He suggested having a list above and below the content of the post, with the format *Discussion Board - Topic Name - Thread Name*. *Discussion Board* would be a link to the main discussion board page, *Topic Name* would be a link to the current topic, with *Topic Name* replaced with the actual name of the topic, and *Thread Name* would be replaced by the actual name of the thread.

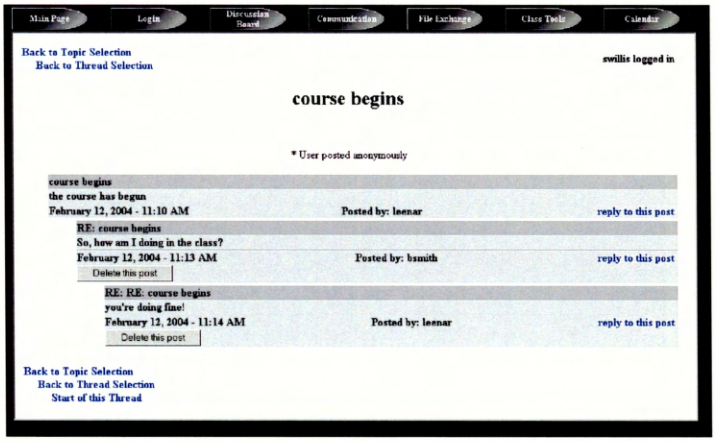


Figure 4.3: A sample thread on the Discussion Board in OfCourse.

Mr. Toth had some general overview comments and suggestions about OfCourse. He suggested having the system automatically e-mail a user once they are registered for the course; it is too much of a burden to put on an instructor to send e-mails individually. Mr. Toth felt that grading was somewhat confusing. When attempting to change a grade, he could not locate the button that would do this. Once Mr. Toth was shown the button, he did not remember it being there. The grading system should be made more user friendly. Mr. Toth felt the chat room was awkward to use because it constantly refreshed. He suggested it would be highly beneficial to redesign the chat room tool to avoid the refreshing.

4.3.3 Formal Laboratory Testing

The formal laboratory test targeted one of the largest secondary goals of the development of OfCourse, an easy-to-use and useful setup utility for getting a course up and running on a web host. This test was conducted by assigning specific tasks to a graduate student, Leena Razzaq, with regard to setting up a course to be taught using the system. Her progress was then monitored, recorded, and analyzed with specific interest paid to usability and the original goals of the setup scripts. Following the test, we also interviewed Ms. Razzaq for some extensive feedback based on her experience.

Unfortunately, at the time of this usability study, there was no functional setup script for OfCourse available to be tested. Prior to the test, we installed the latest version of OfCourse on one of the ACM's servers and created a very rudimentary course content webpage. This would be the platform on which Ms. Razzaq would complete tasks for the usability study. The script of tasks which we developed was carefully worded to avoid any interface specific terms where possible. The main idea of the task was stated, the method by which to accomplish the task was left to Ms. Razzaq. There were approximately fourteen tasks, each of which will be discussed. Following the task descriptions, the outcomes of Ms. Razzaq's attempts to complete the tasks will be discussed.

Scripted Tasks

First, Ms. Razzaq was given the URL to the test course webpage and was asked to register herself. She remarked how the register button was hard to find and that it should be placed more prominently. The information requested on the registration form was clear and well laid out. Acting as administrators, we approved her as a student and elevated her to instructor status. Ms. Razzaq suggested a clearer

method, or possibly separate methods for registering students and instructors. She was unsure whether she was registering as a student or instructor. She also was concerned that a user with malicious intent may attempt to register as an instructor, possibly leading to security breaches. After approving her as an instructor, we asked Ms. Razzaq to login to the course management system, which she successfully did. She noted that logging in was easy and clear; the *Login* tool is a main menu button.

The following set of tasks involved testing the discussion board. First, we asked Ms. Razzaq to create a discussion board topic to welcome the class (see Figure 4.4). The discussion board was simple for her to find as main menu button, and she successfully created the topic. She noticed it may be helpful to place the *Create Topic* button at the top of the discussion board. After many topics have been created, the button may get lost at the bottom. Next, we asked Ms. Razzaq to create a thread to tell the students that the course had begun. She seemed to be confused on how to enter the discussion board topic she had previously created. She needed to be told to click the topic hyperlink text in order to enter that discussion board topic. A small image or more prominent text may remedy this confusion. After entering the topic, Ms. Razzaq was able to create a new thread easily.

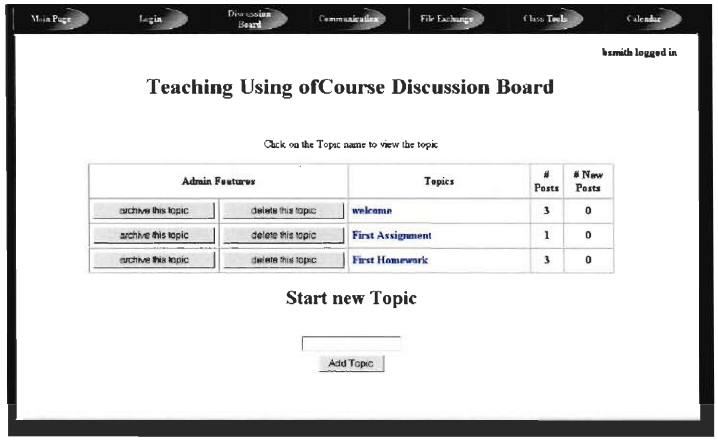


Figure 4.4: The main page of the Discussion Board in OfCourse.

At this point, we created and registered a test student which we would use to interact with Ms. Razzaq. One of the members of our group, Steven, acted as this test student. The next task to be completed was to approve this test student. She was able to find the tool and successfully approve the test student on the first attempt, despite the fact that the tool is deep in the menu hierarchy. The user approval tool is located under *Class Tools - Instructor Tools - Approve Users*, which is a very intuitive organization.

The next task which Ms. Razzaq was asked to accomplish was to create a poll. She found the poll creation tool easily, located under the hierarchy *Class Tools - Polling - Create New Poll.* In the process of creating the poll, Ms. Razzaq was somewhat confused by the meaning of the *Topic* and *Number of Choice* fields. Furthermore, when specifying the choices for the poll, she was confused by the *Question Number* fields. Ms. Razzaq felt that the wording of labels was very misleading. *Topic* should be *Question*, and *Questions* should be *Choices* or *Options*. After filling in answers for the poll, Ms. Razzaq thought she had completed the task. She needed to be prompted to look for an addition step to finish creating the poll, at which time she noticed she needed to change the status of the poll to *Active* (see Figure 4.5). After creating a poll, the need for activation should be more prominently displayed, or active could possibly be the default status for new polls.

While Ms. Razzaq was creating the poll, our test student made a post to the discussion board under the thread which was created earlier. The next task for Ms. Razzaq was to respond to this post. After slight delays while searching the interface for the appropriate buttons, she was able to respond to the test students post. In order to respond to a post, the user must go to the discussion board, click the topic of where the post is, click and read the thread which the test student created, and then click the reply button to be prompted with the field to type in a response. This

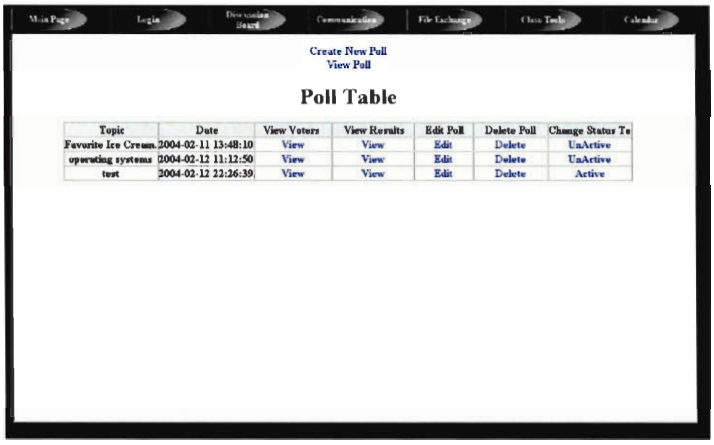


Figure 4.5: The main menu of the *Polling* tool in OfCourse.

straight forward task was accomplished successfully on the first attempt.

During the time which Ms. Razzaq was replying to the test student's post, the test student answered the poll which was previously created. We asked Ms. Razzaq to view the results of the poll. The hierarchy for this task was intuitive. To view the poll, Ms. Razzaq looked in *Class Tools - Polling - View.*

The next set of tasks involved interaction in the chat room. We asked Ms. Razzaq to enter the chat room. She found the chat room easily, located under the *Communication* main menu button. She was a bit confused by the prompting for color selection, and wondered what would happen if she did not select any colors. When Ms. Razzaq entered, our test student was there waiting, and they carried on a short test conversation (see Figure 4.6). Ms. Razzaq noticed the slowness of the chat system, how it needed to refresh every few seconds, but only after being asked about it. She found this to be mildly annoying and recommended implementing an improvement to eliminate the refreshing. The next task was to remove the test student from the chat room. She quickly found the *Boot* button under the test students name on the side of the chat room. The first time she clicked the button, the test student was not removed from the chat room. We speculated that the

page was in the middle of a refresh and did not register the button click, which is another reason to get rid of the constantly refreshing interface. The second attempt at removing the user was successful.

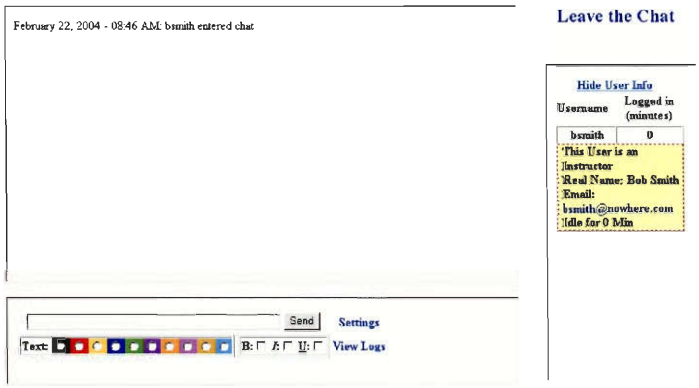


Figure 4.6: A sample chat room session in OfCourse.

After removing the test student from the chat room, we asked Ms. Razzaq to view the logs for that chat session. Initially, she was unsure where to find this feature. Ms. Razzaq browsed in the *Class Tools* and the *Instructor Tools* sections, and then asked where to find it. For her, it was unexpected to find logs in the chat room window. Once Ms. Razzaq was told it was in the chat room window, she found the button easily. After viewing the log, Ms. Razzaq was asked to hide the log from the students. She was confused by the *Archive Log* button and its functionality. After asking, Ms. Razzaq was told that it would hide the log from the students, and she was able to complete the task. This was an additional case of buttons that are mislabeled and need to be renamed to better convey their functionality.

When Ms. Razzaq had completed hiding the chat log from the students, we asked her to view the list of registered users. The tool is located under *Communication - Class Listing*, and she was able to find it very easily. Ms. Razzaq remarked how she liked the layout and that it was easy to understand.

While Ms. Razzaq was browsing the class list, our test student simulated posting an inappropriate message anonymously on the discussion board. We asked Ms. Razzaq to identify the poster, delete the post, and un-register the offending student. The post was located easily by Ms. Razzaq, being in the same thread as the previous posts in this usability test. Initially, it was not clear to her that the post was made anonymously. After being shown the asterisk that denoted the post being anonymous, Ms. Razzaq recommended it be shown more plainly that a post has been made anonymously. She was easily able to delete the post from the discussion board. When searching for the tool to un-register the offending student, Ms. Razzaq seemed unsure where to find it. Eventually, she found it in menu hierarchy *Class Tools - Instructor Tools - Approve Users.* Once again, the semantics of the system confused Ms. Razzaq. She was able to interpret the *Disapprove* button as the tool which would un-register a student, but recommended it have a clearer label such as *Un-Register.* Furthermore, Ms. Razzaq suggested the tool for this be moved to the class list tool, or that perhaps the *Approve Users* tool be renamed to *User Management.* Despite these remarks, she was able to successfully un-register the test student.

The final task Ms. Razzaq was asked to complete was to logout of the course management system. She was able to find the *Logout* button under the *Login* main menu button, but recommended the logout option be displayed on every page.

Feedback

After completing the scripted tasks, we asked Ms. Razzaq to comment on her specific likes and dislikes of the system. She expressed that she liked the system, and that it was fairly easy for her to use. Ms. Razzaq also said it seemed to have all the necessary tools for a good course management system. She also liked the fact that

the layout was not cluttered. Ms. Razzaq also expressed a few dislikes of the system. Her main dislike of the system was the fact that many of the labels were confusing or misleading. She also commented that some of the tool hierarchy was confusing, and recommended it be reorganized. Ms. Razzaq specifically said that the *Approve User* tool was excessively busy, resulting in it being hard to find the options she needed. Finally, she commented that the interface was somewhat plain for her tastes, although this may have been a consequence of the simple template webpage which was created prior to the usability test to be the course content. Despite these dislikes, Ms. Razzaq stressed that she liked the OfCourse course management system very much.

Analysis

Ms. Razzaq believed that OfCourse had all of the tools necessary to be a good course management system. Furthermore, she found it fairly easy to use. On her first use of the system, Ms. Razzaq was able to accomplish the scripted tasks with little to no help. After being explained the theory behind OfCourse, she agreed that the system did indeed satisfy the melding of the management tools into the course content.

However, there are several areas Ms. Razzaq noted which needed improvement. These improvements fall into three main categories. First, there are improvements which need to be made to the chat room interface. Second, there are many wording and semantic errors which need to be corrected. Third, it is important that the menu hierarchy and layout be reorganized in order to be more effective.

During a test chat session, Ms. Razzaq found the constant refreshing to be an annoyance, and recommended that the chat room be upgraded in such a way that it does not require refreshing. We also speculate that the refreshing was the cause of the failure to boot a user on one occasion. Ms. Razzaq claimed to have clicked the

button, but the user was not removed from the chat room. The user was successfully removed from the chat room on the second attempt. We suspect that the chat room was in the middle of a refresh, which caused the system to not recognize the button click. For these reasons, we conclude that the chat interface must be changed not to constantly refresh.

Much of the confusion which Ms. Razzaq encountered during the usability study was a result of improperly labeled buttons and items. On many occasions, she asked for clarification on the meaning of buttons in order to complete the scripted tasks. The most notable mislabeled items were the *Archive Log* button, the *Disapprove User* button, and the labeling of *Polling* options. *Hide Log* more accurately portrays the functionality of the *Archive Log* button. Likewise, *Un-Register* user is more explanatory than *Disapprove* user for cases where the user has already be approved. When creating a poll, the *Topic* field should be replaced with *Poll Question*, and the *Questions* fields should be replaced with *Poll Choices* or *Poll Answers*. It would be very beneficial if the labels of the interface were renamed to be clear and consistent.

The last category of items which caused Ms. Razzaq problems in the usability study was the hierarchy of the menu and layout of items. First, tools such as the discussion board are located in multiple places. Each of the tools should be in one single spot which is easy to find. The menus which contain these tools must be properly labeled in order to convey their contents. Second, some tools are not located where they were expected to be. An example of this is the logout tool being under the *Login* menu. A reorganization of the menu hierarchy could remedy this. Finally, not all of the tools were immediately visible. Many of the tools are hidden in the menu hierarchy. Ms. Razzaq often had to look in multiple menus in order to find the tool she was looking for. More than only the top level of the menu hierarchy should be displayed to the user. With more tools readily visible, fewer errors will

occur and less time will be spent looking for the appropriate tools. If the menu items are reorganized and all displayed, the system will be much more effective and intuitive.

4.3.4 Our Use of OfCourse

Unfortunately, none of our volunteers were able to test out the installation procedure required for installing OfCourse. This was due to several factors. First of all, OfCourse was supposed to be able to be installed on a web-hosting provider, yet this ability was not functioning by the time we needed it. Second, if not installing at a web host, the user needs shell access for an account on the server being installed on. Third, OfCourse and its install script still had many bugs in it, which needed close attention. Fourth, a new database needed to be created for each new course, this was something the install script could not do.

Basically, at this point in development, the install and maintenance required a very solid understanding of Linux/Unix operating systems, MySQL, Perl, PHP, and CGI. This required our group to do all of the installation and maintenance for every course we tested. This gave us a very good look at the structure of OfCourse, and enabled us to make some observations and conclusions from the installer's point of view.

One Course Per Installation

One of the most interesting aspects of OfCourse was the fact that each course required a separate installation of all the tools, and required a separate database. This seemed to follow from the logic that the tools should be placed inside the content; all the CGI scripts exist in every directory that holds a course. This makes upgrading to a new version of OfCourse very difficult since the instructor needs

to upgrade each course separately. It also leads to waste since all the scripts are duplicated across all the courses. This seems to be an unfortunate decision from purely a software engineering, and program maintainer view point.

If you are teaching two courses and you have students who may be in both, all of the students' data will have to be entered and stored twice. Likewise, the instructor will need to register himself for every course. The installation procedure will need to be followed every time a new course is added. This will lead to situations where two different students taking different courses could have the same user name since their information is stored in separate databases. This could lead to a great amount of confusion for the instructor and students.

From the point of view of someone who needs to install and maintain OfCourse, we believe that OfCourse needs to support more than one course per installation. Further, all the course and student information should be stored in one database. We believe this will make life easier for everyone involved: the maintainer, the instructor, and the students.

Installation Procedure

It is the eventual goal of the OfCourse programmers to create a CD which would be used by their target audience to automatically upload and install OfCourse on a web-host. We were unfortunately unable to test this with any volunteers, or even by ourselves.

However, we were able to test the web-based install procedure as noted earlier. And this part of the install will still exist in the final version. The idea of a web-based install procedure which asks the user for input in order to customize the installation was a very good idea. It is all too common for the installation procedures, and methods for customizing software to be very obtuse. Configuration files, compile

options, configure options, and many other difficult to understand configuration options can cause great headaches for a user. Using webpages to ask for all the input that is needed by the user helps wonderfully. Unfortunately, due to current bugs and design problems, the install still needs to be carefully performed, and in some cases corrected.

Chapter 5

Conclusion

Presented below is a summary of our project, including its goals and results. Following this are two sections which discuss the conclusions drawn for our two primary goals. Finally, we present a discussion of what our group suggests and foresees as future work on OfCourse by both IQP and MQP groups.

5.1 Summary

OfCourse is a software package developed at Worcester Polytechnic Institute which provides a multitude of learning, communication, and administrative tools designed to complement an instructional website. The core philosophy behind the development of this software is the idea that tools should be placed into the content of the course, as opposed to the more common methodology where the course content is placed into the tools. The target audience consists of individual course instructors not affiliated with large universities or corporations, or those affiliated with such an organization who wish to offer a course to non-affiliated students. Instructors in this audience often have no access to existing course management system products, of which many have expensive license agreements and require more setup and mainte-

nance than can be managed by a single person. While numerous such systems are also available under free license agreements, they also require extensive resources beyond those typically available to an average independent course instructor. OfCourse attempts to fill this void by being small, easy-to-use and instructor-oriented. The ultimate vision for OfCourse is a compact disc that can be placed into an instructor's computer and will proceed to place its tools into the course website, where ever it may be hosted, and provide the instructor with a simple and easy-to-use setup procedure.

The primary goal of our project was to perform research and testing in support of the development of OfCourse. Specifically, our group was commissioned to perform field and laboratory tests in order to form a coherent evaluation of the software. In addition to this, we set out to extensively research what is necessary for such a course management system. OfCourse was being designed to be easily installed at some form of web hosting company. As such, we also researched companies which provide hosting for websites, in order to determine the level of service available from such a organizations, and the limitations on the developer groups.

The first step taken toward these goals was background research into teaching theory, human-computer interaction, intelligent tutoring systems, and course management systems (also called learning management systems). Our group then proceeded to survey a group of students who were likely to have used other systems such as Blackboard. Prior to actually testing the OfCourse package, we performed an online survey of various types of companies that provide website hosting services. Finally, we began performing actual field and laboratory tests of development versions of OfCourse.

Our group accomplished four successful tests of the OfCourse software package, allowing us to analyze the system and draw numerous conclusions which were uti-

lized by the developers. We were also successful in determining a category of web hosting companies that were superior in terms of being able to host a course management system. Our background research served the important purpose of not only providing a basis for many of our evaluations, but also for providing ideas for future work that could be done in regard to OfCourse.

Unfortunately, due to time limitations, there were two goals which we were unable to accomplish. First, we had originally hoped to be able to evaluate the entire OfCourse package, including the setup scripts. However, at the time of our last laboratory test, the setup scripts and a number of key tools, such as grading and quizzes, were either inoperable or unimplemented. Second, we had hoped to have had more time to perform in-depth research into existing course management systems, which would have given us significantly more data to compare to our results.

Working concurrently with our group were two MQP groups responsible for the actual implementation of OfCourse. These groups each developed a subset of the tools, and then combined them to form the complete system. Both groups relied on our work to make a few key design decisions, especially in relation to what web programming languages to use. They also looked to us to test the product and find major problems and inconsistencies, and to make suggestions on desirable improvements.

5.2 Hosting OfCourse

The results gathered from the survey of web hosting companies and internet service providers suggest that dedicated web hosting companies are, by far, the most appropriate type of host for a course management system. These services tend to offer lower prices, more storage, higher bandwidth, and most importantly, a wider range

of web programming languages and database tools.

The most important consideration when choosing a host for an online course is ensuring that the host will provide the resources necessary to run the course management software chosen. For OfCourse, like many other course management packages, these requirements include the ability to run custom Perl and PHP programs and the availability of a MySQL database. These resources are, however, rarely available to users of most standard internet service providers and almost always found on web hosting providers. We were unable to find any national internet service providers that provided even a good subset of these services to their users, much less everything needed. Local internet service providers occasionally provided one or a couple of these resources, but finding an ISP with all of the web programming resources necessary and that also happens to be local to any given user also appears to be extremely rare. Web hosting providers, on the other hand, commonly provide all of these tools, and more, to their customers, with no limitation on physical location.

Another key, and often overlooked, requirement for choosing an appropriate host is ensuring that the acceptable use policy does not forbid commercial activity. Many instructors, especially the target audience for OfCourse, may wish to charge their students to help recover the cost of teaching the course, if not to also make some profit. Most internet service providers, however, would preclude this type of commercial use of their web space without charging hefty extra fees. Conversely, nearly all web hosting providers allow, and even expect, commercial use. The reason for this appears to be the traditional view of ISP-provided space being used for personal sites and homepages since users do not want to pay extra money for this type of page, while web hosts tend to be the preferred method of hosting small business sites.

Finally, a number of finer points continue to support the use of web hosting

providers over internet service providers for hosting course management systems. Financially, web hosts are more financially fit for individual instructors because they tend to offer very flexible pricing packages, only charging for what the customer needs. This means that hosting a course management system on a web host as opposed to an internet service provider can be anywhere from ten to fifty percent cheaper, which is often a major concern for the target instructors of OfCourse. On the technical side, web hosts tend to provide higher data transfer rates, significantly more storage space, and additional tools to aid customers in building webpages.

In all, it is quite clear from the results of our research into internet service providers and web hosting providers that the latter is the ideal solution for hosting OfCourse. They provide a service which allows the type of website that most OfCourse users would desire to run, provide the tools necessary to run the software, and do so for significantly cheaper prices than internet service providers. Therefore, it is our conclusion that any instructor looking to setup a website using the OfCourse tools should first direct their attention towards web hosting providers, and consequently, developers should target these types of platforms.

5.3 Assets and Liabilities of OfCourse

The three independent tests conducted using the OfCourse system provided much information from which conclusions can be drawn. The conclusions regarding the successes and failures of OfCourse are divided into three sections. The General conclusions section explores the successes of the implementation of the theory of OfCourse. The Tools and Interface conclusions section discusses the places in which the tools and interface of OfCourse excel, and the places in which improvement is needed. The Design conclusions section examines the positive and negative aspects

of the design of OfCourse.

5.3.1 General

The studies conducted on OfCourse confirm that the goal of being an easy-to-use course management system for the individual instructor has been achieved. All of the evaluators of OfCourse had the necessary credentials to evaluate software, all are experienced with using course management systems. All of the evaluators agreed that OfCourse has all the tools necessary to be a good course management system. Some of the evaluators mentioned tools which would make OfCourse better; these are addressed in the Future Work section. Furthermore, the evaluators generally found the system to be easy to use. Despite the presence of a few mislabeled tools, the evaluators were able to teach their classes or complete scripted tasks with little to no help. This affirms that OfCourse is easy to use for the individual instructor.

The theory behind OfCourse of placing the tools of the course management system into the course content was also examined. Despite possible biases as a result of previous experiences with standard course management systems, and despite the bland template course content webpages that had been created for the various tests of OfCourse, the evaluators were able to see the unique theory of OfCourse at work. This confirms that OfCourse satisfies the theory which it is based upon.

It is important to note that the task of having a user install the system on a server was not included in these conclusions. The OfCourse software was installed and setup ahead of time for the evaluation done by Mr. Renard and the course taught by Mr. Toth. The installation and setup of OfCourse was the primary motivation for conducting the formal laboratory test with Ms. Razzaq. Unfortunately, at the time of the test, the installation and setup of OfCourse was unable to be tested because there was no stable setup script available. The phase where the most problems will

likely occur with OfCourse is installation. Adding the task of installing and setting up the system could complicate the evaluation of OfCourse's ease of use.

5.3.2 Tools and Interface

The three tests conducted also allowed for conclusions to be drawn about tool and interface specific issues. Some of the evaluators voiced opinions about the tools which they thought were excellent. Mr. Renard thought that the discussion board was strong and had great functionality. The discussion board has many of the same features as discussion boards which are available in commercial packages. Ms. Razzaq especially liked the class listing tool. The class listing tool is simple and easy to use. The discussion board and class listing tool are the tools which exemplify the quality of OfCourse.

Although many of the tools are good, there are improvements which need to be made to some of the tools. One which needs improvement is the chat room. All of the evaluators of OfCourse commented on the fact that the chat room refreshed every five seconds was a nuisance. When a chat room conversation lags, the participants lose the feeling of social closeness which is achieved with a real-time chat room. Not only does this refreshing cause lag time between conversations, but it also has the possibility of causing unexpected behavior. In one test, Ms. Razzaq clicked the button to remove a user from the chat room, but the user was not removed. This seemed to be caused by the refreshing. For these reasons, we conclude that the chat interface must be changed not to constantly refresh.

One other area of the interface which needs improvement is the labeling of tools and items. Many of the labels in OfCourse are improperly named and do not accurately convey the functionality of the items. Mislabeled items were among the main problems mentioned by all evaluators who had experience with OfCourse. The

most notable mislabeled items were the *Archive Log* chat log button, the *Disapprove* user button, and the semantics of the polling fields. It would be very beneficial if the labels of the interface were renamed to be clear and consistent. This would greatly improve the ease of usability of OfCourse and reduce frustration when using it.

The last interface issue which caused the evaluators problems in the usability tests of OfCourse was the hierarchy of the menu and layout of items. First, tools such as the discussion board are located in many places. Each of the tools should be in one single spot which is easy to find. The menus which contain these tools must be properly labeled in order to convey their contents. Second, some tools are placed in unexpected areas. A reorganization of the menu hierarchy could remedy this. Finally, not all of the tools were immediately visible. Many of the tools are hidden in the menu hierarchy. The evaluators often had to look in multiple menus in order to find the tool they were looking for. More than only the top level of the menu hierarchy should be displayed to the user. It was recommended that a vertical hierarchy would be more efficient and easier to use. With more tools readily visible, fewer errors will occur and less time will be spent looking for the appropriate tools. If the menu items are reorganized and all displayed, the system will be much more effective and intuitive.

5.3.3 Design

Finally, the three tests conducted and our use of OfCourse provided us with conclusions about the design of the software. The software was designed in a fairly modular fashion. Each of the tools are coded in a separate file. This makes the process of upgrading tools to new versions very easy; upgrading is accomplished by simply overwriting the file which contains the tool you wish to upgrade. One other aspect of the design which is very good is the web-based setup procedure included

with OfCourse. Oftentimes, software must be configured through command line arguments; an approach which is very demanding on the user. The web-based setup ensures all of the options are displayed to the user in a way which is easy to understand and respond to. The modular tool design and the web-based setup procedure are points in the design where OfCourse excels.

The current version of OfCourse does not allow for multiple courses in one installation. If an instructor wanted to teach multiple courses, he would need to install OfCourse for each course he wants to teach. This is a direct consequence of the theory behind OfCourse; that the tools are placed inside of the course content. If an instructor were to have multiple courses, it would be expensive in both time and storage space to manage multiple installations of OfCourse.

5.4 Future Work

There exists a large wealth of future work that can be done related to OfCourse. These possibilities include further evaluations of the system and research into other course management systems and related technologies by IQP groups and various improvements and additions to the OfCourse software package by MQP groups.

Further research into existing course management systems would provide much more firm data to compare to the OfCourse results. Much of what our group was forced to compare our data to, because of time limitations, were generalizations and pre-conceived notions of other similar systems, mostly based upon our own and others' experiences with Blackboard. While we believe that this familiarity with Blackboard, along with the understanding of other course management systems that we gained through our background research, was quite sufficient to accomplish our goals and support our conclusions, a more solid basis of facts and observations

of other course management systems would likely provide added significance and additional conclusions to our work.

We also believe that there still exists a plethora of related technologies to research in the context of their significance to course management systems. Discovery of and research into these could be the most significant catalyst towards future innovation in OfCourse. For example, our research into intelligent tutoring systems has revealed an important field of computer science which has not yet been interfaced with course management systems, but appears to have strong potential. Because of this, future MQP groups will likely be attempting to integrate this type of technology into OfCourse, giving it yet another unique and vital feature. We strongly believe that this type of exploratory research by future IQP groups could reveal other such possibilities.

OfCourse itself also needs extensive additional testing, both in its current version and in future ones. OfCourse's current version contains a number of tools and features which we were unable to effectively analyze because of their incomplete state at the time of testing. These components, however, must be evaluated before the entire system can effectively be given a conclusive critical review. Future versions will also need to not only have the new and modified features evaluated, but the entire system will then need to be re-evaluated as a whole to account for the fact that even minor changes can affect the usability and effectives of the entire hierarchy of the system.

Future MQP groups also have a rich field of development opportunities available to them in OfCourse. First and foremost, there are a number of improvements to be made to the existing tools, as discovered by our research and mentioned above in our Results and Conclusions of OfCourse. Secondary to this, we believe that there are a number of more common features that should be added to the system to

bring it to a functionally similar level of other popular course management systems. The registration system should be reconfigured to automatically e-mail registrants when they are either approved for or denied access to the course by the instructor, so as to alleviate this extra step needing to be done manually by the instructor. The grade book feature should have increased functionality added to it, such as calculating averages, to once again alleviate extraneous work on the part of the instructor. It would also be beneficial to allow some sort of file upload directly within the discussion boards, so that students can attach files to their messages to share with other students and the instructor. Finally, a white board feature that could be used concurrently with the chat room utility would greatly enhance the ability for the communications tools to truly act as a virtual classroom, by allowing the instructor to create real-time illustrations for the students, and vice-versa.

There are also a couple of much more significant additions which our group feels would greatly enhance the usefulness of the system. First, based on our background research into this field, we believe that integrating some intelligent tutoring tools into OfCourse would not only considerably enhance the usefulness of the system, but would set it apart from other systems by integrating a truly useful innovative feature that has thus far barely made it outside of the academic circle. Potential applications of this type of technology, within the context of OfCourse, could involve adaptive tutorial lessons and quizzes or an interface that adapted to the user's most common activities. However, because of the broad nature of intelligent tutoring technology, the applications of it within OfCourse are truly limited only by the imagination of the designer.

Finally, based on our personal experiences with repeatedly installing OfCourse both to upgrade to new versions and to setup multiple courses for testing, we believe that the system's tools could be modularized somewhat more extensively. In this

vision, we see a single copy of the tools that could be installed and then be configured, by a simple administrative tool, to provide the same set of services to multiple courses being taught from the same website account. This would theoretically provide some significant savings in storage space for those users who have a limited amount, would make upgrading the system much easier, and would likely reduce overhead work for the instructor when managing multiple courses. We believe that this would apply to a significant number of instructors within the targeted audience, as many people who teach online courses tend to either teach more than a single course at once, or over time teach multiple revisions of the same basic course.

Appendix A

Definition of Terms

Blackboard: A software company specializing in providing course management systems to schools, businesses, and governments.

Distance Learning: Any type of learning in which the student does not have physical contact with the instructor because of distance.

CGI: Common Gateway Interface: A standard for external gateway programs to interface with information servers such as HTTP servers.

course management system: A package or set of tools used to manage and provide functionality for courses with an online component.

courseware: See **course management system**.

human-computer interaction: The study of how humans interact with computers.

intelligent tutoring system: (ITS) A learning aid which modifies its direction of questioning based on student responses.

ISP: Internet Service Provider: A company which provides access to the internet to subscribers.

MySQL: A database implementing the SQL standard.

myWPI: A course management system used at WPI, provided by Blackboard.

OfCourse: A course management system being developed at WPI, and the subject of this paper.

Perl: A popular scripting language often used through CGI to serve dynamic webpages.

PHP: PHP Hypertext Preprocessor: A popular scripting language which can be embedded into HTML to serve dynamic webpages.

SQL: Structured Query Language: A standard language for querying relational databases.

web host: A company which provides space and computer resources to subscribers who want to store information (usually web sites) on the internet.

Appendix B

Commuter Survey

B.1 Commuter Survey Questions

1. How many times do you use myWPI in an average week? In an average day?
2. Rate the ease of use of myWPI?

   Hard Easy
   1 2 3 4 5 6 7 8 9

3. What of myWPI do you like or not like? Why?
4. Which tools or features do you think are most important?
5. How effectively does your teacher use myWPI?

   Poorly Well
   1 2 3 4 5 6 7 8 9

6. How does your teacher inform you on using myWPI?
7. Which tools do you use most often?
8. What impact does myWPI have on your learning in a course?

   Useless Helpful
   1 2 3 4 5 6 7 8 9

9. How do you think myWPI could make learning in another one of your classes easier?
10. Which tools do you find most helpful?
11. In general, do you think myWPI is a good program?

    Bad Good
    1 2 3 4 5 6 7 8 9

12. If you have experience with another program similar to myWPI, how good was it?

Bad								Good
1	2	3	4	5	6	7	8	9

13. What do you think would make myWPI better?

14. Comments regarding myWPI or other online learning tools.

B.2 Commuter Results

The following table shows the responses to questions which called for a numerical answer.

Question	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12
1a	5	20	1-2	7	0	1		7	7-14	5	2	
1b		1		1	0			1	1-2	1		
2	8	9	9	8		8	8	8	8	7	8	
5	1	9		1				5	9	1	1	
8	1	9		1		1	7	3	9	2	1	
11	9	9	9	9		7	7	7	8	5	8	
12	9	9		9							5	

Table B.1: Results from the Commuter Student Survey

Questions 3,4,6,7,9,10,13, and 14 required written responses, they are as follows:

Question 3: What of myWPI do you like or not like? Why?

Respondent 1: like how it is organized, don't like the design

Respondent 2: Discussion Boards

Respondent 3: it's good

Respondent 4: discussion board

Respondent 5: to hard to get registered

Respondent 6: discussion board

Respondent 7: nothing

Respondent 8:

Respondent 9: discussion boards are good and can get grades from classes

Respondent 10:

Respondent 11: I like the discussion board

Respondent 12: I don't use myWPI.

Question 4: Which tools or features do you think are most important?

Respondent 1: discussion board

Respondent 2: Email & discussion boards

Respondent 3: don't know yet...

Respondent 4: none

Respondent 5: ?

Respondent 6: discussion board

Respondent 7: bulletin board

Respondent 8:

Respondent 9: Assignments + grades

Respondent 10: billboard

Respondent 11: the blackboard

Respondent 12:

Question 6: How does your teacher inform you on using myWPI?

Respondent 1: N/A

Respondent 2: Never

Respondent 3: No use

Respondent 4: discussion board

Respondent 5: ?

Respondent 6:

Respondent 7:

Respondent 8: not a lot

Respondent 9: says go on it

Respondent 10: doesn't

Respondent 11: They don't

Respondent 12:

Question 7: Which tools do you use most often?

Respondent 1: groups sections

Respondent 2: Discussion Boards

Respondent 3: discussion

Respondent 4: discussion board

Respondent 5: ?

Respondent 6: discussion board

Respondent 7:

Respondent 8: just class documents

Respondent 9: Assignments + discussion boards course documents

Respondent 10:

Respondent 11: Discussion board

Respondent 12:

Question 9: How do you think myWPI could make learning in another one of your classes easier?

Respondent 1: by posting the agenda there

Respondent 2: Provide online H.W. help

Respondent 3: yes

Respondent 4: would not know

Respondent 5: ?

Respondent 6:

Respondent 7:

Respondent 8:

Respondent 9: If assignments + grade were put on it

Respondent 10: if teachers used it

Respondent 11: Have teachers use it more

Respondent 12:

Question 10: Which tools do you find most helpful?

Respondent 1:

Respondent 2: Discussion Boards

Respondent 3: -

Respondent 4: don't use any

Respondent 5: ?

Respondent 6:

Respondent 7:

Respondent 8:

Respondent 9: assignments

Respondent 10:

Respondent 11: none yet

Respondent 12:

Question 13: What do you think would make myWPI better?

Respondent 1: better accessibility to all features

Respondent 2: Virtual Classroom

Respondent 3: to make it compulsory for all classes

Respondent 4: nothing

Respondent 5: ?

Respondent 6: more teachers used it

Respondent 7:

Respondent 8:

Respondent 9: Don't know

Respondent 10:

Respondent 11:

Respondent 12:

Question 14: Comments regarding myWPI or other online learning tools.

Respondent 1: keep up with the good work

Respondent 2:

Respondent 3: -

Respondent 4: decent

Respondent 5: ?

Respondent 6:

Respondent 7: I think it is a good program, but I don't know much about.

Respondent 8:

Respondent 9: NONE

Respondent 10:

Respondent 11:

Respondent 12:

Appendix C

RoBoTL Class Surveys

Survey	Number of Responses
Pre-Use	5
During-Use	1
Post-Use	0

Table C.1: Number of responses to surveys from students who were in the RoBoTL course.

C.1 Pre-Use Survey Questions

1. Which, if any, online course management system(s) have you previously had experience with?

2. Your experience with the online course management system(s) mentioned above was satisfactory (if you mentioned multiple course management systems in the previous question, please indicate which system you are rating):

Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	N/A

3. Your experience with the online course management system(s) mentioned above aided in your learning/teaching the material in a course.

Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	N/A

4. A successful online course management system should offer the following components (on a scale of 5 being the most important and 1 being the least important):

a.	Message Board	5 4 3 2 1
b.	Real-Time Chat Room	5 4 3 2 1
c.	Announcement Section	5 4 3 2 1
d.	Access to Student Grades	5 4 3 2 1
e.	Course Document Repository	5 4 3 2 1
f.	File Turn In Utility	5 4 3 2 1
g.	Online Quiz Tool	5 4 3 2 1
h.	Other (please indicate)	5 4 3 2 1

C.2 Pre-Use Results

Question	Respondent 1	Respondent 2	Respondent 3	Respondent 4	Respondent 5
1.	none	Blackboard	none	Blackboard	Blackboard
2.		Strongly Agree		Agree	Strongly Agree
3.		Strongly Agree		Agree	Strongly Agree
4a.	5	2	4	5	4
4b.	5	2	4	4	5
4c.	5	5	4	5	4
4d.	5	5	5	5	5
4e.	5	3	5	4	4
4f.	5	5	4	3	3
4g.	5	3	4	3	5
4h.					

Table C.2: Results from the Pre-Use Survey.

C.3 During-Use Survey Questions

1. Approximately how many times a week do you use the Message Board tool in OfCourse?

0 times a week	1 to 5 times a week	5 to 10 times a week	greater than 10 times a week

2. Your experience with the Message Board tool is that it has been helpful to your learning the course material.

Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

3. The Message Board tool has performed up to your expectations.

Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

4. What, if any, changes would you recommend to be made to the Message Board tool?

C.4 During-Use Results

Question	Respondent 1
1.	0 times a week
2.	Neutral
3.	Neutral
4.	

Table C.3: Results from the During-Use Survey.

C.5 Post-Use Survey Questions

1. You were satisfied with OfCourse as it was used in your learning experience. Please elaborate on your choice:

Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

2. Complete the following statement: "I feel that OfCourse has ____ my original expectations of it." Please elaborate on your choice:

fell short of	met	exceeded

3. Complete the following statement: "This website _ my learning experience." Please elaborate on your choice:

helped	hindered

a.	Message Board	5 4 3 2 1
b.	Real-Time Chat Room	5 4 3 2 1
c.	Announcement Section	5 4 3 2 1
d.	Access to Student Grades	5 4 3 2 1
e.	Course Document Repository	5 4 3 2 1
f.	File Turn In Utility	5 4 3 2 1
g.	Online Quiz Tool	5 4 3 2 1

4. Rank the following components based on how often you used them. (on a scale of 5 being the most important and 1 being the least important):

5. How would you compare your experience with the OfCourse software package to your experience(s) with any other (please specify), if any, online course management software packages?

6. What suggestions would you make for future improvements to this software?

Appendix D

Slides from Michael Bleyhl's Presentation

Using Technology to enhance your blended learning.

Michael Bleyhl
Director of e.Learning and Infrastructure
Global Learning Organization
Invensys Plc
December 2003

Learning Technology Acronyms/Glossary

- **LMS**: Learning Management System- Database driven system that tracks student records, course administration and instructor management.
- **LCMS**: Learning Content Management System- similar to a LMS but focused on "Course Content" versioning and reuse
- **ILT**: Instructor Led Training: Traditional stand-up class room learning event
- **WBT/CBT**: Web Based Training, Computer/CD-ROM Based Training- Self Paced Asynchronous delivery method for learning
- **AICC**: Aviation Industry CBT Committee – Standard for E.Learning course tracking
- **SCORM**:Sharable Courseware Object Reference Model - Standard for E.Learning course tracking
- **RIO/RLO**: Reusable Information Object / Reusable Learning Object
- **Web-Cast**: Either a live or recorded event delivered via the Internet/Intranet usually a "one to many" approach e.g. WebEx, Interwise, Placeware, LearnLink
- **Virtual Classroom**: Internet/Intranet based collaboration tool that allows for a high degree of live interactivity between participants e.g. Centra

Statistics

- By 2004, what percentage of students will be taking higher education-related e-learning?
  - 5%
  - 10%
  - 15%
  - 25%

2

Statistics

- Eventually, e-learning will replace classroom learning altogether.
  - False. E-learning is an essential part of the distributed learning landscape, but the need for in-person instruction won't go away, especially in the areas of skills development and behavior modification.

3

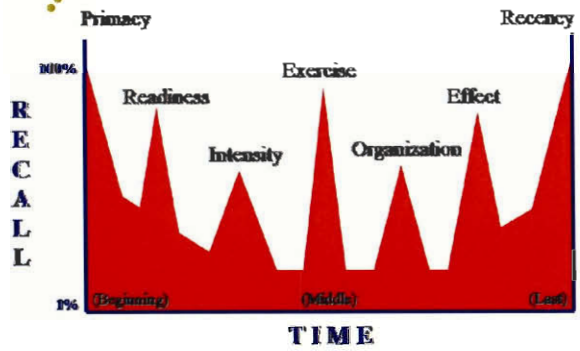
Statistics

- Analysts predict that by 2004, e-learning will comprise 30% of enterprise training.
  - False. It is predicted that in 3 years, e-learning will replace classroom training as the favored corporate training method.

4

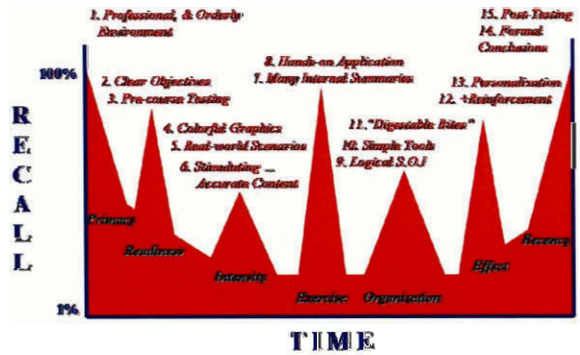
Learning is a social event

A Little Learning Theory



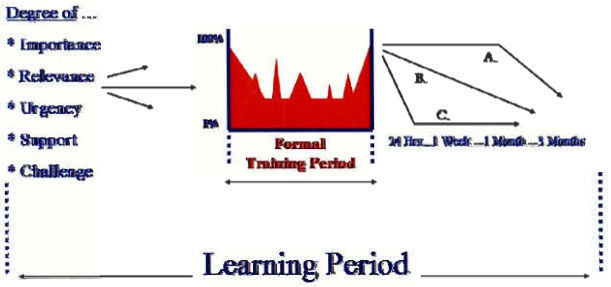
Theory-into-Application (T.I.P.)

What "Competent" Trainers Ensure during Training Sessions ...



Individual Learner's Perspective ...

Training is only ONE part of a person's Learning Process.
What happens "before" & "after" is often equally important
as what happens "during" training.

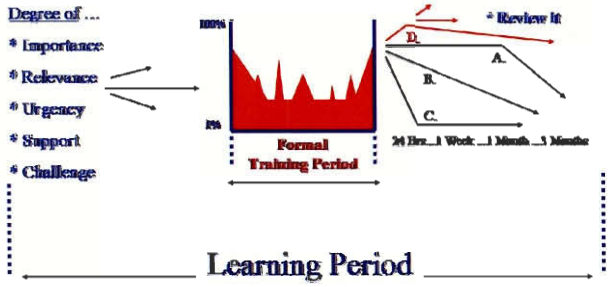


Individual Learner's Perspective

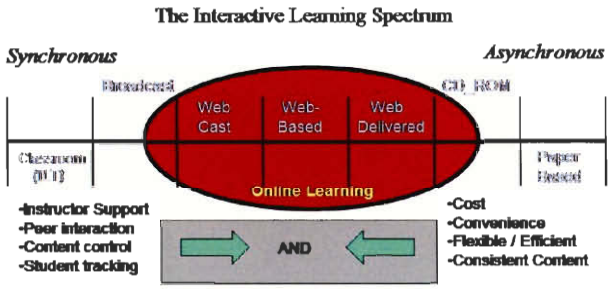
(or ... Lose It Forever !)

Training is only ONE part of a person's Learning Process.
What happens "before" & "after" is equally important
as what happens "during" training

* Reward it
* Teach it
* Apply it
* Share it
* Review it



A Distributed or Blended Solution



Learning Technology: Use it appropriately

- Technology based learning works for these types of events:
  - Knowledge transfer or informational sessions
  - One to many conferences with geographically dispersed audiences
  - Just in time learning events or reference tools
  - "Hard" Skills learning e.g. Programming, How to..., New Products, etc...
  - Pre work for ILT events and Post work for ILT events
  - Enterprise wide information sharing
  - Teams who have worked together in the past
- Technology based learning does not work for these types of events:
  - Newly formed teams especially with culturally diverse populations
  - "Soft" Skills learning e.g. Interviewing, Phone etiquette, Sales, Counseling, Leadership, etc...
  - "Hands on" practice e.g. Open Heart Surgery
  - Areas with a lot of distractions

11

Invensys University Learning Strategy: Blending Formal and Informal Learning

Formal Learning

- Instructor led programs
- Certifications
- Structured Web based learning modules
- Virtual Classrooms
- Assessments and evaluations

Informal Learning

- Articles
- On the Job Training
- Web searches
- Discussion groups and chat rooms
- Virtual meetings with Peer groups

12

Invensys University's sample of Blended Curriculum

Formal

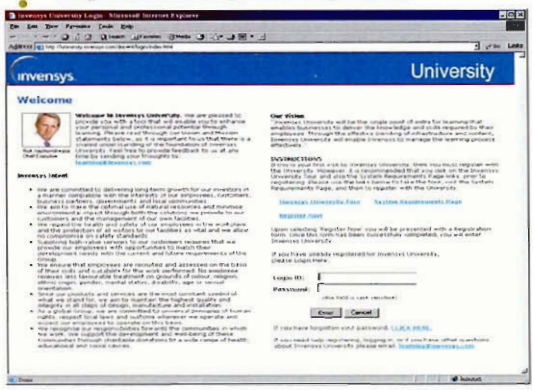
- EH&S Certifications
- New and Emerging Leaders Programs
- Sales Training Programs
- Tuck Business School online learning
- INVEST curriculum
- Project Tracking System e.Learning
- Lean Supply Chain e.Learning
- Project Management Certifications course

Informal

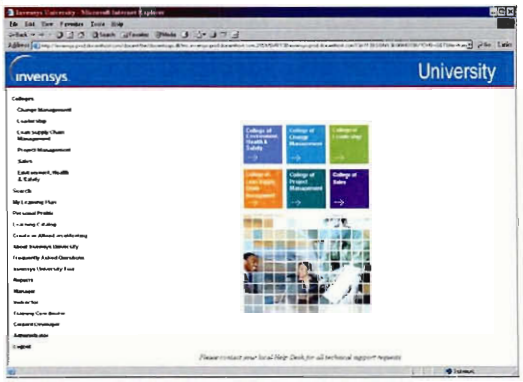
- Harvard Business Review articles
- Discussion Groups
- eMeetings
- Brown bag Web-casts

13

http://university.iinvensys.com



14



15

Questions ?

Learning is for a lifetime

Bibliography

[1] ANDREW ARCH. Auxiliary Benefits of Accessible Web Design. `http://www.w3.org/WAI/bcase/benefits.html`, March 2002.

This W3C article describes the many business, technical and other benefits to the organization above and beyond the straightforward benefits to people with disabilities that can be realized by applying the Web Content Accessibility Guidelines to web sites.

[2] KEVIN D. ASHLEY and VINCENT ALEVEN. Toward an intelligent tutoring system for teaching law students to argue with cases. In *Proceedings of the third international conference on Artificial intelligence and law*, pages 42–52. ACM Press, 1991. ISBN 0-89791-399-X.

This paper describes a research project to devise and test an intelligent tutoring system that teaches law students to argue cases in a simulation environment.

[3] MICHAEL BLEYHL. Using Technology to enhance your blended learning, December 2003.

A presentation given at WPI by Michael Bleyhl, the director of e-Learning at Invensys Plc. Topics covered included teaching theory, problems with distance learning, methods of blending synchronous and asynchronous learning tools, and the differences of corporate CMS uses as opposed to academic uses.

[4] JOHN D. BRANSFORD, ANN L. BROWN, and RODNEY R. COCKING. *How People Learn: Mind, Brain, Experience and School*. National Academy Press, 1999. ISBN 0-309-06557-7.

The result of a research commission on how people learn. It addresses what instructors need to be aware of when trying to impart new knowledge to students. It addresses pitfalls that can arise, and ways around them.

[5] JUDY BREWER. Overview of the Web Accessibility Initiative. `http://www.w3.org/Talks/WAI-Intro/slide1-0.html`, 2003.

This is a W3C slide presentation that answers many questions about web accessibility such as what is web accessibility, why accessibility is important for people with or without disabilities, what resources are available to make web sites accessible, and what actions are useful in promoting accessibility.

[6] WENDY CHISHOLD, GREGG VANDERHEIDEN, and IAN JACOBS. Web Content Accessibility Guidelines 1.0. `http://www.w3.org/TR/WCAG10/`, May 1999.

These guidelines explain how to make web content accessible to people with disabilities. The guidelines are intended for all web content developers (page authors and site designers) and for developers of authoring tools. The primary goal of these guidelines is to promote accessibility.

[7] CIA. World Fact Book 2003. `http://www.cia.gov/cia/publications/factbook/`, 2003.

The CIA's World Fact Book provides a plethora of information and statistics on all the countries in the world.

[8] MICHAEL FELDSTEIN. Ignore Usability at Your Peril. *eLearn*, volume 2002(9):page 3, 2002.

This short column explores the importance of good usability for a web based learning course management system. The author explains how users can be scared off by poor usability, and that expensive course management systems often do not do tests to determine the usability of their interfaces

[9] REVA FREEDMAN, SYED S. ALI, and SUSAN MCROY. Links: what is an intelligent tutoring system? *Intelligence*, volume 11(3):pages 15–16, 2000. ISSN 1523-8822.

This source gives a definition of intelligent tutoring systems and a very brief overview of the basic theory behind them.

[10] ARIF IQBAL, REINHARD OPPERMANN, ASHOK PATEL, and KINSHUK. A Classification of Evaluation Methods for Intelligent Tutoring Systems. `http://www.fit.fraunhofer.de/ oppi/publications/SE99_ITS-Eval.pdf`.

This paper proposes a classification of evaluation methods to simplify the selection task. The classification is based on two primary questions relating to the target of evaluation and learning environment in which the evaluation would be pursued.

[11] CRAIG M. KAPP. Implementation & Evolution of a Course Management System. In *Proceeding of the 30th annual ACM SIGUCCS fall conference on User services conference*, pages 264–266. ACM Press, 2002. ISBN 1-58113-564-5.

This paper details the College of New Jersey's efforts to create an online course management system. Previous to this effort they had several dispersed and non-integrated software solutions for course management. They decided to look into a complete course management system, and eventually decided to create an "in-house" system.

[12] David G. Kay. Large Introductory Computer Science Classes: Strategies for Effective Course Management. In *Proceedings of the twenty-ninth SIGCSE technical symposium on Computer science education*, pages 131–134. ACM Press, 1998. ISBN 0-89791-994-7.

This paper discusses several points with regard to the problems facing large introductory computer science classes. It uses the term course management not specifically refering to an online management tool, but as an overall approach to a course. The topics it brings up are applicable whether or not one is just concentrating on web-based tools.

[13] Robert Luke. Courseware Accessibility: Recommendations for Inclusive Design. In *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT'01)*, page 381. IEEE Computer Society, 2001. ISBN 0-7695-1013-2.

This paper presents an overview of the results of research into courseware accessibility. Recommendations for inclusive design are presented within disability-specific requirements for accessible courseware.

[14] Christoph Meinel, Harald Sack, and Volker Schillings. Course Management in the Twinkle of an Eye - LCMS: a Professional Course Management System. In *Proceeding of the 30th annual ACM SIGUCCS fall conference on User services conference*, pages 281–283. ACM Press, 2002. ISBN 1-58113-564-5.

LCMS is a freely available course management system. It was built up from a registration tool, and is now used as a complete system. One of the more interesting parts of LCMS is its ability to understand LaTeX documents and convert them to html. Many science and math courses require such formatting for homeworks, labs, and handouts.

[15] Corina Mulholland and Wing Au. Improving Courseware Evaluation via the Use of a Model. In *Proceedings of the International Conference on Computers in Education*, page 1369. 2002. ISBN 0-7695-1509-6/02.

This paper presents an overview of a model designed to aid evaluators of courseware by providing a formal structure for analyzing course management systems.

[16] Ann Quigley. Usability-tested e-learning? Not Until the Market Requires it. *eLearn*, volume 2002(2):page 1, 2002.

This article explores more of the lack of usability of e-learning systems. The author pleads for developers of distance learning systems to make the simpler, user centered, and rely more on the practices that human-computer interaction studies have revealed to be effective.

[17] Jeff W. Rickel. Intelligent Computer-Aided Instruction: A Survey Organized Around System Components. In *IEEE Transactions on Systems, Man, and Cybernetics, Vol. 19, No. 1*, pages 40–50. 1989.

This paper presents a survey of the issues and previous research in Intelligent Computer-Aided Instruction (ICAI). Various learning scenarios are discussed, including computer coaches, gaming environments, mixed initiative dialog, Socratic tutors, articulate experts, interactive simulation and discovery learning.

[18] Irene Luque Ruiz, Gonzalo Cerruela Garcia, and Miguel Angel Gomez-Nieto. A Java dialog system for use in computer-aided teaching. In *Proceedings of the 2nd international conference on Principles and practice of programming in Java*, pages 71–74. Computer Science Press, Inc., 2003. ISBN 0-9544145-1-9.

This study shows the use of a Java dialog system used to create a teacher-student conversation using an Intelligent Tutoring System.

[19] Ben Schneiderman. *Desiging the User Interface, Third Edition.* Addison Wesley Longman, 1998.

This section of the book discusses expert review and laboratory testing of software.

[20] Julika Siemer and Marios C. Angelides. Embedding an intelligent tutoring system in a business gaming-simulation environment. In *Proceedings of the 26th conference on Winter simulation*, pages 1399–1406. Society for Computer Simulation International, 1994. ISBN 0-7803-2109-X.

Intelligent tutoring systems have recently been added to gaming-simulation environments to enhance their pedagogical effectiveness as a teaching environment. This paper presents an example of this which embedded an ITS into the Metal Box Business Simulation Game.

[21] A. Vantaggiato. Some considerations on intelligent tutoring systems. In *Proceedings of the first international conference on Industrial and engineering applications of artificial intelligence and expert systems*, pages 1163–1167. ACM Press, 1988. ISBN 0-89791-271-3.

The goals, functions and structure of Intelligent Tutoring Systems are described in this paper. A prototype is then presented: SEDAF, a tutor in elementary Calculus. Particular emphasis is given to two major problems: the diagnose of a student error, and the interpretation and modeling of his knowledge.