

# Developing Courseware to Support Online Discussion

by

Benjamin Luke Wang

Submitted to the Department of Electrical Engineering and Computer  
Science

in partial fulfillment of the requirements for the degree of

Master of Engineering in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

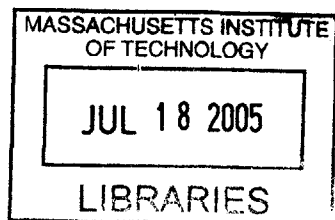
May 2005 *[June 2005]*

© Massachusetts Institute of Technology 2005. All rights reserved.

Author .....  
Department of Electrical Engineering and Computer Science  
May 19, 2005

Certified by .....  
Jonathon Cummings  
Assistant Professor  
Thesis Supervisor

Accepted by ...  
Arthur C. Smith  
Chairman, Department Committee on Graduate Students



**BARKER**



# Developing Courseware to Support Online Discussion

by

Benjamin Luke Wang

Submitted to the Department of Electrical Engineering and Computer Science  
on May 19, 2005, in partial fulfillment of the  
requirements for the degree of  
Master of Engineering in Computer Science and Engineering

## Abstract

For my Master of Engineering thesis, I designed and implemented an online discussion forum in an attempt to improve student learning. Aided with the help of Professor Jonathon Cummings 15.351 course (Introduction to Innovation and Entrepreneurship), I measured student and peer-to-peer interaction via online discussion boards and user-generated interest in the subject matter. Based on the data received from this discussion board, I have found that students have their own optimal learning environments, and that anonymity can affect the way students interact with one another.

Thesis Supervisor: Jonathon Cummings  
Title: Assistant Professor



## Acknowledgments

I would like to thank my supervisor, Professor Jonathon Cummings, for the opportunity to work on this project. Though tough at times, he kept me on my toes, making sure that progress was being made, on both the code as well as this thesis.

Furthermore, I would like to thank my parents for their unending support. Without their love and encouragement, I would not have had the opportunity to experience an education here at MIT in my undergraduate years, as well as through the Master of Engineering Program. Finally, I would like to thank God for the many blessings he has poured down upon me, and my faith is the one thing that keeps me going in times of trouble and distress.



# Contents

<b>1</b>	<b>Introduction</b>	<b>15</b>
1.1	Introduction to Online Learning . . . . .	15
1.2	Foundational Work . . . . .	16
1.3	15.351: Innovation and Entrepreneurship . . . . .	17
1.3.1	Class Overview . . . . .	18
1.3.2	Online Discussion Board Participation . . . . .	18
1.4	Technical Background . . . . .	19
1.4.1	Terminology . . . . .	19
1.4.2	PHP . . . . .	20
1.4.3	Apache, MySQL, and HTML . . . . .	21
1.5	Hypothesis and Goals . . . . .	22
<b>2</b>	<b>Design and Development of the Online Discussion Board</b>	<b>25</b>
2.1	Overview . . . . .	25
2.2	Functionality . . . . .	26
2.3	Design . . . . .	29
2.4	Issues . . . . .	30
2.4.1	User Identification and Uniqueness . . . . .	30
2.4.2	Modularity . . . . .	31
2.5	User Experience . . . . .	32
2.5.1	Biographical Pages . . . . .	33
2.5.2	View All Threads . . . . .	33
2.5.3	Class Lists . . . . .	34

2.5.4	Statistics Page . . . . .	34
<b>3</b>	<b>Administration and Maintenance</b>	<b>35</b>
3.1	Overview . . . . .	35
3.2	Design . . . . .	36
3.3	Maintenance Issues . . . . .	37
3.3.1	Concurrent Data Entry . . . . .	37
3.3.2	Antivirus Protection Problems . . . . .	37
3.3.3	Customer Service . . . . .	38
3.3.4	Concurrent Code Fixes . . . . .	39
<b>4</b>	<b>Learning</b>	<b>41</b>
4.1	Experimentation . . . . .	41
4.1.1	Setup . . . . .	41
4.1.2	Results . . . . .	43
4.2	Trends and Patterns . . . . .	45
4.2.1	In-Class Participation Scores . . . . .	45
4.2.2	Online Participation Scores . . . . .	46
4.2.3	Correlations . . . . .	47
<b>5</b>	<b>Conclusion</b>	<b>49</b>
5.1	Progress Update . . . . .	49
5.2	Lessons . . . . .	50
5.2.1	Ramp-up Period . . . . .	50
5.2.2	Standards and Implementations . . . . .	51
5.2.3	Experimentation . . . . .	51
5.3	Future Work . . . . .	52
5.3.1	Modularity and Cleanup . . . . .	52
5.3.2	New Features and Modules . . . . .	53
5.3.3	New Version Releases of Phorum . . . . .	53
5.4	Contributions . . . . .	53



<b>A</b>	<b>Screen Shots</b>	<b>57</b>
<b>B</b>	<b>Implementation Details of the Discussion Board</b>	<b>69</b>
B.1	Database Structure . . . . .	69
B.2	Code . . . . .	71



# List of Figures

2-1	Phorum Flow Chart . . . . .	28
2-2	Overall Structure of the Phorum Subsystem . . . . .	29
2-3	Architetural Design of Phorum . . . . .	32
4-1	Experimental Setup . . . . .	42
4-2	Experimental Results for Mean Values on Incoming Entropy . . . . .	44
A-1	A user is currently logged into the homepage. . . . .	58
A-2	The discussion page shows all active and previous forums via URL links. . . . .	59
A-3	Accessing the feedback page is possible from the main website and from Phorum. Here a user is emailing the administrators about a posting problem he is having. . . . .	60
A-4	A user has just entered Phorum. He is looking at the two assigned readings and forums for Lecture 11. . . . .	61
A-5	Upon entering Forum 11A, the user sees a list of available threads. . . . .	62
A-6	The user clicks on a thread to view. He has no ability to respond, but the header at the top lists the option to post a new topic. . . . .	63
A-7	After clicking on the New Topic link, the user now has the option of posting a new topic. . . . .	64
A-8	The user has posted a new topic, and it is now viewable on list.php. The ability to post new topics is no longer available on the header. . . . .	65
A-9	A few weeks later, the anonymity change has been made. Students can still see the threads, but the other users are anonymous. However, users can still see their own posts. . . . .	66

A-10	When reading a thread, the user still cannot ascertain the identity of the other user. However, the correct posting window is open, so the user can respond to the post. . . . .	67
A-11	Upon responding to the post, the user can now view the identity of the authors of all posts. . . . .	68

# List of Tables

2.1	The general <i>students</i> table for access to the 15.351 home page . . . . .	30
2.2	The <i>phorum_users</i> table for access onto the Phorum Discussion Board	31
4.1	Tables in the Database Used by 15.351 . . . . .	44
4.2	Section A: Correlations in Class and Online Participation . . . . .	47
4.3	Section B: Correlations in Class and Online Participation . . . . .	47
B.1	Tables in the Database Used by 15.351 . . . . .	70
B.2	PHP Files Used by the System . . . . .	73



# Chapter 1

## Introduction

### 1.1 Introduction to Online Learning

The state of higher education in the United States is undergoing vast changes across the university campus. The price of tuition is rising every year; Peter Drucker, one of the leading experts in management and education strategy, argues that “the cost of higher education has risen as fast as the cost of health care” [8]. In addition, advances in technology continue to help evolve the classroom, ranging from wireless internet capabilities, online distance learning, or the breadth and thoroughness of search engines. However, technological advances do not necessarily imply improvements in the learning of our higher education establishments. Courseware needs have evolved beyond the requisite instruction and grade assessment; additional mediation, tutoring, and feedback is now available via online course websites, which may include supplementary features like online discussion forums, web-based lecture slides and audio files, databases of course information, and technical tutorials and labs.

Higher education is at a serious crossroads: the strategic academic technology path currently being taken may not be the better one. Privateer argues that we can either continue an administrative approach to the management of learning, or we can seek another more promising change by “redesigning instructional technology in terms of being both a strategic and cognitive tool” [11]. Rather than continue our battle of knowledge acquisition and rote memorization against the fast-changing

world, Privateer argues that we should be more focused on the opportunity to enhance our intelligence by learning the reasoning behind technological change, which may lead to greater understanding of the world around us. For my Master of Engineering thesis, I propose to design and test a system that will consider the class participation and discussion capabilities of students, and how it can be affected by new methods of collaboration and feedback. The online discussion tool will allow students to ask and answer questions posed by one another, which will stimulate thinking and innovation.

## 1.2 Foundational Work

If we are to further understand higher education and the learning process from a computational point of view, we must first consider the groundwork that has already been laid by researchers with the same vision and motivation. In a growing fight to improve the state of higher education, there are instances of university courses and professors no longer having the responsibility over all facets of a course (design, development, delivering, mediating, and assessing grades); rather, faculty roles have become “unbundled” and reconfigured in attempts to improve learning. With the emergence of virtual online postsecondary institutions nationwide like the University of Phoenix (UoP) and the Western Governors University (WGU), more and more opportunities have arisen to make it easier for students to obtain an education. The new idea behind these postsecondary opportunities is that “rather than incorporating the responsibility for all technology-and competency-based functions into a single concept of ‘faculty member,’ WGU disgregated faculty instructional activities and assigned them to distinct professionals” [10]. In other words, the role of the university and its accompanying faculty are evolving from jack-of-all-trades faculty members, to more specialized individuals, where up to five distinct individuals are involved with:

- Designing the course or curriculum
- Developing the course or curriculum by selecting appropriate instructional method and course materials, or creating those course materials



- Delivering the subject matter previously selected either in person (lectures, etc.) or through the use of various forms of media
- Mediating (also called tutoring) the learning process, which helps students understand materials in ways tailor to their learning styles and levels of understanding; and
- Assessing individual student learning through appropriate methods and assignments designed to certify the attainment of a given level of competence.

Through the emergence of these new approaches to secondary education, the groundwork for new methods of university instruction and edification has also been laid. Additionally, critics and university officials feel the need for change in course websites and available information is imminent. With the materialization of search engines, such as Google, quickly outpacing and outdistancing itself from the now-obsolete libraries and card catalogs of books and journals, librarians and faculty members are now favoring anything that leads students away from Google and toward vetted scholarly material. Scott Carlson contends that, as libraries and academics hope, course websites will soon become a locus of research activity for students that will “introduce them to new and better sources of information and help wean them away from search engines” [7]. If we are to keep our students from becoming too reliant on external sources, a change to higher education and its constituent technological tools is necessary.

### **1.3 15.351: Innovation and Entrepreneurship**

Innovation and Entrepreneurship (15.351) is a course taught at the Sloan School of Management by Professor Jonathon Cummings. This class meets twice a week, on Tuesdays and Thursdays. The majority of the class is made up of MBA candidates in the Sloan School of Management.

There are two main objectives to the course. The first objective is to give students a sense of why building a rich understanding of innovation is both exciting and critical

to modern managerial practice. The second objective is to give its students a solid grounding in the tools necessary to manage it.

The course is divided into two types of class sessions: lectures and case studies. Lectures involve direct teaching from Professor Cummings, and the case studies are interactive discussions that look into established and entrepreneurial firms that have had successes and failures in business. Some examples of case studies include: IDEO Product Development, The BMW Digital Car Project, and Corporate New Ventures at Proctor and Gamble. Along with each lecture, the students are assigned two readings that deal with the lecture's discussion content.

### **1.3.1 Class Overview**

To go along with the course, the grading for the course is broken up into two constituent parts:

- Class Participation 40%
- 4 out of 5 reaction papers 60%

The Class participation grade, in turn, is divided into two components that comprise the total score: an in-class participation score and an online participation score. The online participation score is based on participation in the course website's discussion board. Each student is given a username and password for login verification.

### **1.3.2 Online Discussion Board Participation**

For each lecture date, the students are required to use the online discussion forum to post thoughts, ideas, and inspirational concerns and criticisms about the assigned readings for that day. Students are restricted to posting one new topic and one reply in every forum. From the end of the class prior to the lecture until the morning of the lecture, the students are required to start a new topic thread by posting an inspirational and thought-provoking idea for each assigned reading. Following each

lecture, the students also have until the following class to reply to one other topic that was posted by another member of the class.

For example, on the Thursday before the following week's Tuesday lecture, the students are reminded that they are responsible for the required readings for Tuesday, and they have until the 3:00 AM Tuesday morning to post one new topic for each reading. However, they are not allowed to respond to any topic, and the one topic posting limit is strictly enforced. From 3 AM until the end of class on Tuesday, the forums are frozen, with no posting or replying privileges given to anyone; the users can only browse and read other topics. After the end of class on Tuesday, the students have until 3 AM Thursday morning to reply to exactly one topic posted by another student in the class.

The online participation score uses the posts and replies made by the students to assign grades. A particular student is said to receive a "hit" whenever another student in the class responds to a topic he/she started. A student's online participation grade in the course is based on the number of "hits" he/she receives with respect to the rest of the students. Therefore, it would be in the student's best interest to post funny, witty, or insightful comments and/or questions about the assigned readings in hopes of garnering enough interest so the other students will reply. Since they are restricted to replying to only one topic, they should also be selective in replying to the thread that is most insightful to them.

## **1.4 Technical Background**

### **1.4.1 Terminology**

To simplify the understanding of this thesis, any future use of the term "client" will refer to the HTML browser. Any action on the client side are actions performed by a student or user on a web browser (such as Internet Explorer, FireFox, Mozilla, and Netscape). Therefore, client-side actions are performed purely by the students and users by interfacing with their web browser and computer. On the other hand, the

“server” will refer to the web server that hosts and executes all the PHP scripts. “Server-side” actions will include any time a PHP script is invoked and run by a user, and how the information is presented back to the user, or client. The database is also a separately-run server, but for clarity this part will be described as the “database.”

The term “user” describes any of the researchers, students or other users that are using the system. This involves any course website access, viewing Phorum topics, and responding with their own replies. The word “administrators” will refer to Professor Cummings, who does not actively interact with the Phorum learning bulletin board, but monitors its activity daily, and is interested in observing patterns and information resulting from this learning effort. In addition, “developers” will include those people who are directly involved in the development and maintenance of the system.

To avoid confusion regarding the online discussion board, it will be referred to as “Phorum,” taken from the online open-source template that was used to build the discussion board. Furthermore, Professor Cummings taught two separate sections of the course. Both sections met on Tuesdays and Thursdays, with one section taught from 10:00 AM to 11:30 AM, and the other one from 1:00 PM until 3:30 PM. For the sake of ease and clarity, I will refer to the morning section as “Section A,” and the afternoon section as “Section B.” Finally, 15.351 is known as Innovation and Entrepreneurship, and will be referred to as “iande.”

## **1.4.2 PHP**

PHP is a scripting language that is “designed for the development of dynamic web pages” [5]. It is a language embedded into HTML, and is known for being useful in enabling easy and rapid development of online web applications. PHP is an open source language with no corporate support, but with wide community backing. While there is no single person responsible for PHP support, there is an abundant amount of documentation and tutorial information online, and, much like any programming language, there are functions that exist for all sorts of purposes. PHP is also equipped with a full set of SQL functions, simplifying the use of databases in conjunction with PHP-created web pages. This aspect of PHP makes it enticing to web designers; the

use of PHP with SQL is well-documented and very common. This project was also implemented with this combination of tools.

There is one other important aspect regarding PHP that has significant implications that is worth mentioning: PHP is a purely server-side scripting language. While PHP can create HTML pages dynamically as a result of an HTTP query, PHP cannot perform any actions from the client-side. As a result, PHP is quite limited in what it can do for the user interface. Unlike Java applets or other normally compiled programs, PHP has limited and indirect control over the user interface. Any client-side action handling, such as error checking, animation, and effects, must be handled by Javascript; PHP dynamically creates Javascript functions, which handle any client-side effects.

### **1.4.3 Apache, MySQL, and HTML**

The other technologies used in my implementation of the Phorum discussion board are the Apache Web Server, a MySQL database server, and HTML. These are become more common in every-day computer use, yet I will quickly describe each technology, why it is used within Phorum, and a quick cost/benefit analysis.

The Apache Web Server is an open source web server that has been in existence since 1995. It is developed and maintained by the Apache group, which has now grown to become an open source community that is developing more than simply this web server. Apache prides itself on projects that are “characterized by a collaborative, consensus-based development process, an open and pragmatic software license, and a desire to create high quality software that leads the way in its field” [1]. The Apache server is a good choice for this project because it performs well compared to proprietary web servers, there is plenty of online documentation, and its stability is very dependable. Moreover, there are many plug-ins and additional tools that are compatible with Apache, such as PHP and HTML.

The MySQL database is well-known and proclaims that it is “The World’s Most Popular Open Source Database,” and it is developed and maintained by the MySQL group [3]. While there are performance-based issues related to MySQL and potentially

better open-source databases available on the Web, MySQL has recently become more popular due to its compatibility with PHP. Development combining PHP and MySQL is well-documented online, and is very simple to learn.

HTML is currently the default interface on the Internet. Short for *Hyper – Text Markup Language*, its implementation can be accessed and used on the Web regardless of operating system and web browser. This autonomy makes it popular amongst web interfaces, and is used for nearly all forms of web development. PHP, Apache, and MySQL are all used to output HTML pages that the user can read, process, and use.

## 1.5 Hypothesis and Goals

At the heart of this thesis lies the social networking aspect of the discussion board Phorum. Jackson argues that current evaluation strategies of learning technology “fail to illustrate subtle links between student experiences with learning technology and various outcomes of interest” [9]. He asserts that accurate evaluation of learning technology involves observation, evaluation, and good research design. I hope to use these three concepts in my analysis of the 15.351 discussion forum.

Using the online interaction amongst students via the bulletin board, plenty of relationships and dependencies can be formed based on how many times they interact; i.e. Student A responding to a thread started by Student B, and vice versa. In Chapter 4, I will delve into the analysis of the posting and replying tendencies of the students in an attempt to find learning based on their network interactions. I will examine the diversity of posts made by students within the Phorum discussion board.

Halfway through the course, changes were made to the system guaranteeing anonymity in the system; students were no longer able to see the author of a new topic until after they had replied, or until after the time frame for open posting had been closed. It is my belief that this anonymity will cause diversity of replies to decrease. Prior to anonymity, students could possibly base their outgoing replies on the author of a particular thread. With author anonymity, hits will no longer be based on the name

behind the author, but on the content of the post. This thesis seeks to answer this question, as well as other endeavors regarding online interaction, with the use of the Phorum discussion board.





## Chapter 2

# Design and Development of the Online Discussion Board

### 2.1 Overview

The Online Discussion Board used for 15.351 is based off of the open source shareware of Phorum [4]. Phorum is a web based message board written in PHP, and it prides itself on being “designed with high-availability and visitor ease of use in mind.” In previous years, the 15.351 website has used older versions of Phorum, but with the release of Phorum Version 5.0 in 2004, the newer version was available for download and customization for the 2004 version of 15.351.

The online discussion board is intended to allow students in iande the ability to enter ideas and thoughts about class readings easily in the form of topic posts and replies. While I assumed most of the students would be very comfortable with computers, usability was still an important issue. Online discussion and community would possibly be seen as too abstract a layer of communication to some, and others questioned the privacy and security of posting their thoughts in an online forum. These concerns needed to be addressed, for a good user experience is crucial in assuring participation and quality of data. The online forum needed to be extremely reliable and consistent in order to develop a level of trust with the students.

There were also a two other factors that galvanized the development of the sys-

tem. One is the ability to easily modify or update the different PHP files without sacrificing the operation of the system. The other factor is the general presentation and the professional look of the system. This chapter will talk about the design and implementation of this system, as well as some of the main issues during development.

## 2.2 Functionality

The flow of Phorum is designed with user friendliness in mind. On one hand, there is a default flow that a first time user can take with little thought or effort. This flow takes the client directly to a particular day's forums from the discussion.php page from the 15.351 Innovation and Entrepreneurship home page. Depending on the link the user chooses, the browser directs the client to a page showing the two forums containing the assigned readings for that day.

Additionally, there were reminder emails sent out the day prior to a posting deadline, reminding the students about their posting duty. The emails included direct links to the forums, which would make navigation to the actual posting page literally one click away.

The following is a brief explanation of each step in the Phorum process (See Figure 2.1):

- **User Login and Initialization** The first things the system must do is verify the user. The username and password are checked with the database for accuracy. If there is a match, the user's information is retrieved and the user can begin using Phorum.
- **15.351 Index** The home page of the 15.351 website will show any relevant readings for the day. If there is an assigned reading, the index.php page will show links to the correct Phorum page. Alternatively, students can browse the discussion.php page within the 15.351 website to choose a forum.
- **Forum Selection** This step assumes the user has successfully entered the Phorum subsystem by choosing a lecture date which corresponds to a pair of forums.

Inside the Phorum index.php page, the user has two forums to choose, where each forum responds to a different assigned reading for that day. The user must post and reply in both forums eventually, so the order he enters the forums does not matter.

- **Post a Thread** If the student is entering Phorum during the correct time period, he/she has the option of posting a new thread. The correct posting time is the period from the session immediately before a lecture until the early morning of the lecture. For example, if it was a lecture on Tuesday morning, the proper time period to post a new topic is from Thursday afternoon until Tuesday morning at 3 AM.
- **Read other Threads** The other part of the Phorum process is replying to topical threads. Users read over the topics started by the other students, and choose one topic to reply to. They are encouraged to reply to the one topic that inspires and challenges them the most. Since they are limited to only one reply, this reply should be made in response to their favorite topic.
- **Post your Reply and Receive Hits** After all posts have been made, new topic posting privileges are frozen, and replying begins. This happens immediately **AFTER** the lecture, and until the early morning of the next class period. For our above example, replying would be enabled from Tuesday afternoon until Thursday morning. At the same time, other students are replying to other posts as well; this is when “hits” are received on a topic started by a user.

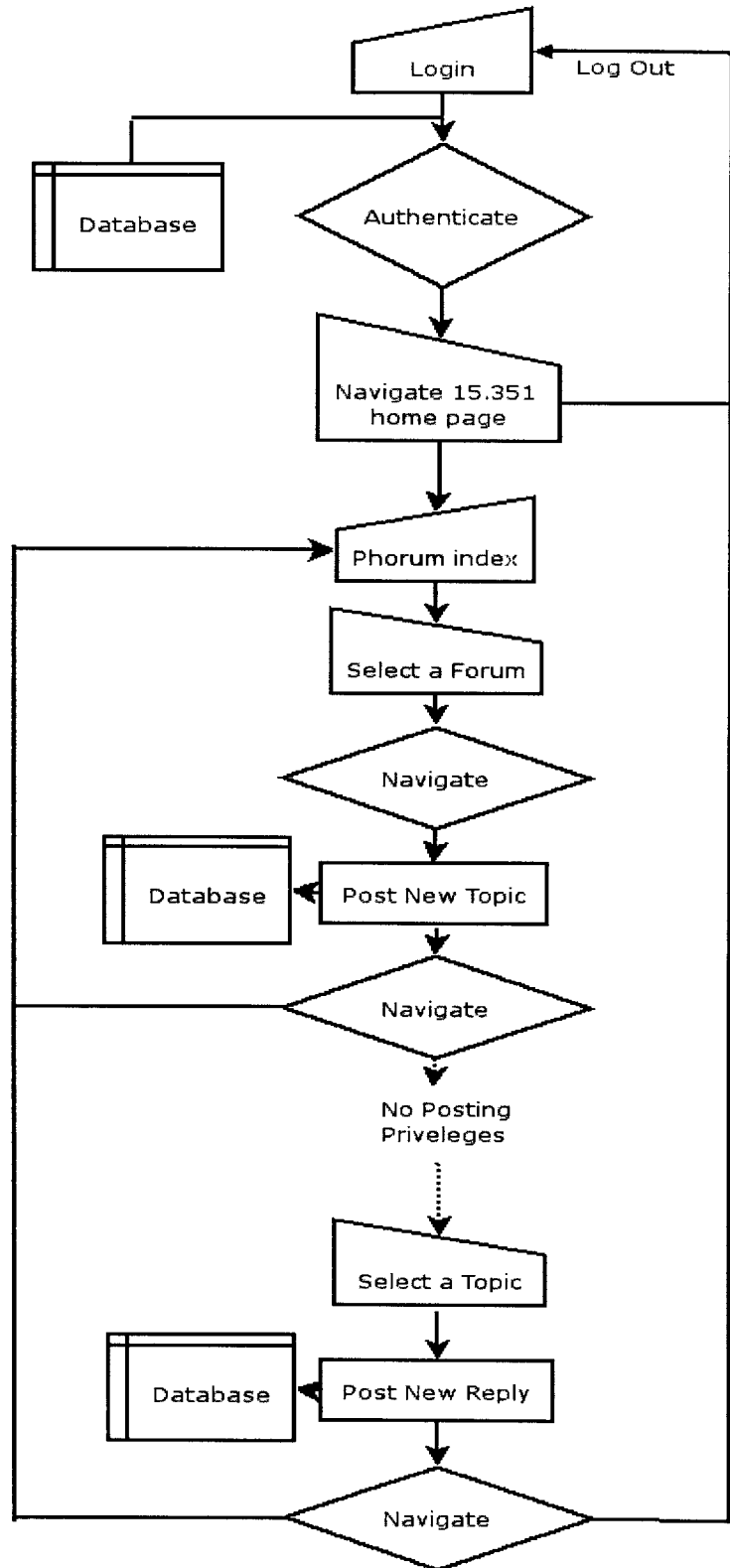


Figure 2-1: Phorum Flow Chart

## 2.3 Design

The basic design of Phorum is straight forward. The HTTP connection with the client browser is handled by the Apache Web Server. The HTTP request then calls the appropriate PHP script to interact with the MySQL database. The resulting HTML page from the PHP is then sent back to the client by the Apache Web Server.

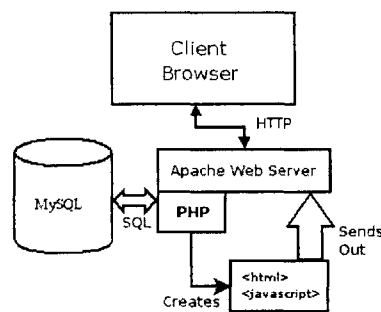


Figure 2-2: Overall Structure of the Phorum Subsystem

The code and the database tables can be roughly divided into two categories. One category is the core framework. The core framework allows for the basic functionalities for this online discussion board, and is expected to be reused without modifications for any other subsequent courses that might come in the following years. Basic functionalities include user authentication and initialization, displaying the general 15.351 Innovation and Entrepreneurship home page, basic design of class scheduling, the Phorum list of forums, and listings of individual threads. The other category is the Phorum-specific files. The code and the database tables in this category are specific to the given year and class roster. This may include any hard-coded exception handling, all the Phorum messages and class rosters, and the General Syllabus for the course. This part of the code is expected to be Phorum- and year-specific, and is not expected to be reused without modification.

## 2.4 Issues

While the design of the bulletin board system is very straight forward, there were a few important architectural issues worth mentioning. Nearly every part of the Phorum system interacted with the MySQL databases, and data manipulation was performed using PHP scripts. However, there were problems and concerns raised in previous versions of the Phorum 15.351 Discussion Board that needed to be addressed before development began.

### 2.4.1 User Identification and Uniqueness

The previous implementations of the system used a discussion board that was built into the 15.351 Innovation and Entrepreneurship home page. These implementations required only one table to maintain information regarding personnel, the *students* table. The *students* table stored personal and relevant information for every student in the class, including written Reaction Paper scores, email addresses, login password information, etc. This also meant that only one login verification was necessary, the initial log-in to the 15.351 home page.

However, with the release of Phorum Version 5.0, another login became necessary. Phorum now stands independent of all extraneous systems, and requires its own log-in and password verification and authentication. It would be inconvenient and unnecessary for the students to log into the system twice, so the user identification tables, while in two separate tables, needed to be merged in the code.

Person ID	Full Name	Username	Password	Email	etc...
1	Bob Jones	jones	ej3hsk34dn	bob.jones@email.com	...
2	Mary Smith	smith	12sdjek4mp	msmith4@email.com	...
3	James Brown	brown	5da1775365	brownj@email.com	...

Table 2.1: The general *students* table for access to the 15.351 home page

The *phorum.users* table is very similar to the *students* table, but has key differences in the way passwords and usernames are stored. To create familiarity and comfort among the users, the usernames in Phorum were a combination of the first

and last names; i.e. James Brown's username for the home page was simply *brown*, but in Phorum it became *James\_Brown*. With a first and last name to associate with a user, the discussion board process became less technical and more personable. Additionally, the passwords in the Phorum subsystems were stored as a hash value of the actual password to protect the privacy of the students.

Person ID	Username	Password	etc...
1	Bob_Jones	5327b04a3b1a0e0a10f073af32b9a470	...
2	Mary_Smith	464ed654360f391381caae5c5b5ac004	...
3	James_Brown	eb044d54c7cf27fee9683cf21f9f8312	...

Table 2.2: The *phorum\_users* table for access onto the Phorum Discussion Board

Finally, once the two tables were synchronized, the remaining problem was automatically signing into the Phorum subsystem once logged into the general website. The script *iande.php* would run upon entering any Phorum-related page, and would manually change the global session variables to signify the user as logged in. These global variables were passed from the previous page by variable values appended to the end of the HTML tags.

## 2.4.2 Modularity

Modularity of the system was the most important design decision; it was critical for future maintenance and expansion. Specifically, since the course is taught every year, it was important to design the Phorum subsystem so that the possibility of modifying, removing, or adding new information each year would be possible without much additional work.

The *iande* database is a year-sensitive database, holding information pertinent to the current year's version of the course. This database includes the schedule of all lectures, lecture topics, assigned readings, and due dates. Making this database link directly to Phorum allows changes to be easily made when the Syllabus and course schedule are changed in subsequent years.

For ease of future use, there needed to be only one instance of Phorum, despite the presence of two separate sections. To solve this problem, the *students* table

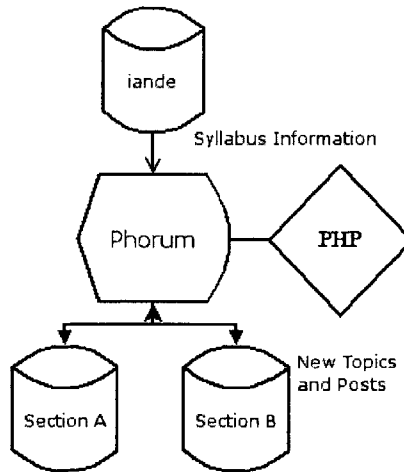


Figure 2-3: Architertural Design of Phorum

was accessed every time the system visited a new page within Phorum. The system would look-up the correct database to access, based on the database the username was located in. Every instantiation of Phorum would run a real-time test to determine the proper database, and dynamically change the database connected to Phorum.

This was crucial to the design and development of the Phorum subsystem, since it eliminated the redundancy of having two separate instances of Phorum being run for every section. Moreover, any code changes made to the Phorum PHP code would only need to be saved once, and not saved for each copy for every individual section. In future years, when classes may have many more sections, only one running instance of Phorum code is necessary for a stable and reliable discussion board subsystem.

## 2.5 User Experience

One of the main objectives and goals of the Phorum subsystem was to create a positive user experience. It was important that students could rely on Phorum and trust the system would ensure good discussion and online posting. Additionally, there were many features that were added to the system to enhance user-friendliness.



### 2.5.1 Biographical Pages

One of the first features added to Phorum was the biographical information page. This was added to Phorum at the beginning of the term, and was necessary at the onset of the semester so students could enter information about their hobbies and interests. Each time their name appeared on the Phorum subsystem, through a post or a reply, their name would be a hyperlink to their own personal bio page. The bio page would have personal information on the user, including a profile picture, email address, work experience, extracurricular activities, and hobbies.

A biographical information sheet was handed out to students on the first day of class, listing the above categories, and students were asked to fill in their personal information by hand. All bios were then written to the *students* table of each database, which required much time and effort. If the users clicked on their own bio pages later on in the semester, there would be an option to update or add new personal information. Any updated information would get immediately written to the *students* table, and all updates would be viewable by the rest of the users instantaneously. The information was added to the system manually at the beginning of the term was for user convenience. I figured that this feature would be used much more often if the users did not have to output any effort to enjoy this element, and any unfilled information bios would be much more likely to be completed if the majority of the bios were already complete.

### 2.5.2 View All Threads

Another aspect of the Phorum discussion board is a “View All” Link seen at the header of each page within Phorum. The “View All” link, when accessed, links to `all.php`, a listing of all the threads in the current forum displayed in a “flat” mode. “Flat” mode means that all the threads, with their subsequent replies, are listed on a single page together. This feature was added onto Phorum to make it easier for the students to read all the topic threads at once. In addition, it helped students navigate the entire forum from a different perspective, allowing users to compare topic threads

and responses.

### **2.5.3 Class Lists**

Another important design aspect includes the involvement of all the students in a class bio page. This page, accessible from the header of every page within the Phorum subsystem, shows the entire roster of the class of students in a matrix format. Each row contains the information of one student, and each column denotes a hobby, interest, or other piece of personal information regarding a student. This feature was added to create familiarity amongst the users and to generate a strong rapport with one another. For students that were not friends with any of the other students, they could use this feature to learn more about the other members of the class. This is very similar to the biographical pages, except the class list is a listing of all the members in the class, and not just a single user.

### **2.5.4 Statistics Page**

With most online systems, there is always the fear that any information filled is not being stored properly. For this, we have the statistics page, found as a hyperlink on the footer (from footer.php) of every page within Phorum. The user can review their current number of new topics and replies for each forum/reading assignment and check for the number of other students who have replied to their topics, denoted as “Hits”. Since this is a page viewable from a web browser, it is printer-friendly. The bottom of the page also lists scores the students have received for their writing assignments, or Reaction Papers.

# Chapter 3

## Administration and Maintenance

### 3.1 Overview

Immediately after launch, I was the sole developer, and was responsible for any subsequent data changes, bug fixes, and feature and module additions. While I wrote several makeshift scripts to simplify the process of modifying or fixing the data, much of the work was done directly to the database using phpMyAdmin, a database management tool. This approach was very efficient, since posts written to Phorum were automatically written to the database, and definitely maintainable in the long run. With this discussion board, the initial goal was to achieve reliability and consistency, and any anomalous changes had to be changed in the database manually, so Phorum could continue running smoothly.

Initially, I knew very little of what I would need for an Administrative Subsystem. I had some guesses as to what kind of functionalities would be necessary, but not a complete picture. Fortunately, my advisor guided me well, and the Administrative Subsystem was relatively simple to learn, especially because it ran concurrent with the Phorum subsystem.

I decided that the two most important features of the Administrative Subsystem were going to be fast development time and a common framework that was flexible enough that I could add any kind of administrative modules that the system might need later.

## 3.2 Design

Compared to the Phorum subsystem, the Administrative Subsystem has a very minimal framework. Since the admin pages had less interface requirements compared to the Phorum subsystem, the structure could be simpler and less focused on aesthetics and more on functionality. Since the Admin pages were run off of phpMyAdmin, the general user interface was already accessible via any web browser. Additionally, any manual management of the site could be accessed via any client computer by logging in as an administrator.

The following are some of the administrative tools that were used to oversee the system. They were all useful in overlooking general day-to-day operations and visualizing interaction among users.

- **Ranking** Administrative tool breaks down total posts, replies, and hits for all students. This also shows posts, replies, and hits broken down into both halves of the course, which is useful for the Experimentation section in Chapter 4. Finally, this lists tables ranking the students by order of hits received, broken down for the entire semester, the first half, and the second half of the course.
- **Stats All** Statistical tool that lists all students in both sections, as well as the number of posts, replies, and hits received for each forum, an aggregate sum for the entire semester, and written Reaction Paper scores.
- **Entropy** Shows an array of each student and the number of incoming hits and outgoing replies over the course of the entire semester, and broken down into two halves. This is useful for comparing the diversity of hits received and given, and is also used to compute entropy levels for every student.
- **T-Stats Formatting** Once entropy levels were computed, this administrative tool compared entropy levels across sections. This also output an array listing the number of hits received each student.

## 3.3 Maintenance Issues

One major challenge of online systems is the need for constant maintenance. By maintenance, I mean both the fixes necessary for the continued execution of the system, as well as supporting the current users of the system. Once again, since it was high priority to ensure user happiness and smooth interface interaction, quick and up-to-date maintenance checks were necessary every day.

### 3.3.1 Concurrent Data Entry

A potential pitfall testing the robustness of the Phorum discussion board was the issue of double posting in the system. This presented a major challenge, because students were restricted from posting more than one new topic and one reply for every reading (in its respective forum). The fix made on Phorum that denied users posting and replying more than once was a hard-code into the system's HTML code that did not show the HTML "new topic" or "reply box" once the quota had been reached.

However, there were instances when the users happened to post the same topic/reply twice. This generally occurred when the users were saddled with poor connections on their home terminals, and often hit the "Post" button twice. It also happened to occur whenever students would successfully post once, for some reason hit the "Back" button on their browser and return to the previous screen, and post again. Fortunately, it was quite easy to detect these redundant posts, since their *message\_ids* differed by only one number, and the content found in the body of their posts was exactly the same. I ran a simple PHP script through the *phorum\_messages* table to delete redundant posts.

### 3.3.2 Antivirus Protection Problems

A few students also complained about inconsistencies with their reply box not appearing whenever they were ready to respond to another user's post. A handful of users shared repeated instances of inability to reply, while the rest of the users re-

ported no such problems. It immediately occurred to me that I had reached a web browser-dependent problem. Simple testing on the server-side showed no problems with reply boxes, so the problems were undoubtedly on the client side.

Debugging this maintenance issue became an individual client-specific case. Some users reported success once they had shut down their Anti-Virus Protection Programs, while others continually experienced failure. Exception handling became user-specific, so I ran troubleshooting tests using the `exception.php` script. My advisor and I determined that the failure was being caused by a malicious variable name, so our goal was to pinpoint the specific variable(s) causing mischief. For each user experiencing difficulty, I created a troubleshooting report, echoing out all pertinent variables on a user's system at the time of the reply box failure. Then, asking the user to send the variables to me via a feedback report, it was easy to compare the user's variables with the server's *correct* variables to find the erroneous settings.

I determined that an `HTTP_REFERER` variable was causing all of the problems. The issue occurred with internet browsers that did not store referral variables from previous pages, so the value being sent each time was *null*. Once the problem was identified, it was fixed rather quickly. Phorum ran smoothly without any hiccups for the remainder of the term.

### **3.3.3 Customer Service**

Throughout the course of the entire semester, there were a lot of emails sent to Professor Cummings and myself. The feedback was initially directed at my advisor during class, but became more email and web-oriented as the semester progressed. While reading most of these emails and responding would have been draining and time-consuming, I was fortunate that most of them were filtered before they reached me. I think this system worked quite well, since I already had enough bugs and maintenance to keep me busy for the majority of the term. The extra emails and feedback criticisms would have taken up a large amount of my time.

A lot of my time was spent fixing problems with posting and replying. While the system generally had no problems at all with the system specifications, the majority

of the problems dealt with users who missed posting a message within its certain time period. Fixing these user issues required signing in as the particular user, circumventing the hardcoded fixes restricting posting time periods, and adding these posts to the database. It was also sometimes approached by making manual changes to the database's *phorum\_messages* table (See Table B.1 for more information).

### 3.3.4 Concurrent Code Fixes

Once the system had launched, the task of fixing bugs became more constrained. I could no longer change the structure of the database tables, since the users had already begun entering data into current formats via posts and replies. I also had to be more careful with my fixes, since any new bugs introduced will directly hurt users and result in a flood of complaint emails. The most important aspect to the entire system remained user happiness and system reliability.

Fortunately, my project already had an infrastructure in place to handle much of this problem. The project had a three-tiered development environment. The first tier was running Phorum on my localhost, which also had the Apache Web Server and MySQL databases running locally. It was very simple to play around with the code and data without any possibility of data corruption or system failure. The second tier was a secondary server entitled *netvis* ([netvis.mit.edu](http://netvis.mit.edu)) which served as a staging area. On *netvis*, any code changes I made could be tested and retested before being introduced to the public. Every night at midnight, the SQL databases from the *idi* server (for the actual website) was saved and backed up onto the *netvis* server. These daily database updates made it easier to test any changes or new module add-ons. Before unleashing any changes to the students, Professor Cummings and I would run several tests on this server to ensure correctness. The final tier was the actual *idi* server on which the course website ran for its users. *idi* is always running an accurate and correct version, and it is the server the users with which the students interacted.

The three-tiered system worked quite well. There were some instances where I was forced to fix bugs directly to *idi* due to time constraints, but fortunately these instances were for relatively simple bugs.





# Chapter 4

## Learning

Along with the system development and administration, I was eager to utilize this discussion board to track trends and patterns amongst the students. Professor Cummings and I viewed the Phorum discussion board as a tool that could be used to analyze how students interact with one another. We were also eager to correlate online participation scores with in-class participation scores to see how the two were related to one another.

### 4.1 Experimentation

#### 4.1.1 Setup

Using our partially developed discussion board, a main goal was to use this board to analyze interaction amongst the students.

The basis of our experiment required a control group and an experimental group. We chose Section A to be our experimental group and Section B as our control group. However, they did not know that the anonymous feature was being used in an experiment.

The basis of our experiment involved a change in Phorum. In Chapter 2, we mentioned that students were required to post a new topic and reply to an existing topic (without replying to their own topic) for every lecture and the pair of assigned

papers that accompany the lecture. Users were only allowed *one* reply per session. Moreover, students' online participation grade directly correlated to the number of replies their own topic garnered. It would therefore make sense for students to begin discussions and topic threads that are the most engaging and thought-provoking, and would create more discussion via other replies.

Halfway through the semester for the experimental group, it was announced that the students in Section A would no longer be able to view the authors of threads until either:

- The student has already replied to a topic for that particular reading
- The window for replying for that reading has passed

However, the students in the control group were not informed of the difference in Phorum. Everything on the discussion board continued to operate the same way for the control group. The reasoning behind this change was to test how much students would base their replies on people rather than content. With anonymity, students would (theoretically) no longer be able to reply to threads based on familiarity with the authors, but that a true egalitarian environment would be created where the quality and intrigue of a post drives the number of posts to a higher number.

	First Half	Second Half
Section A	+	-
Section B	+	+

Figure 4-1: Experimental Setup

The semester included 8 lectures of posting, so each semester was divided into halves of 4 lectures each. Since each lecture had two accompanying assigned readings, there were eight forums (or readings) belonging to each half. Based on the above

diagram, the control group, or Section B, would continue posting as normal once the half-way point was reached. However, the experimental group, Section A, was approached with the new anonymous posting regulations for the second half of the semester. It was designed this way so that this second semester of change could be compared against the normal, controlled data from the other three possible quadrants of data.

### 4.1.2 Results

The results were tallied by running a PHP script that analyzed the data to find how users would give out replies and receive hits from the other students. For each student in the class, I determined the diversity of people he/she replied to and the diversity of people that replied to her posts (the people she received a “hit” from).

Using these arrays, the structural diversity in these posts and replies were measured by calculating their entropy. The entropy-based index I used to calculate such measures is defined by the equation:

$$H = - \sum_{i=1}^s P_i (\ln P_i). \quad (4.1)$$

where there are  $N$  possible states  $\{s\}$  in which a system can be, {and}  $P_i$  is the probability that the system will be found in state  $i$ , and  $H$  is the heterogeneity of the system.

The total entropy results were tallied for students in both classes, with each student receiving an entropy score for:

- Incoming Entropy (Hits Received) for the Entire Semester
- Incoming Entropy for the First Half
- Incoming Entropy for the Second Half
- Outgoing Entropy (Replies Made) for the Entire Semester
- Outgoing Entropy for the First Half

- Outgoing Entropy for the Second Half

The results can be found in Tables C.1 and C.2 of Appendix C.

However, the crux of this experiment dealt with comparing the difference in Section A before and after the system change, with Section B as the control group. Additional statistical tests were performed, using t-test calculations to determine if any of the entropy values are statistically significant when compared to other groups:

	T-score	Significance
Overall	.312	.756
Group 1	-.017	.987
Group 2	.417	.678
Section A, Group 1	.649	.523
Section A, Group 2	2.123	.044
Section B, Group 1	-.893	.381
Section B, Group 2	-.951	.351

Table 4.1: Tables in the Database Used by 15.351

\*Group 1- Outgoing Entropy/Replies Given to other Users  
 \*\*Group 2- Incoming Entropy/Hits Received from other Users

Using the t-test, a data set is statistical different from other sets if the Significance value is lower than .05. Using this standard, the only data set that showed statistical significance is found in Section A for Incoming Entropy, or Hits Received.

A closer look at the average values in Incoming Entropy across all four quadrants of data resulted in these values:

	First Half	Second Half
Section A	1.75	1.57
Section B	1.62	1.66

Figure 4-2: Experimental Results for Mean Values on Incoming Entropy

There is a large drop in incoming entropy from the first half to the second. In other

words, after the system changed to anonymous posting, the diversity of hits received in Section A decreased dramatically AFTER the system change was implemented. I investigated further by examining the people from whom every user received a hit. In Section A, there is a startling drop in the diversity of people who respond to every user. For example, whereas a user might have received 10 hits from 10 different people in the first half of the class, they might receive only 6 hits from 3 members of the class. The additional hits were being redistributed to the people with higher hit totals, but also receiving multiple hits from the same person.

This supports my hypothesis stated in Chapter 1. After the anonymity was achieved, users no longer received hits due to the person behind the name, but they received hits based on the quality of their posts. In the first half of Section A and during the entire semester for Section B, the system encountered higher entropy values. One possible explanation for this phenomenon is that personal relationships among friends created biased reply totals. When the names were hidden, an unbiased system of equality and fairness was created, with the entropy values decreasing to a lower level.

## **4.2 Trends and Patterns**

In Chapter 1, I described how the class participation grade for the course depended on two components: in-class participation and online learning. The reasoning behind the partitioning of the grading was the belief that students had their own preference for student interaction, whether it be personal or online. As a result, here are some of the results and trends that can be compared when analyzing participation in the course.

### **4.2.1 In-Class Participation Scores**

The in-class participation scores were determined based on classroom discussion and insight. Attending class is not the only requirement for scoring high on this portion of the class; students are expected to come to class prepared and to actively participate

in discussions. Professor Cummings determined the in-class participation scores based on involvement in classroom discussions. Scores were assigned on a range of 0-100, with 100 being the highest score attainable.

#### 4.2.2 Online Participation Scores

Online participation scores were determined using a formula, and calculated after the in-class participation scores had already been assigned. In-class scores were determined first, to minimize the chance of bias based on the online score. Since the in-class score was very subjective, it was important that there were no external factors influencing grades. On the other hand, online participation scores were determined using a strict formula that incorporated a combination of things:

- Posting the requisite number of new topics and replies
- The number of hits garnered from other students in the class

Using these items, the online participation score ( $\sigma$ ) was determined using this formula, with the score being based on a 1-100 scale, with a score of 100 being the maximum:

$$\sigma = 92 - \eta + \alpha, \tag{4.2}$$

where

$$\begin{aligned} \eta &= \text{the number of total posts/replies below 30} \\ \alpha &= \frac{\text{total hits}}{8} \end{aligned}$$

In other words, students were given an initial default value of 92 for online participation, and they had one point deducted for every missing post/reply below the required 30, and they received an additional  $\frac{1}{8}$  point for any hits they may have received. Online participation scores for each half of the semester were also calculated the same way, with the required post/reply set at 15, and with an increase of  $\frac{1}{4}$  point for any

hit received. Students, however, were only assigned an online participation grade for the entire semester. Calculating online participation grades for each half were strictly for correlation and statistical analysis purposes.

(To see the online and Class Participation Scores, see Tables C.3 and C.4)

### 4.2.3 Correlations

An interesting aspect of the course we wished to trend is how the online score and in-class participation score compare to one another. Trends in learning have evolved with the advent of the World Wide Web, and along with its prevalence comes new ways to express learning and reinforcement. Correlation, also called the correlation coefficient, is a numeric measure of the strength of linear relationship between two variables (online and in-class participation). It is found by dividing their covariance by the product of their standard deviations.

	Class	Online	<i>Online</i> <sub>1</sub>	<i>Online</i> <sub>2</sub>
Class	1	-.126	-.113	-.150
Online		1	.906**	.708**
<i>Online</i> <sub>1</sub>			1	.385*
<i>Online</i> <sub>2</sub>				1

Table 4.2: Section A: Correlations in Class and Online Participation

\*\* . Correlation is significant at the 0.01 level (2-tailed).

\* . Correlation is significant at the 0.05 level (2-tailed).

	Class	Online	<i>Online</i> <sub>1</sub>	<i>Online</i> <sub>2</sub>
Class	1	.155	.083	.287
Online		1	.765**	.776**
<i>Online</i> <sub>1</sub>			1	.258
<i>Online</i> <sub>2</sub>				1

Table 4.3: Section B: Correlations in Class and Online Participation

\*\* . Correlation is significant at the 0.01 level (2-tailed).

Shown above are correlation tables when the Class Participation Grade and Online Participation grade were compared for both sections. In addition, tests on the online

participation for the separate halves of each course were also compared. While I did not expect to see any major change for across sections for either course, I was interested to see how the anonymity change would affect Section A when compared to Section B.

According to the correlation tables, there is strong correlation across both sections comparing the overall online participation grades with either half, but this was expected. However, the most enlightening data was the non-significant correlation coefficients between the online and class participation grades. The values of  $-.126$  and  $.155$ , respectively, show that the the grades have no correlation at all. In other words, a high score for one value does not guarantee a high score for the other, and vice versa. The evidence from the participation scores show that the two types of learning, in-class and online, are statistically unrelated. This reinforces my hypothesis that learning can come in all types of packages, and with the growth of the Internet and other new technologies, many of these new learning techniques should be pursued.



# Chapter 5

## Conclusion

### 5.1 Progress Update

I signed onto this project in January 2005, and began immediate work with database setup and administration. My learning period for the administrative database and PHP code required a quick grasp of the system intricacies; the first prototype needed to be ready for launch by the first week of February 2005, for the beginning of the Spring 2005 semester. Moreover, the initial launch needed to have accurate and consistent functionality, so the testing period was very important. It was an opportunity to look over the entire web site with the Phorum subsystem, and to test all login capabilities, posting regulations, and all reply scenarios.

The system, with bare-bones functionality (but consistent and accurate posting and replying functionality), was released to the Professor Cummings 15.351 class in early February. Students began using the board to post thoughts and ideas almost immediately and populating the databases with data. Once basic functionality was working, I began working on additional features to the system that would enhance the user experience. During this period, I also discovered the pressing need for solid administrative management and practices to ensure data accuracy and system reliability.

The month of March was hectic due to many minor bug fixes and feature additions that required a prompt turn-around time. The next few weeks brought about many

of the user features listed in Section 2.5. Towards the end of March, however, the bug reports subsided and the features were completely integrated into the system, and by the beginning of April, the experimentation change of adding anonymity to Section A users was completed.

Currently, the system has finished its discussion board role for the semester, with a total of 36 forums across two sections, and a total of 1680 posts made by 52 registered and active users.

## **5.2 Lessons**

### **5.2.1 Ramp-up Period**

As advertised, PHP is a very simple language to learn. The syntax is straight forward, the library of functions is complete, and there is an abundant amount of tutorials and documentation on the Internet. There is, however, still a ramp-up period that is required to use the language effectively. Because it is a scripting language, PHP is different from Java or C++ because it does not follow the design patterns and engineering practices common in compiled languages. Using a scripting language was favorable over object oriented programming for this project because most of the occurrences were actions and reactions between the client and server. Very little state was maintained, except for the data in the database. Furthermore, writing PHP code will nearly always require the knowledge of Javascript and HTML. While the three languages are quite simple to learn, combining them into a functional executable is not always trivial. Differences in standards and implementations could lead to mysterious problems and errors. Using PHP for this system gave me a lot of flexibility in deciding the flow of control or the behavior of the user interface, compared to writing a user application in Java. Several iterations were required before I created a PHP framework that worked around the weaknesses of the language and worked to its strengths.

## 5.2.2 Standards and Implementations

One problem I did not foresee was difficulty caused by all the different browser and platform combinations. One problem we had concerned the antivirus detection program that many of the students used, which would not accept HTTP referrals as variables. In addition, there were different web browsers that did not accept cookies for any sessions, so I decided avoiding headache and using global session variables instead. These kinds of incompatibilities make the system look very unprofessional, while resulting in frustration and sometimes anger by its users. What I should have done was decide beforehand on browser and platform combinations to support. Future changes should also include creating other testing and debugging environments to reduce the likelihood that other incompatibilities (like the antivirus protection bug) will occur.

## 5.2.3 Experimentation

The experiments conducted in Chapter 4 resulted in two statistical pieces of evidence. The first piece of evidence showed a decrease in incoming entropy of received hits when anonymity was introduced into the system. The second piece of evidence showed there was no correlation between online learning and in-class participation.

Introducing anonymity into the Phorum subsystem resulted in lower amounts of entropy for incoming replies. When the affected data from the second half-semester of Section A was compared to Section B's data from the same period, a statistical t-test showed the drop in entropy was statistically significant, and not simply a matter of negative variance. The change allowing anonymity in usernames and posting diminished the possibility that online participation is influenced by anything other than the content of posts.

Comparing online participation scores with in-class participation scores showed no correlation between the two variables. Correlation coefficients relatively close to 0 (actual values were  $-.126$  and  $.155$ ) showed that class participation and online participation were statistically unrelated. This supports my belief that students all

have different mediums with which they may excel academically, and more of these mediums should be utilized within the classroom.

## **5.3 Future Work**

Because of the yearly recurrence of the class, there is plenty of future work to be done, with new versions of the course every year. This section will describe a snapshot of things that must be done to the system. Generally speaking, future work can be divided into three categories. The first category is concerned with the underlying structure of the system. Changes here will not be seen by the users, but will help future development and improvements. The second category includes major features and modules that can be added in anticipation of future students and user friendliness. The final category is major overhauls that may be done with newer releases of the Phorum discussion board template.

### **5.3.1 Modularity and Cleanup**

As mentioned earlier, modularity was one of the key principles when the design of the system was outlined. However, in order to attend to some last-minute and time-sensitive administrative controls, some hard-coding was included, and to some degree modularity was lost. As a result, adding or deleting messages became a very manual and belaboring process that required a very thorough knowledge of the system. Furthermore, there were risks that a change in one part of the system could have affected the behavior of other parts of the system.

Much of these interdependencies could be cleaned up. One potential future goal of the project is to make the system modular enough so that other organizations or classes could customize the framework and run very similar discussion boards. The system should be edited even more to provide better administrative tools. While the core framework is modular (see Figure 2.3) enough to reuse year after year, there are still other small areas of code that could use cleanup to create a more modular design.

### **5.3.2 New Features and Modules**

The next time Phorum is expected to be used is Spring 2006. Based on feedback from this year, there are some features that could be added that would improve the system.

One feature that would be crucial for the users is a Search feature. This feature would scan all the existing messages in the database for a specific user-entered query and return all matches. This would be greatly helpful to the students as they are reviewing past forums and looking for specific types of posts and replies.

In the administrative side, the system needs more tools that will allow for better administration and overview. Added features may include checks for redundant posts, automation that allows administrators to post messages outside of the designated time frame, and dynamic statistical tests checking entropy and diversity of posting for general feedback.

### **5.3.3 New Version Releases of Phorum**

Finally, major overhauls may be needed for the core framework of the system in future years. Phorum.org releases new versions of their software templates (with new features and add-ons) every few years, and depending on the importance and need for the newer versions, it may be necessary to redesign the system with the newer HTML templates in mind. However, with the knowledge of PHP and MySQL, as well as the robust structure of the current system, new version releases of the Phorum discussion board for 15.351 may prove to be beneficial and relatively easy to implement.

## **5.4 Contributions**

This thesis has achieved three goals. The first goal was to develop a robust and modular framework that can be used for online discussion boards, used specifically in 15.351. The current system is modular enough that it may be reused year after year by only changing the course-specific data. In addition, I have added features that

are user-friendly and accomodating. Future work may include the addition of new features and packaging the system into a portable system that can be used by other classes.

The second goal achieved involved learning about online interaction among students. When presented with a system in which the users depend on one another for a good grade, the students might base outgoing hits on user familiarity and name recognition. However, when the system no longer accomodated this recognition, the system was marked with objectivity and equality.

The last goal investigated the correlation between online and in-class participation. This thesis found zero correlation between the two, signifying that the two forms of learning are independent of one another. If learning techniques in higher education are to be improved in the future, universities and institutions should expand their horizons by using more techniques that can captivate and fascinate the mind from many perspectives.

## Bibliography

- [1] The Apache HTTP Server Project. <http://httpd.apache.org/>.
  
- [2] Innovation and Entrepreneurship. <http://iande.org/>.
  
- [3] Mysql: The World's Most Popular Open Source Database. <http://www.mysql.com/>.
  
- [4] Phorum: Forum, Message Board, Whatever... we call it Phorum, a PHP/MySQL open source application. <http://phorum.org/>.
  
- [5] PHP: Hypertext Preprocessor. <http://php.net/>.
  
- [6] phpMyAdmin: MySQL Database Administration Tool. <http://www.phpmyadmin.net/>.
  
- [7] Carlson, Scott; New Allies in the Fight Against Research by Googling; The Chronicle of Higher Education (March 21, 2003); <http://chronicle.com/free/v49/i28/28a03301.htm>
  
- [8] Drucker, Peter; Chronicle of Higher Education, 43(32); 1997, April 14; p. B7.
  
- [9] Jackson, Gregory A.; "Evaluating Learning Technology: Methods, Strategies, and Examples in Higher Education;" The Journal of Higher Education, Vol. 61 (May-June 1990): 294-311.
  
- [10] Paulson, Karen; Reconfiguring Faculty Roles for Virtual Settings; The Journal of Higher Education, Vol. 73(1): 123-140.
  
- [11] Privateer, Paul; Academic technology and the future of Higher Education; The Journal of Higher Education, Vol. 70, No.1.





# Appendix A

## Screen Shots

Appendix A will consist of a series of screen shots of Phorum in action. The following pages will illustrate the general process a user may go through to navigate and post successfully within the pages of Phorum. In addition, it will illustrate some of the key concepts and features that were added to the system to promote user friendliness and accessibility.

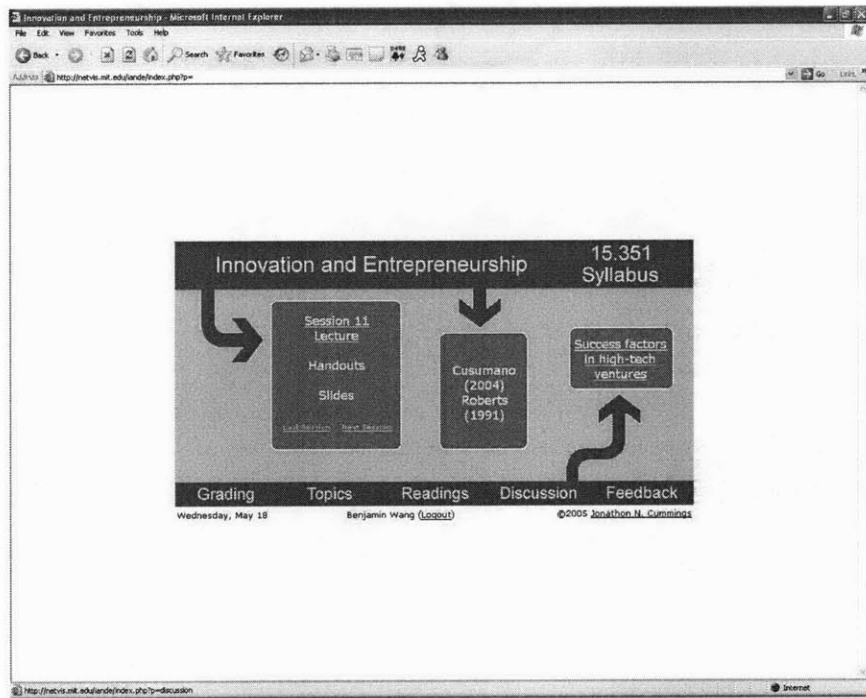


Figure A-1: A user is currently logged into the homepage.

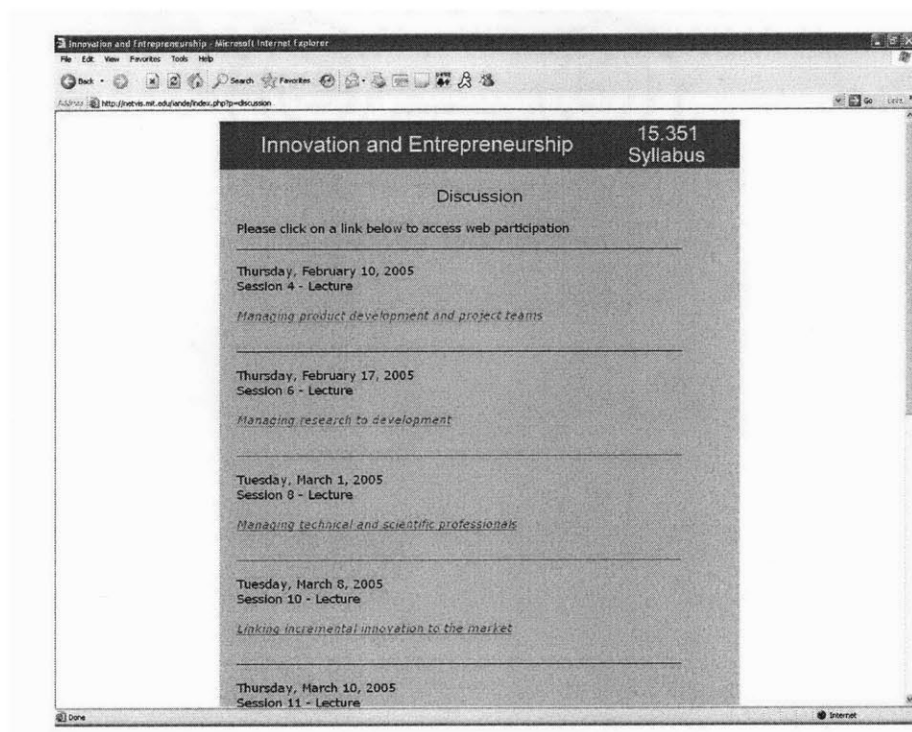


Figure A-2: The discussion page shows all active and previous forums via URL links.

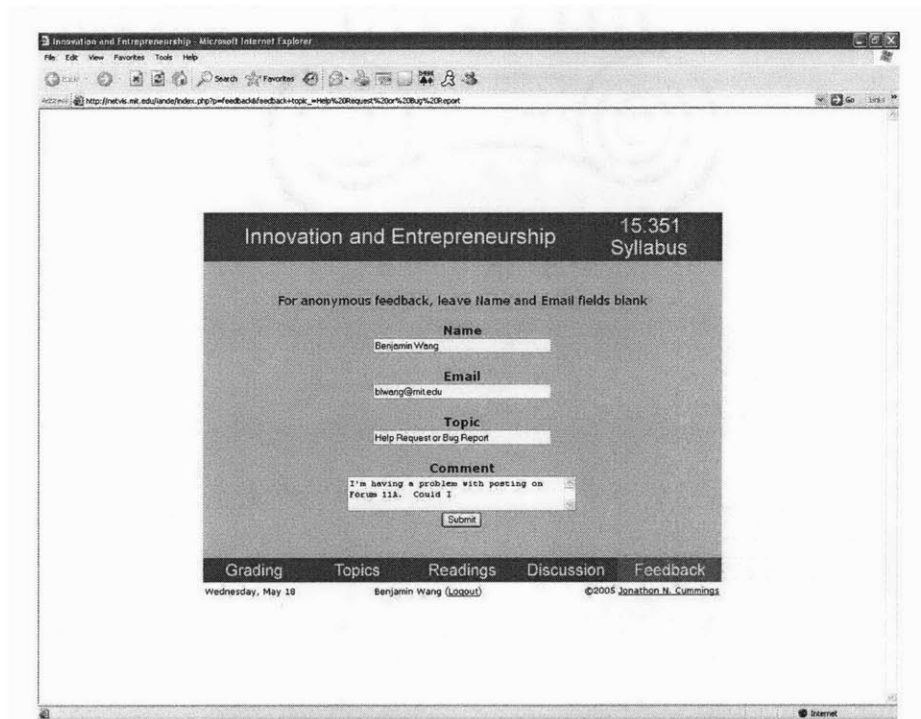


Figure A-3: Accessing the feedback page is possible from the main website and from Phorum. Here a user is emailing the administrators about a posting problem he is having.

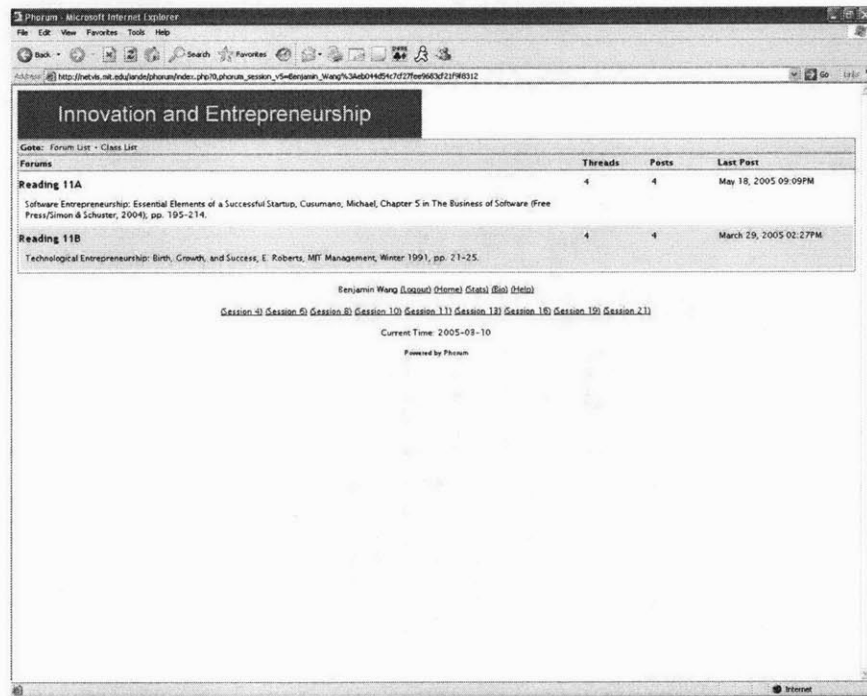


Figure A-4: A user has just entered Phorum. He is looking at the two assigned readings and forums for Lecture 11.

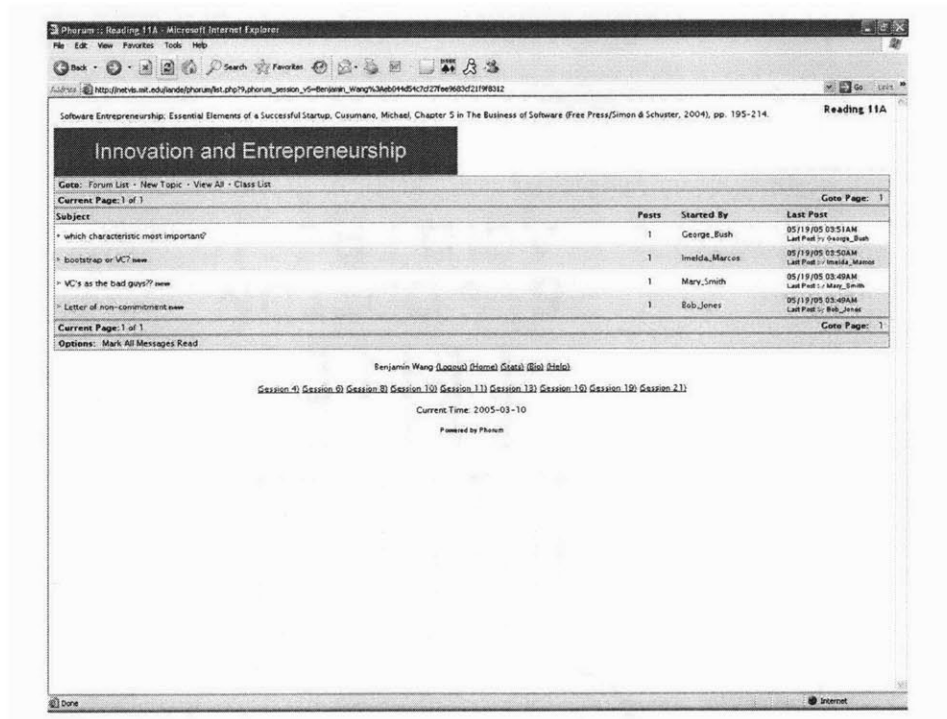


Figure A-5: Upon entering Forum 11A, the user sees a list of available threads.

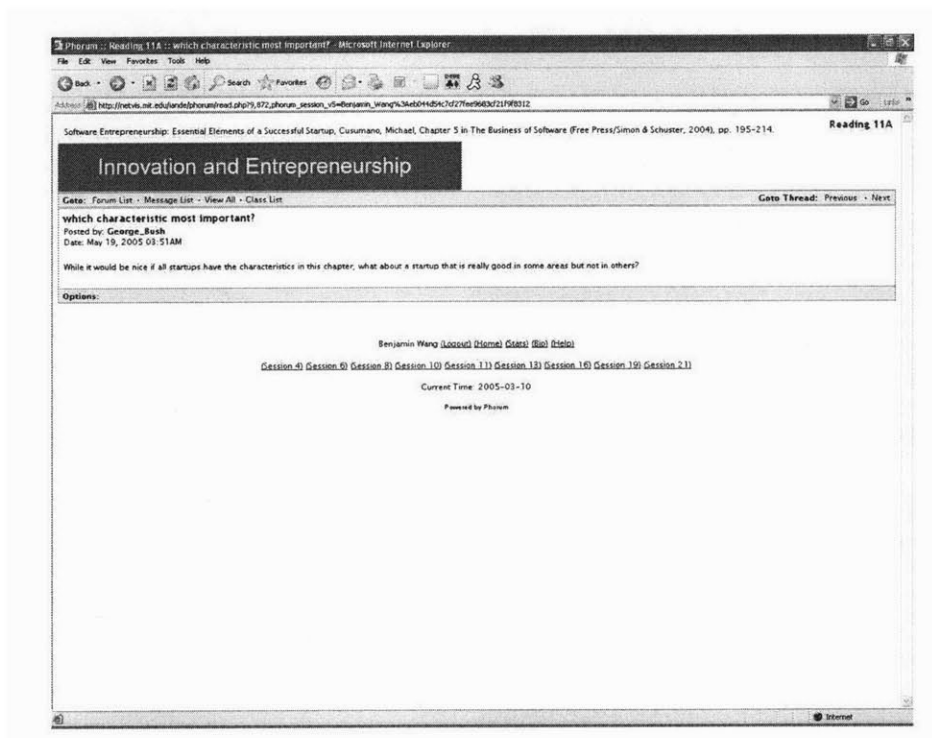


Figure A-6: The user clicks on a thread to view. He has no ability to respond, but the header at the top lists the option to post a new topic.

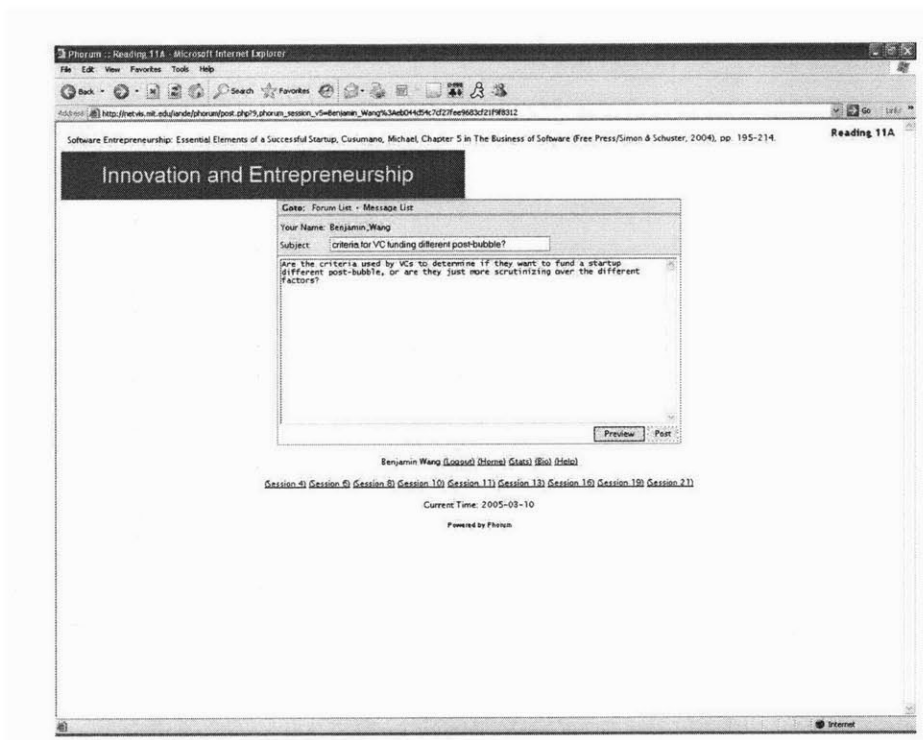


Figure A-7: After clicking on the New Topic link, the user now has the option of posting a new topic.



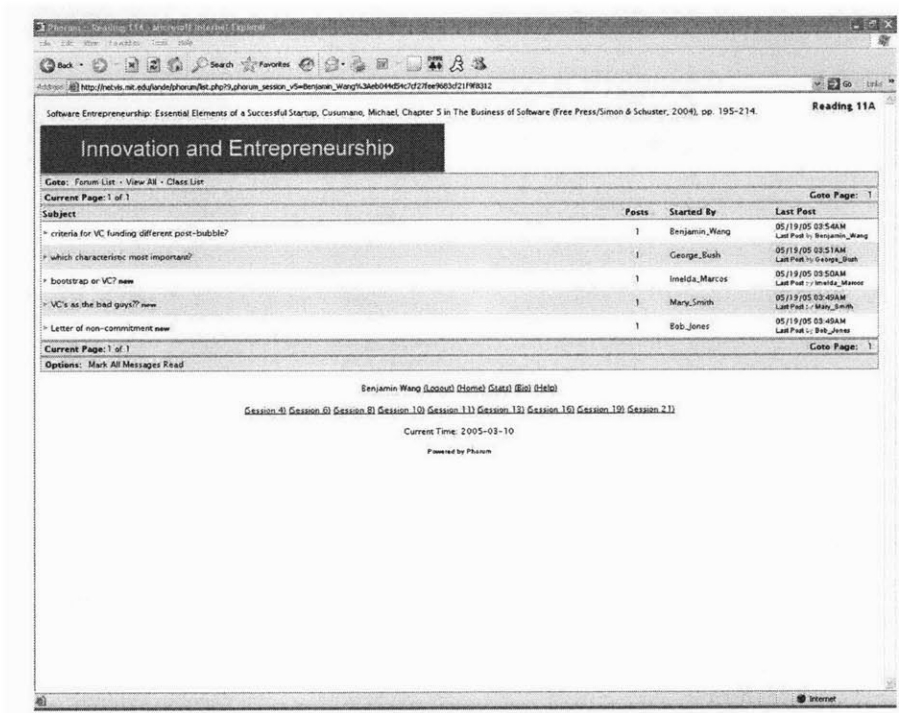


Figure A-8: The user has posted a new topic, and it is now viewable on list.php. The ability to post new topics is no longer available on the header.

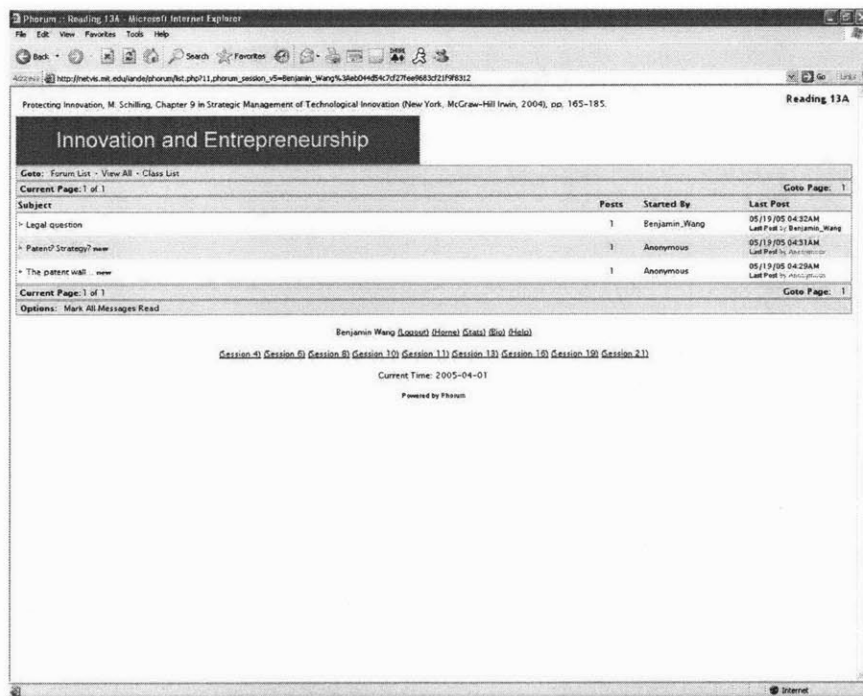


Figure A-9: A few weeks later, the anonymity change has been made. Students can still see the threads, but the other users are anonymous. However, users can still see their own posts.

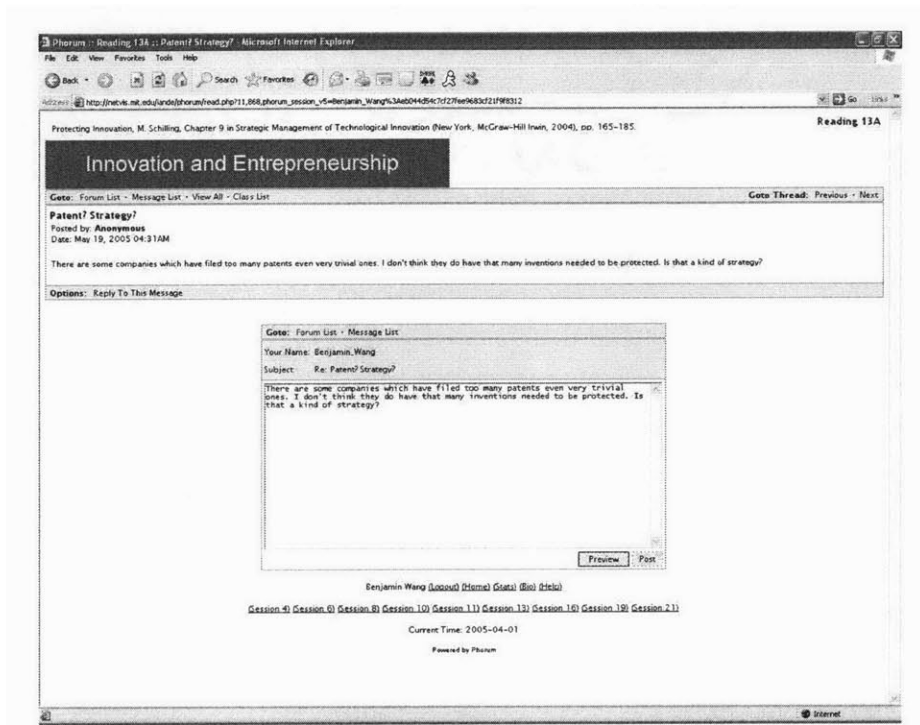


Figure A-10: When reading a thread, the user still cannot ascertain the identity of the other user. However, the correct posting window is open, so the user can respond to the post.

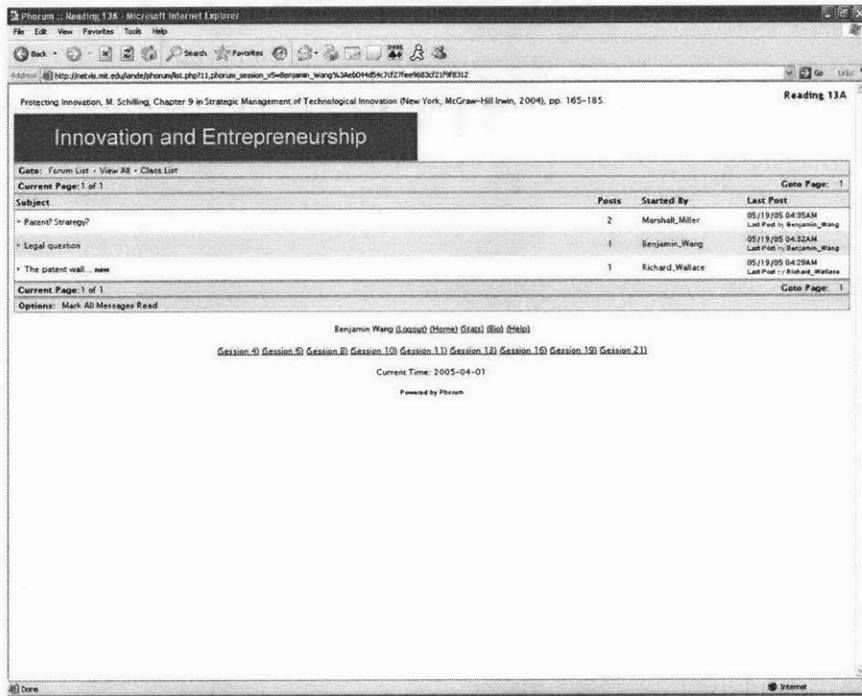


Figure A-11: Upon responding to the post, the user can now view the identity of the authors of all posts.

# Appendix B

## Implementation Details of the Discussion Board

This section will illustrate the details of the online discussion board. First I will describe all the tables in the database, explaining the structure and purpose for each table. I will then describe the groups of code and what they do.

### B.1 Database Structure

Like many online applications, the backbone of the system is the database. The database structure defines much of what can and cannot be done with the system. The database structure and the queries also dominate much of the computation and performance issues. The following is some of the main database tables used in the Phorum System. There were two concurrent databases used to run the online discussion board for the two sections: sectiona2005 and sectionb2005. However, both databases carry the same structure, and are filled with different user-dependent data. Table B.1 shows a complete list of the database tables used in the system. Below is a more in-depth description of some of the more important tables.

- **Students Table** This table is the general backbone for all database activity. This table lists all the class rosters, as well as password login verification for

Table Name	Description
students	Displays the index page for the 15.351 homepage
access	General statistics page for students
phorum_forums	All forums for the system
phorum_messages	Every message ever posted in the online discussion board
phorum_settings	All the basic settings for general daily execution
phorum_user_newflags	Table Listing all new messages and posts for all students
phorum_users	All the users and password verification for the Phorum subsystem

Table B.1: Tables in the Database Used by 15.351

the general 15.351 homepage. The password was taken by using a randomly hash-generated string based on user login names. In addition, extracurricular interests and hobbies are listed, as our scores received for written Reaction Papers.

- **Phorum\_Messages Table** This is a table listing *every* message ever posted in the system. It includes information denoting whether it is a thread beginner, or simply a reply. Additionally, messages are stored based on authors, forum IDs, subjects, and message content.
- **Phorum\_Forums Table** A listing of all forums in the Phorum subsystem. Forums are organized by forum name, ID, and description. Phorum messages belong to one and only one forum.
- **Phorum\_Users Table** Listing of all registered members for access into the Phorum subsystem. This table closely relates to the *students* table, with similar usernames, and identical passwords. This allows the system to circumvent signing into the Phorum bulletin board, and log-in to Phorum and the general 15.351 website all at once.
- **Phorum\_Settings Table** General administrative setup for the entire Phorum Bulletin Board. Many of the notification controls and permissions are set in this table. Additionally, the http\_path is set here to note the HTTP address for Phorum.

## B.2 Code

The PHP code in the system can be roughly divided into framework code and Phorum-specific code. The framework code allows the user to log into the 15.351 website, look at the class schedule, post feedback, and find class materials. The Phorum-specific code handles the actual forums and posts of the system, and this code is dynamically changing based on student interaction. All Phorum-specific code is hierarchically found in the “phorum” subfolder of the 15.351 iande (Innovation and Entrepreneurship) directory.

As a general rule, the HTML generating code and the data handling code is separate. Consequently, a lot of the functionalities have two files, the PHP-based data handling dynamic manipulation code, and the HTML display templates. Since much of the framework of Phorum has been taken from the Phorum template, I don't mention the HTML templates and files here. Following is a list of the important PHP files that merit description, as well as Table B.2, which lists all pertinent PHP files necessary for complete data manipulation and total effect.

- **index.php** This file is essentially the entry point for all users. Apart from having the user type the username and password, this page also initializes all global variables. Furthermore, this is the default page that users get referred back to from the Phorum discussion board.
- **feedback.php** The general feedback page for the course. Students are directed here with any questions they may have with the general website or with the Phorum discussion board. All questions and queries are emailed directly to Professor Jonathon Cummings.
- **phorum/index.php** This file is the general entry point for the Phorum subsystem. Students are referred to this general listing of forums from `index.php`, `discussion.php`, or in a direct link link sent in the email reminder. Based on the session date, this will list reading-specific forums and posts.
- **phorum/list.php** This page will list any contents found within one particular

instance of a forum. Other aspects of this page include showing thread authors, and the last people to respond to a particular thread.

- **phorum/read.php** This file will display the flat version of a single thread. This includes the original thread topic, as well as any subsequent replies that may have resulted.
- **phorum/popup.php** The bio page for any student can be reached by clicking on the hyperlinked name whenever it is listed within Phorum. In addition, if students are logged in and clicks on their own hyperlink, they has the option of changing the information stored for all to see.
- **phorum/stats.php** General statistics page for a user. This displays the number of posts and replies made by the student for the semester, as well as the number of “hits” received. Moreover, posts, replies, and hits are broken down into individual sessions and their readings, divided by forums. Finally, there is also a listing of current class grades, which shows scores received on written Reaction Papers, and class and online participation progress scores.



File Name	Description
index.php	Displays the index page for the 15.351 homepage
topics.php	Schedule of class sessions and links to accompanying Handouts and Slides
discussion.php	Shows hyperlinked URLs to all forums
grading.php	A breakdown of grading specifications and guidelines for the course
feedback.php	General Feedback page for voicing opinions
phorum/index.php	Displays the index page for Phorum
phorum/list.php	Lists all topics under a certain forum
phorum/read.php	Displays the “flat” version of a thread
phorum/iande.php	General Verification and Login page for Phorum
phorum/all.php	“View All” Threads listed in a flat mode for one particular forum
phorum/popup.php	General biographical page, in a pop-up window, for a single student
phorum/bio.php	“Class List” of all students in the course, basic information, and extracurricular interests
phorum/stats.php	General statistics page for students
phorum/exception.php	Exception handling page specific to certain users

Table B.2: PHP Files Used by the System