THE DESIGN, IMPLEMENTATION AND EVALUATION OF CLEW: AN IMPROVED COLLEGIATE DEPARTMENT WEBSITE

A Senior Honor Thesis
Presented to
the Faculty of the Department of Information and Computer Sciences
University of Hawai'i at Manoa

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Computer Science with Honors

by
Aaron Akihisa Kagawa
April 2003

To

My Family

Acknowledgements

There is nothing more important to me than my family. I am eternally gratefully for the strength, support, and above all, the love, that all of you have given to me.

This Undergraduate Honors Thesis would have not been possible without the tutelage of Professor Philip Johnson. Thanks for your encouragement and guidance. I would also like to send my thanks to all the CSDLers. They have provided much needed support and encouragement.

Thanks goes out to Jesse Tom, which is my right hand man in Extreme Programming. Without you, the CLEW System would not have been possible. Also, thanks goes out to the bunch of outstanding programmers that helped me throughout the entire evolution of the CLEW System. Thanks to all of you who put up with my persistence and demands.

I would also like to thank my committee members Professor Dan Suthers and Professor Wes Peterson for taking the time to read and evaluate this thesis.

Last but not least, I thank Jim Caron, Vincent Kelly Pollard, Laura Lyons, and Emmanuel Dreschel for your guidance through the University of Hawaii Honors Program. Obviously, without the Honors Program and caring people like you, I would not have had the opportunity to achieve my full potential at the University of Hawaii.

Abstract

The purpose of a collegiate department website is to provide prospective students, current students, faculty, staff, and other academic and industry professionals with information concerning the department. The information presented on the website should give the user an accurate model of the department, even as it changes overtime. Some of these changes include: adding new faculty members, new students, new courses, etc. The more accurately the website models the department, the more aware the website's users will be of the department.

Traditional collegiate department websites have two primary problems in creating an accurate model of their department. First, only a few people, usually the department webmasters, can add information to the website. Second, it is difficult to enable website users to be informed of changes to the website that might be of interest to them. These two problems decrease the accuracy of the model and hamper its effectiveness in alerting users of changes to the website. As a result, user awareness of the department is also decreased.

The Collaborative Educational Website (CLEW) is a Java web application intended to support accurate modeling of a collegiate department. CLEW is designed to solve the traditional collegiate department website's two main problems. First, it provides interactive services which will allow users to add various kinds of information to the website. Secondly, CLEW addresses the notification problem by providing tailored email notifications of changes to the website.

CLEW was developed by a Software Engineering class in the Information and Computer Science Department at the University of Hawaii at Manoa. My role in this development as project leader is to design and implement the framework for the system. CLEW currently contains approximately 28,000 lines of Java code and it contains upwards of 500 web pages.

In the Spring 2003 semester, CLEW replaced the existing Information and Computer Science Department website. I evaluated CLEW to measure its effectiveness as a model of the department using a pre and post release questionnaire. I also evaluated usage data of the CLEW System to assess the functionality provided by CLEW.

If CLEW provides a more accurate model of a collegiate department, then the next step is to provide the CLEW framework to other collegiate departments worldwide. It is my hope that the users' of CLEW will get a *clue* about their department!

Table of Contents

List of Tables

List of Figures

Chapter 1. Introduction

1.1 The changes in a collegiate department

A typical collegiate department is a large community devoted to education. As with other organizations, collegiate departments are an evolving entity. Within a collegiate department things change on a regular basis.

Changes to the department include the addition of new information and removal of old information. Changes that are additions of new information to the department include: new faculty members, new students, new courses, etc. On the other hand, changes that remove old information include: faculty members that have left the department, students that have graduated, course that have completed, etc.

The following list shows examples of potential collegiate department changes that may occur:

Category of Changes	**Changes (Additions and Removals)**
Students	• (Addition) New students are accepted into the department. • (Removal) Students graduate from the department and are now alumni.
Faculty	• (Addition) New faculty members are accepted into the department. • (Removal) Faculty members leave the department.
Staff	• (Addition) New staff members are hired in the department. • (Removal) Staff members leave the department.
Courses	• (Addition) New courses are offered in the department • (Removal) Courses of the past semester are no longer offered.

Facilities	• (Addition) New facilities are offered to the department members to be used. • (Removal) Old facilities are discontinued
Policies	• (Addition) New policies are introduced. • (Removal) Old policies are removed.
Contact Information	• (Addition and Removal) Contact information of department members (students, alumni, faculty, staff, etc.) change

Table 1- List of potential collegiate departmental changes

There are many other potential changes in both addition of new information and removal of old information that can occur within a department. These changes occur seamlessly within the collegiate department and is the natural process in which an organization changes. The current state of information available in a collegiate department at a given point in time is the result of an ongoing evolution of additions and removals of information.

The next two sections describe two models of how a department website represents the current state of information within a collegiate department.

1.2 The traditional collegiate department website model

The main purpose of a traditional collegiate department website is to provide prospective students, current students, faculty, staff, and other academic and industrial professionals with information concerning the department. The website serves to increase the awareness of the issues that face the collegiate department. The following figure is a model of the traditional collegiate department's use and maintenance of a website.

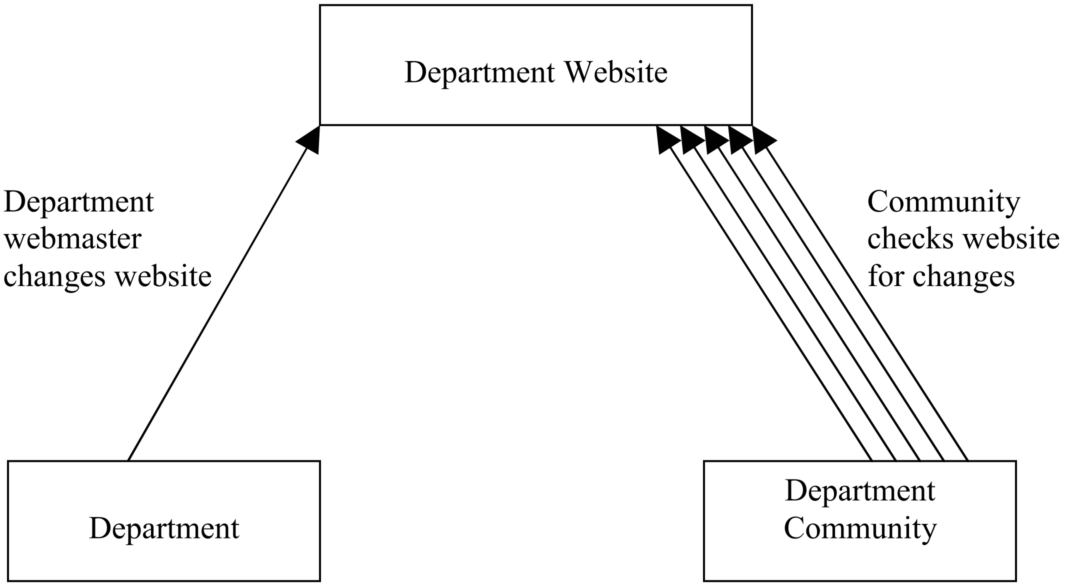

Department Website
Department webmaster changes website
Community checks website for changes
Department
Department Community

Figure 1 - Traditional Collegiate Department Website Model

This model represents how the traditional collegiate department maintains and uses its website. There are three components in this model. The Department Component represents the collegiate department. The Department Community Component represents anyone who is or wishes to be associated with the department. Finally, the Department Website Component is the information bridge that connects the two components. The Department Website Component is also a “snapshot” of a collegiate department at a given point in time (for an explanation about a “snapshot” see Section 1.2.7).

The following sub-sections describe the advantages and disadvantages associated with various parts of the traditional collegiate department model. These parts are: the department website, how the department webmaster changes the website, and how the community checks the website for changes.

1.2.1 Advantages of the traditional department website

The primary advantage of the traditional department website is that it takes relatively few resources to implement it. The creation of a website that provides static information does not require sophisticated software. Static information is best represented using HTML, which does not require highly experienced software developers.

1.2.2 Disadvantages of the traditional department website

Traditional collegiate department websites tend to be passive information repositories. The burden is on the user to periodically visit the website and search through web pages looking for new and changed information of interest. This active searching hampers their awareness, because no one actively searches all the time. According to Krug [1], users scan, skim, and glance at web pages trying to find the information they need in the least amount of time possible. Krug analogizes a user's reading of a web page to a "billboard going by at 60 miles an hour." Thus, expecting users to read for detail is unreasonable. However, collegiate websites are generally designed with precisely that expectation.

Students require interaction with other educational professionals as a part of their student development [2]. Van Duyne, et al., states that websites that offer interactivity "can be the conduits, the networks that bring together and help build school communities." Van Duayne, et al., defines the problem of educational forums:

> "By making the educational forum the central communication tool, all students benefit from sharing ideas, questions, and problems with classmates, parents, and teachers. The forums enhance intellectual and social life at school when students, faculty and parents interact and share ideas and values, thereby enhancing students' complete education."

The fundamental problem of creating interactivity and using a website as the central communication tool is having the technical capabilities for the website to be able to function as such. Collegiate department websites differ from the larger set of university websites in that they are relatively under funded and don't have the resources to provide the necessary tools to enable their website to be the "central communication tool." In fact, Van Duayne, et al., lists this as one of the major problems in creating an interactive educational website.

1.2.3 Advantages of relying on the community to check website

The primary advantage of requiring the community to check the department website is simplicity. It is the easiest and most low cost way of providing information to the community. If the department can rely on the community to check the website on a regular basis for new information, then the department would not have to worry about creating a mechanism to notify the community of changes to the website.

1.2.4 Disadvantages of relying on the community to check website

After the initial visit to the website, the user generally feels that he/she is aware of all departmental information and rarely revisits the website to check for new issues on a regular basis. As time goes by, the user may feel that the website has nothing new to offer, thus the user does not use the website to increase his/her awareness of current department information. These non-repeating users have outdated information, since the website could have changed. Users who do not want to check the website on a consistent basis cannot obtain all the new information and issues about the department. This particular problem gives rise to another; the department has no way to announce or notify the users of the website when new information is posted. Therefore, inability to alert the community of changes to the website, and ultimately changes to the department, is a problem.

1.2.5 Advantages of only webmasters making changes to the website

Relying on one or a few webmasters to manage the content of the department website allows the webmasters to be in direct control of the information on the department website. This has several advantages. First, the department can be assured that the webmasters will provide correct and appropriate information on the website. Second, having a few skilled webmasters create and edit web pages on the department website keeps the “look and feel” of the website consistent.

1.2.6 Disadvantages of only webmasters making changes to the website

The ability to provide new information about the department through a website relies too heavily on the webmaster. The webmaster has the immense responsibility to change the necessary HTML source code to reflect all new information. An example of the process to add new information to the website is as follows: a professor just published a new paper, the professor wants to post this paper online and announce its existence, the professor then contacts the webmaster, the webmaster puts this request on his/her things to do list, eventually the webmaster completes the request, then the professor must check the webmaster's work, and so on. Imagine this scenario for every professor in the department to route all new information to one webmaster who is solely responsible to make the changes. This process slows down the delivery and frequency of new information being added to the website.

The collegiate departments do not have the resources available to constantly improve their website or create features/services. New information and issues are placed on websites less frequently than they occur and problems or errors with the website are not fixed promptly.

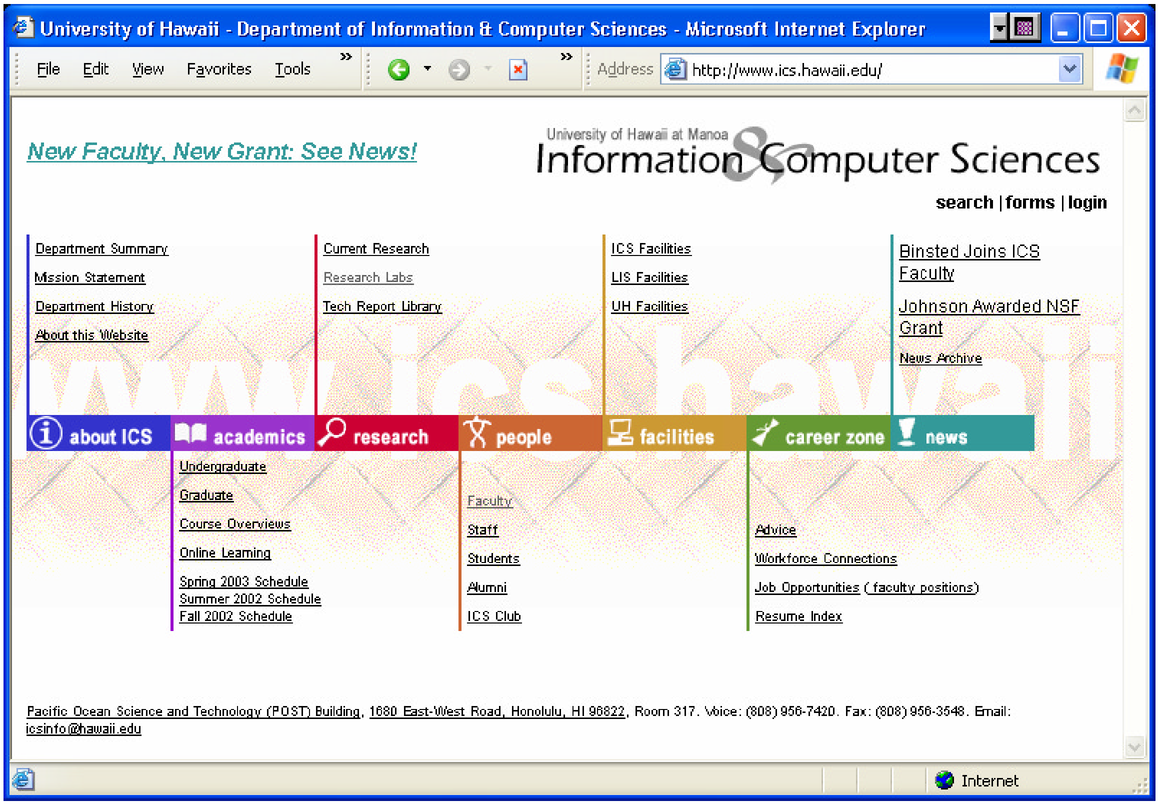
1.2.7 Representing departmental changes in the department website.

The traditional collegiate department website model does not provide the functionality to represent the departmental changes, explained in Section 1.1. Rather, it provides only high-level information to the users of the website. For example, it provides academic requirement information, which, does not change on a regular basis.

I claim that the traditional collegiate department website models a “snapshot” of the department at a given time. I call this a “snapshot” because the traditional department website model does not have the tools to continually update the department website with the changes (additions and removals) that occur in the actual department. Therefore, the traditional department website represents the current state of information at the time the “snapshot” was taken.

1.3 Issues with the UH ICS department website

The University of Hawaii at Manoa’s Department of Information and Computer Science’s website (Figure 2), which can be found at http://iwi.ics.hawaii.edu, is a specific example of the traditional collegiate department model.



Pacific Ocean Science and Technology (POST) Building, 1680 East-West Road, Honolulu, HI 96822, Room 317. Voice: (808) 956-7420. Fax: (808) 956-3548. Email: icsinfo@hawaii.edu

Figure 2 - University of Hawaii Department of Information and Computer Science's website, as of December 2002.

1.3.1 The UH ICS department website

The UH ICS department website, available since Summer 2002, is graphically well designed and rich in content. The website provides a set of department information. This set includes a department summary, a mission statement, a department history, research, undergraduate and graduate information, employment opportunities, and information about the people of the department: faculty, staff, and students.

Like the traditional collegiate department model, the UH ICS department website suffers from the amount of reading required to gain information. Also, the website is not designed to be the main communication tool that the department relies on to get important information to the community.

1.3.2 The community checks the website

Relying on the community to check the website for new information causes a major problem. The UH ICS department website tries to highlight the new issues that face the department by providing a news page. This news page can be effective if the user can be told to check this page when new information is available. However, the UH ICS department does not have that functionality, so the department simply hopes that the community will frequently visit the website. In fact, the entire website is based on the assumption that the community will check websites for changes.

1.3.3 The process of making changes to the website

The UH ICS department primarily relies on one webmaster to update and manage the website. Since, the organizational and intellectual state of the department changes on a daily basis, modeling the department on the website is not easy for the webmaster.

The UH ICS department website does offer minimal interactive features such as the ability to have your email address listed in the "People" section of the website and the ability to have your resume listed. By using interactive features, the webmaster is removed from the responsibilities of updating that particular section. This shows that the UH ICS department website does deviate, in small ways, from the traditional collegiate department website model. However, there are problems with the UH ICS department's interactive features. One problem is that it requires extensive searching to find the interactive services. They are "hidden" and not explicitly advertised to the user. As a result, on November 12, 2002, only 80 people posted their resumes, only 66 students are listed in the student listing, and only 27 alumni are listed in the alumni listing. However, as of the Spring 2003 semester, the ICS department has 845 declared major students, 29 faculty and instructors, and 1200 alumni that have graduated from the UH ICS department. The low ratio of the number of resumes and student listings compared to the number of people in the department reflects the website's inability to facilitate user interaction with the website.

1.3.4 Representing departmental changes in the department website.

The previous section explains how the UH ICS department website's interactive services allow some changes of additions by allowing the department community to post new information to the website. The previous section also explains some of the problems associated with the use of the UH ICS department website's interactive services. The UH ICS department website also does not provide interactive services to manage the changes of removals that occur in their department. Therefore, the UH ICS department website does not model the actual department very well. There is a lot of outdated information on the department website and a relatively low amount of additions of new information.

1.3.5 Interview with the UH ICS department webmaster

The UH ICS department has one webmaster. However, this person is also a full-time faculty member. This dual role of faculty and webmaster provides a strain on the webmaster's responsibilities. To find some of the tasks that the webmaster is responsible for, I conducted an email interview with the webmaster and the following questions were asked:

1. What are your specific duties as the department webmaster?

2. Do you think that the webmaster's duties are too overwhelming for one person to adequately accomplish?

3. Do you feel that our department website provides adequate functionality and information to the community?

4. Is there outdated information on the department website? If so, do you remove that information?

1.3.5.1 Webmaster's response to question 1

Question 1: What are your specific duties as the department webmaster?

1. Course Schedules (Each Semester):

Each semester the course schedules are sent to the webmaster either in MS Word or Excel documents. Then, the webmaster posts the course schedule to the website for students before they appear on Pa'e (the UH Manoa student registration system). During this process the webmaster removes old course schedules, but has not been able to do so for the last semester because the webmaster has lost edit access to the server. Also, the webmaster provides links to specific course websites, descriptions of special topic classes, and provides other information where applicable about the courses.

The webmaster expressed that this process is "pain" as the website construction does not support easy updates to the navigation system.

2. News Bulletins (As Needed):

The webmaster posts news bulletins to the website for new hires, grants, awards, and equipment donations. The current system provided a dynamic administrative news posting service. This system posts news items to a "news category" automatically.

However, it did not allow the webmaster to edit the news item once it was posted. It also did not provide a link to the news item from the homepage.

3. Current Research (Occasionally):

The webmaster admits that this page has not been updated in a while.

1.3.5.2 Webmaster's response to question 2

Question 2: Do you think that the webmaster's duties are too overwhelming for one person to adequately accomplish?

The webmaster definitely thinks that the responsibilities of a webmaster are too great for one person. The webmaster expressed, it is "definitely too much for a faculty member who is also running several other projects." The webmaster wanted to branch some work out to other staff members, however that did not occur.

1.3.5.3 Webmaster's response to question 3

Question 3: Do you feel that our department website provides adequate functionality and information to the community?

The webmaster states that the department website is lacking in interactive services. He expressed a need for a better news facility and a technical report library. Also, he stated that the student listing needs some advertisements to let students know to go to the website and create their student listing.

1.3.5.4 Webmaster's response to question 4

Question 4: Is there outdated information on the department website? If so, do you remove that information?

The webmaster realizes that the removal of information is a definite issue of concern to keep the department website up-to-date. The webmaster states that, "this is surely where I fail in my duties!" The webmaster explains that there are possibly many pages that have dated information on it.

1.4 Introduction to the Collaborative Educational Website (CLEW)

The Collaborative Educational Website (CLEW) is a Java web application designed to support accurate modeling of a collegiate department by solving the disadvantages of the traditional collegiate department model.

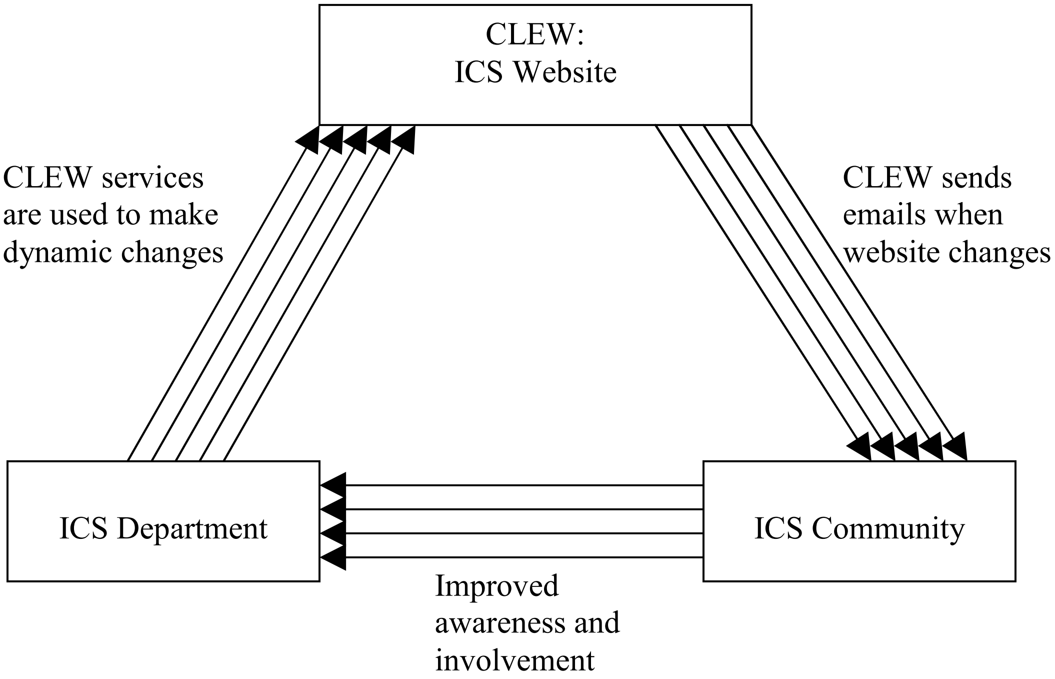

CLEW:
ICS Website
CLEW services are used to make dynamic changes
CLEW sends emails when website changes
ICS Department
ICS Community
Improved awareness and involvement

Figure 3 - CLEW Collegiate Department Website Model

The CLEW collegiate department website model consists of the same three components as the traditional collegiate department website model: the department, the community, and the website. However, the means which the department, community, and website communicate have been improved. The model depicted in Figure 3 shows a circular movement. Based on this model, communication between the department, website, and community is designed to be a continuous process. The following sections explain how the CLEW collegiate department website model is designed.

1.4.1 The CLEW Website

The CLEW Website assists the user in searching for information. The ability to interact with the website, by filling out a form or through a series of clicks, allows the user to get information tailored to their request. This allows easier retrieval of information, which allows the user to scan through the web pages as suggested by Krug [1].

1.4.2 The CLEW Email Notifications

CLEW alerts the department community of changes to the website by email. This allows the user to be notified when new departmental information is available. When new information is posted, email notifications are sent out once a day. The user may view the new information at their discretion and need not continually check the website. Email notifications are an optional feature.

1.4.3 The CLEW Services

CLEW aids the department in the posting of new information. CLEW allows the user, who has the permission, to post information that pertains to the department. The user does not need to alter HTML source code. Instead, users can fill out a form about their "posting" and the information is made instantly available.

The "posting" of information to the department website CLEW is interactive. Within CLEW are several services that provide interactivity; a frequently asked questions service, a questionnaire service, a technical reports service. Each of these services provides functionality allowing the user to interact with the department website.

1.4.4 Advantages of the CLEW Model

There are several advantages of the CLEW collegiate department website model. First, by utilizing the CLEW Email Notifications, the department community can be automatically informed of changes to the department. Users of the CLEW Website can rely on the CLEW Email Notifications to have tailored alerts to new information that they care about. Email notifications solve the problems associated with the traditional collegiate department website's reliance on users checking websites for changes.

Second, the CLEW Services provide users with a dynamic way to update the website without having to go through the long process of seeking out and requesting the webmaster to add new information. This has the potential to enable more robust and timely information about the department. Interaction with the department website helps the users, the department community, become involved with the department. Also, if the interactive and dynamic CLEW Services are used effectively, it will enable the department website to better model the physical department by allowing the changes in the physical department to be mirrored in the department website.

1.4.5 Disadvantages of the CLEW Model

There are several disadvantages of the CLEW Model. First, the department community can only take advantage of the CLEW Model if they happen to be introduced to the website. For example, the CLEW website, email notifications, and services can only benefit someone who is aware of their existence. Second, creating a CLEW Website requires substantially more resources than the traditional collegiate department website. Third, users need to learn how to enable these services to contribute content and be informed of changes. Fourth, the CLEW collegiate department website model requires the department community to actively participate in keeping the website updated.

1.4.6 Representing departmental changes in the department website.

The CLEW collegiate department website model does provide the functionality to represent changes of additions, explained in Section 1.1, in the CLEW Website. This functionality is provided by the CLEW Services which are designed to improve the UH ICS department website's interactive services. The effectiveness of CLEW's ability to support the additions of information will be evaluated in Chapter 4. However, like the UH ICS department website, CLEW does not currently provide interactive services to manage the changes of removals that occur in their department. Therefore, the CLEW Website does not model the actual department very well. There could be a lot of outdated information on the CLEW Website. However, the CLEW System provides the ability to easily implement services that monitor the potential for old information on the department website.

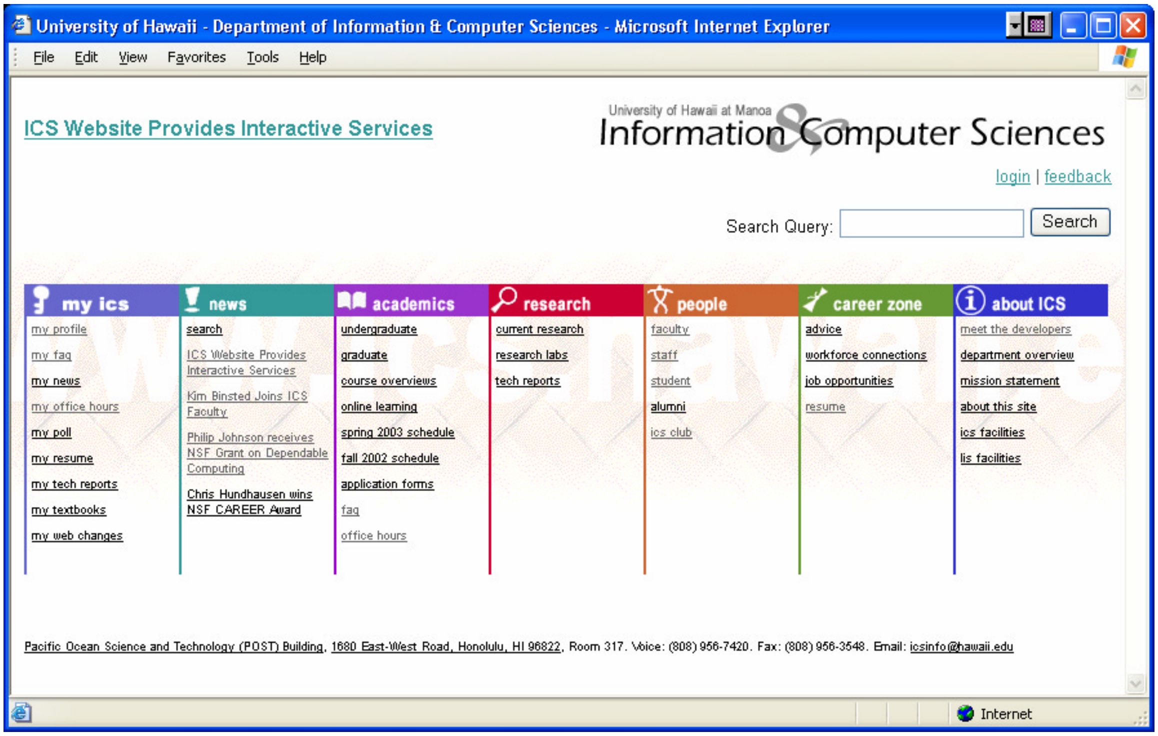
1.5 A guided tour of CLEW

The following section is a quick-guided tour of the CLEW System.

1.5.1 CLEW user scenarios

At the top level, there are two kinds of users in the CLEW System: registered and unregistered. An unregistered user does not have an account with CLEW and cannot take advantage of the interactivity provided by CLEW. However, unregistered users have access to the “public” portions of the website. A registered user, however, can utilize the interactive features of CLEW. In the registered category there are six subtypes or “roles”: administrators, faculty, staff, students, alumni and employers. CLEW provides different permissions and access to each of these roles. The next sections provide descriptions of several usage scenarios which a user may have with CLEW.

1. **User visits the CLEW Website.** The user visits the website to interact with the website. The user can select any service listed in the "my ics" section of the website or they can view any webpage in any of the other sections, including: about ics, academics, research, people, career zone, and news.

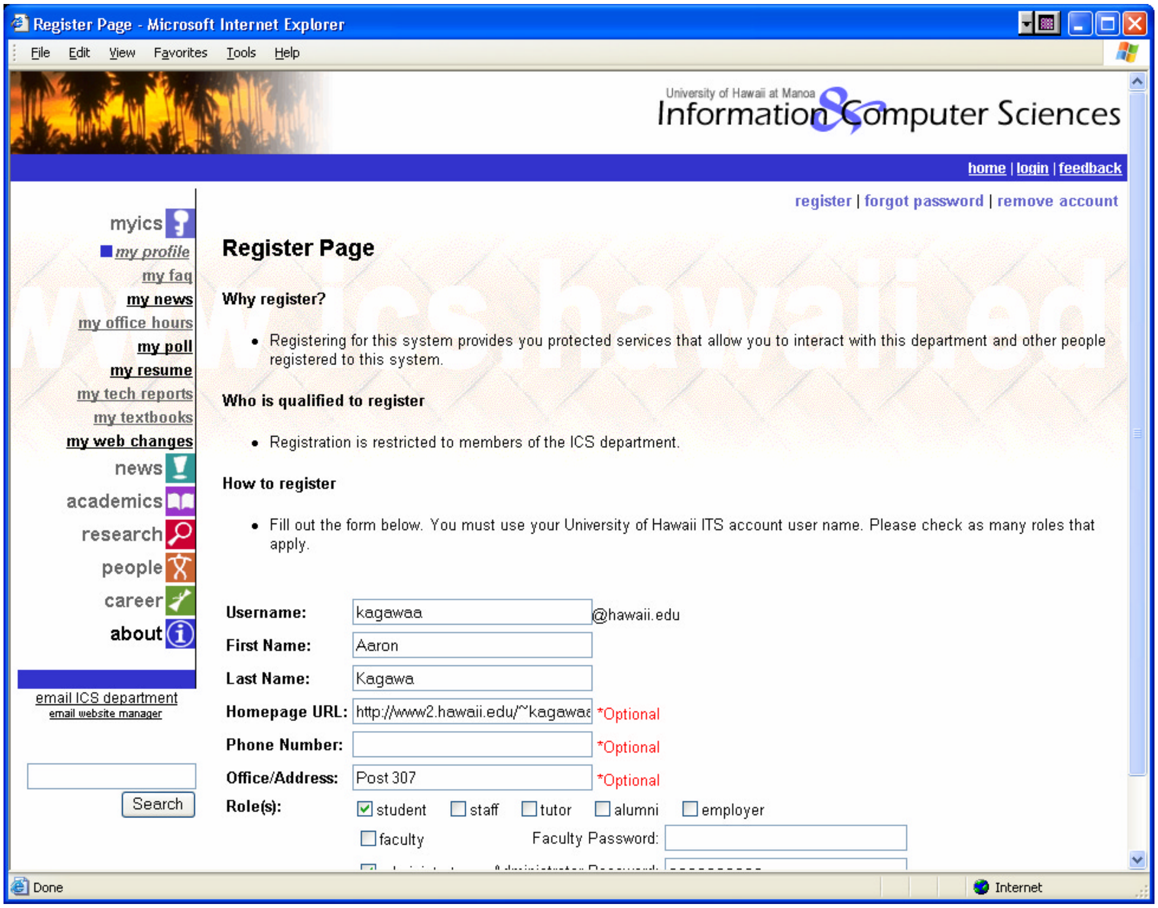


Pacific Ocean Science and Technology (POST) Building, 1680 East-West Road, Honolulu, HI 96822, Room 317. Voice: (808) 956-7420. Fax: (808) 956-3548. Email: icsinfo@hawaii.edu

Figure 4 - CLEW home page, as of January 2003.

2. **An unregistered user registers with CLEW.** An unregistered user or potential user of CLEW will have to go to this webpage to register and create an account with CLEW. This registration will allow this potential user to be able to login to the login-protected services that CLEW provides. Without registering, this user will only be able to access the unprotected web pages such as, mission statement, undergraduate information, etc.

Registration includes giving CLEW the user's UH ITS username which will be used by CLEW as the username. CLEW also requires the user's first name, last name, and most importantly what roles the potential user wants to have. For example, a faculty member must register with CLEW selecting the "faculty" role. Otherwise, this faculty member will not have special services associated with the faculty role.



Register Page

Why register?

- Registering for this system provides you protected services that allow you to interact with this department and other people registered to this system.

Who is qualified to register

- Registration is restricted to members of the ICS department.

How to register

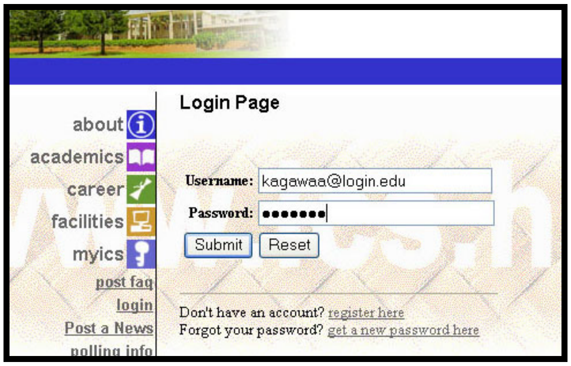
- Fill out the form below. You must use your University of Hawaii ITS account user name. Please check as many roles that apply.

Username: kagawaa @hawaii.edu
First Name: Aaron
Last Name: Kagawa
Homepage URL: http://www2.hawaii.edu/~kagawaa *Optional
Phone Number: *Optional
Office/Address: Post 307 *Optional
Role(s): ☑ student □ staff □ tutor □ alumni □ employer
□ faculty Faculty Password:
☑ administrator Administrator Password: ••••••••••

Figure 5 - CLEW registration page

3. Login Page and My ICS Homepage. After accessing the website the user can either browse the unprotected web pages, or may proceed to access the CLEW Services. All CLEW Services are login-protected. The Login Page will appear when trying to access any service. The user then proceeds to enter the login information.

At the completion of the login verification, the user is directed to the service they requested or to the My ICS Homepage. The My ICS Homepage provides information about the user's account profile and provides services to edit or remove the user's profile or log out entirely from the system.



Login Page

Username: kagawaa@login.edu

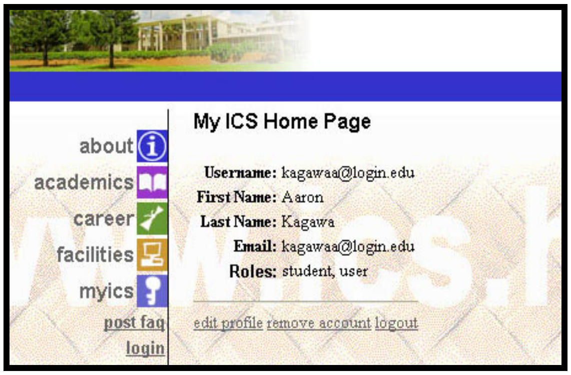
Password: •••••••

Submit Reset

Don't have an account? register here

Forgot your password? get a new password here

Figure 6 - CLEW login page



My ICS Home Page

Username: kagawaa@login.edu

First Name: Aaron

Last Name: Kagawa

Email: kagawaa@login.edu

Roles: student, user

edit profile remove account logout

Figure 7 - CLEW my ics home page

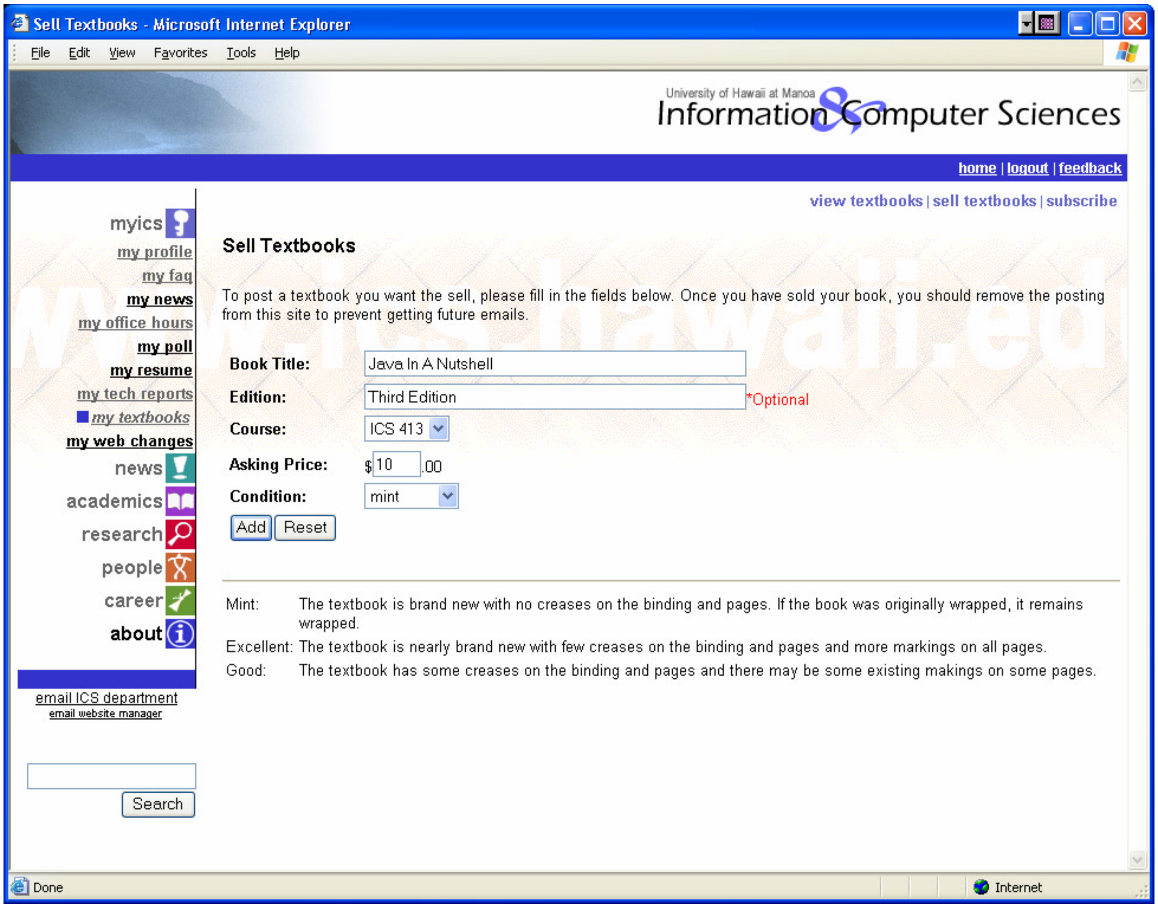
1.5.2 CLEW Services

Each of the CLEW Services has a login-protected portion of the service. This allows only the registered users to have access to the protected portion of these services. This is an incentive to get users to register with the system. Also, this provides minimum protection for the system from “crackers” and other disruptive actions.

Some of the CLEW Services also have an unprotected portion of the service. This allows unregistered and registered users alike to get the dynamic information that is created by the service.

The following sections illustrate some of the services that CLEW provides.

1. CLEW Textbooks Service. The CLEW Textbooks Service is one example of the services that CLEW has to offer. This service allows the user to purchase and post books for sale.



Sell Textbooks

To post a textbook you want the sell, please fill in the fields below. Once you have sold your book, you should remove the posting from this site to prevent getting future emails.

Book Title: Java In A Nutshell

Edition: Third Edition *Optional

Course: ICS 413

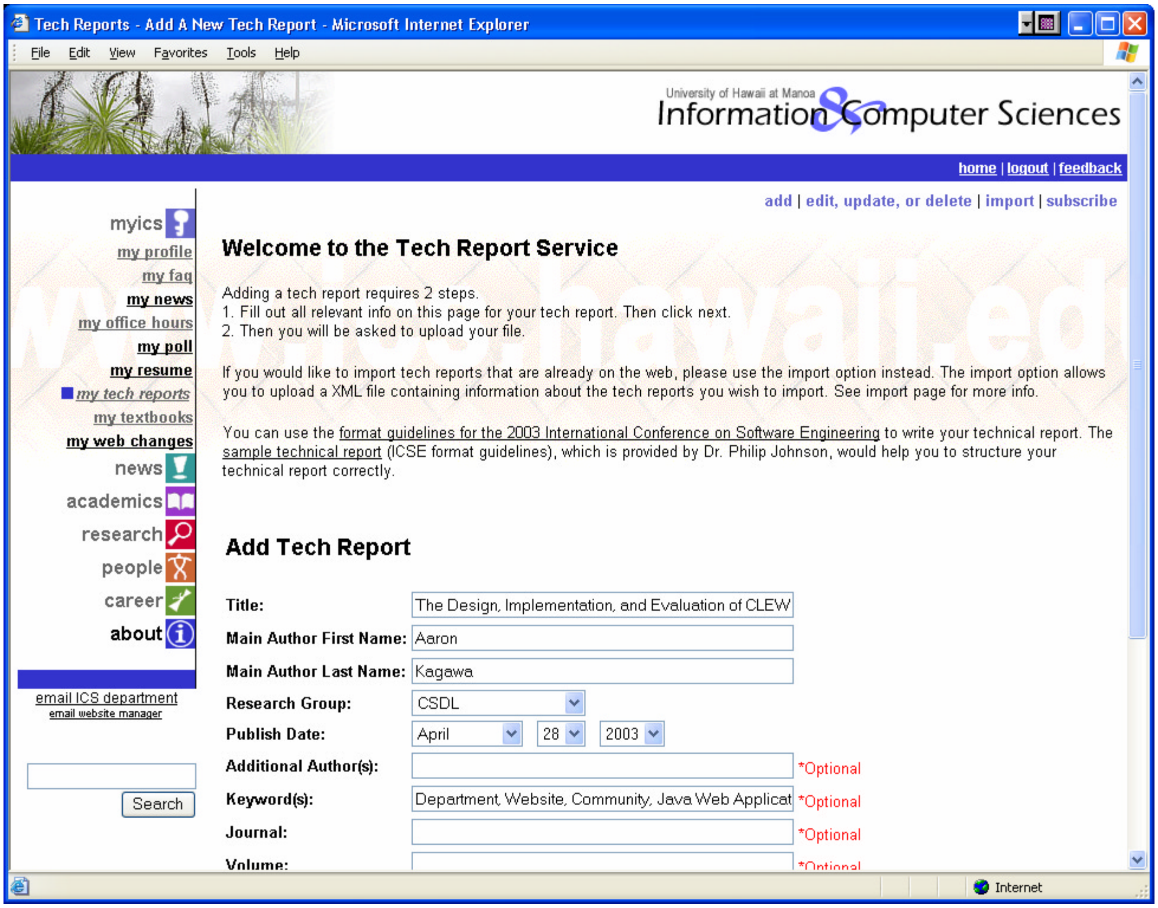
Asking Price: $ 10 .00

Condition: mint

Mint:	The textbook is brand new with no creases on the binding and pages. If the book was originally wrapped, it remains wrapped.
Excellent:	The textbook is nearly brand new with few creases on the binding and pages and more markings on all pages.
Good:	The textbook has some creases on the binding and pages and there may be some existing makings on some pages.

Figure 8 - CLEW textbook index page

2. CLEW Tech Report Service. The CLEW Technical Report Service allows users to post technical reports on the website; generally there is no distinction between technical reports and any other publications. Information pertaining to the technical report and the actual technical report file, which must be uploaded to the system, is posted on the website using a form that the Tech Report Service provides. Figure 9, provides a screenshot of the login-protected section of the technical report service. However, there is an unprotected section to this service, where anyone (unregistered users and registered users), may view all the technical reports on the website.



Welcome to the Tech Report Service

Adding a tech report requires 2 steps.
1. Fill out all relevant info on this page for your tech report. Then click next.
2. Then you will be asked to upload your file.

If you would like to import tech reports that are already on the web, please use the import option instead. The import option allows you to upload a XML file containing information about the tech reports you wish to import. See import page for more info.

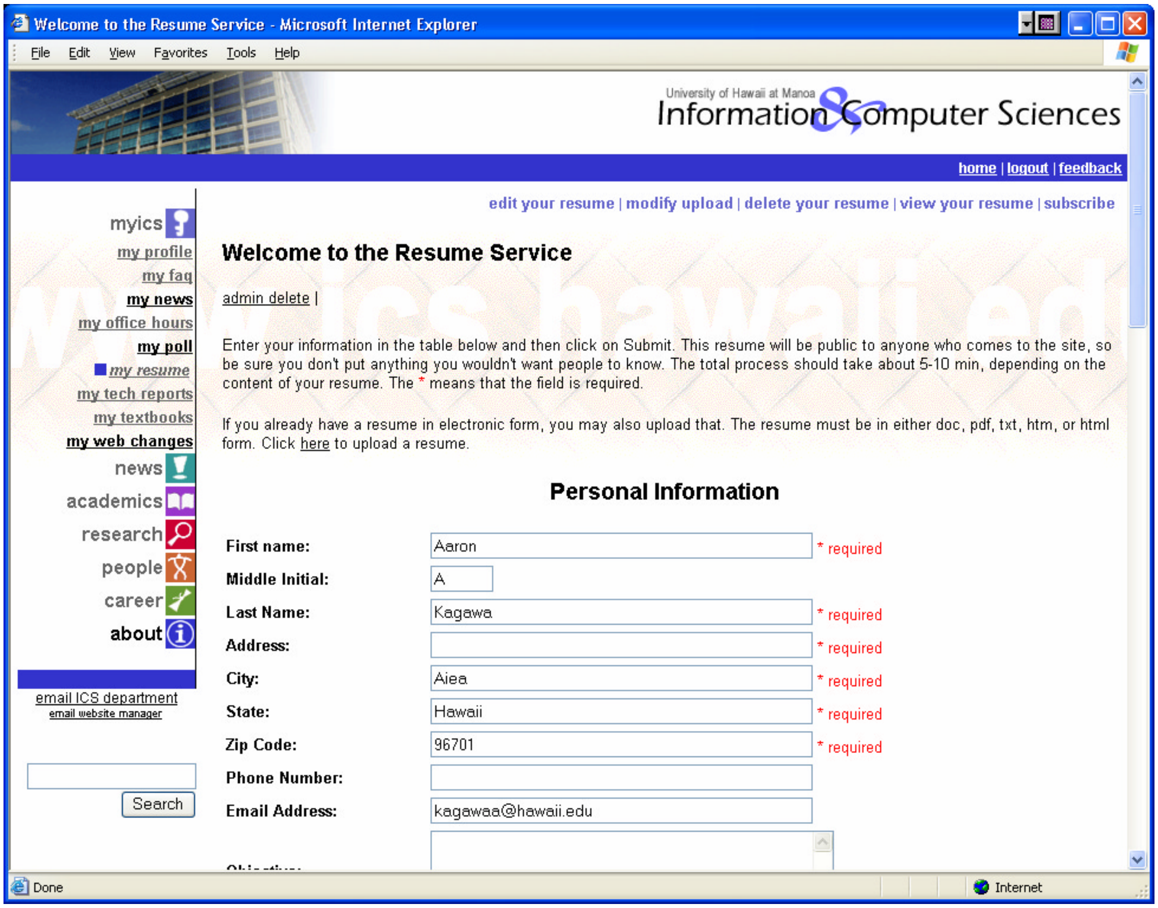
You can use the format guidelines for the 2003 International Conference on Software Engineering to write your technical report. The sample technical report (ICSE format guidelines), which is provided by Dr. Philip Johnson, would help you to structure your technical report correctly.

Add Tech Report

Title:	The Design, Implementation, and Evaluation of CLEW	
Main Author First Name:	Aaron	
Main Author Last Name:	Kagawa	
Research Group:	CSDL	
Publish Date:	April 28 2003	
Additional Author(s):		*Optional
Keyword(s):	Department, Website, Community, Java Web Applicat	*Optional
Journal:		*Optional
Volume:		*Optional

Figure 9 - CLEW technical report index page

3. CLEW Resume Service. The CLEW Resume Service allows registered users to create and edit their personal resumes. This function is login-protected and ensures that the user only modifies their personal information. However, there is an unprotected, public section to this service which allows any user, unregistered users and registered users, to view all posted resumes from the department website.


Welcome to the Resume Service - Microsoft Internet Explorer
File Edit View Favorites Tools Help
University of Hawaii at Manoa
Information & Computer Sciences
home | logout | feedback
edit your resume | modify upload | delete your resume | view your resume | subscribe
myics
my profile
my faq
my news
my office hours
my poll
my resume
my tech reports
my textbooks
my web changes
news
academics
research
people
career
about
email ICS department
email website manager
Search
Done
Internet

Welcome to the Resume Service

admin delete |

Enter your information in the table below and then click on Submit. This resume will be public to anyone who comes to the site, so be sure you dont put anything you wouldn't want people to know. The total process should take about 5-10 min, depending on the content of your resume. The * means that the field is required.

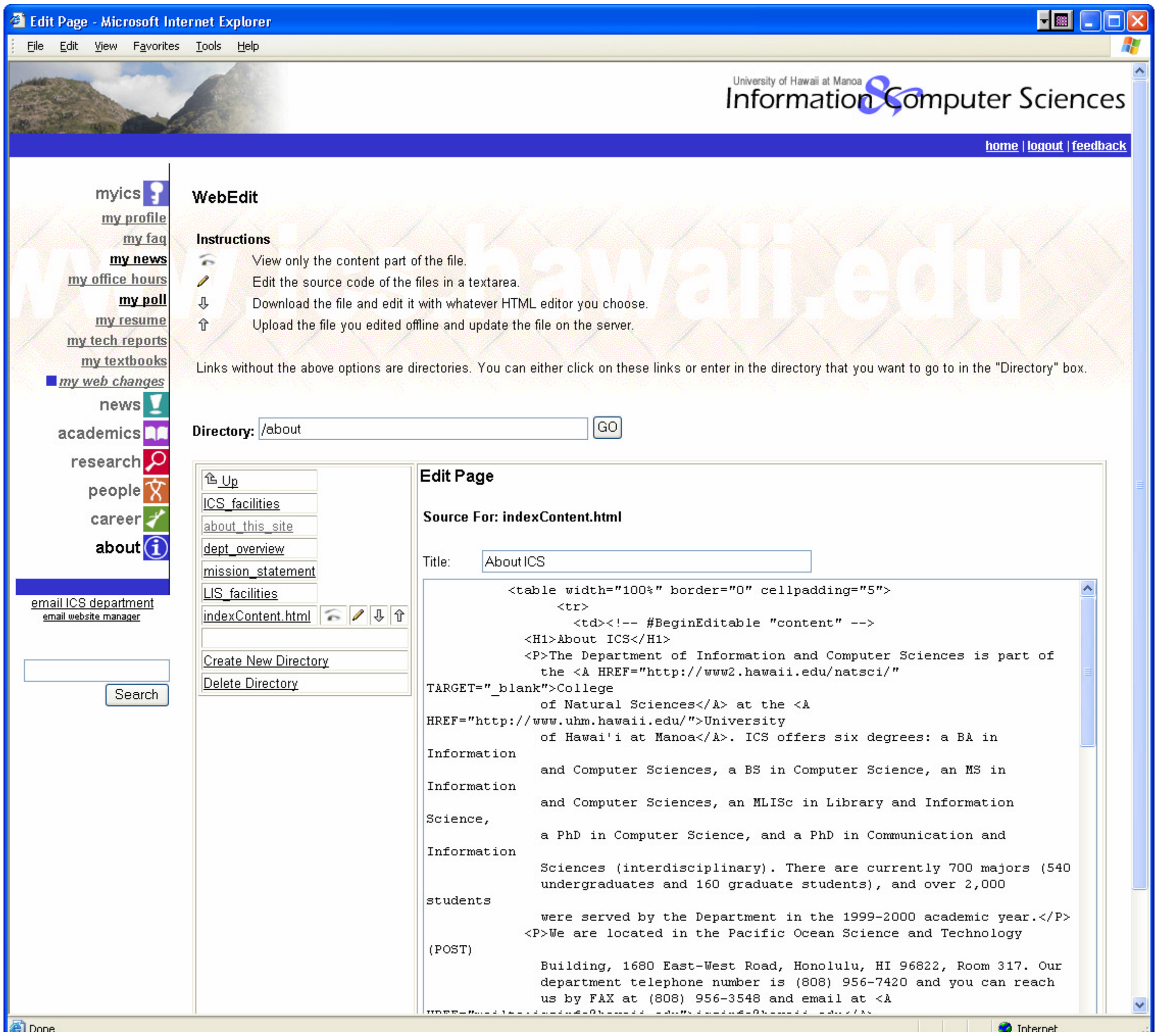
If you already have a resume in electronic form, you may also upload that. The resume must be in either doc, pdf, txt, htm, or html form. Click here to upload a resume.

Personal Information

First name:	Aaron	* required
Middle Initial:	A	
Last Name:	Kagawa	* required
Address:		* required
City:	Aiea	* required
State:	Hawaii	* required
Zip Code:	96701	* required
Phone Number:		
Email Address:	kagawaa@hawaii.edu	

Figure 10 - CLEW resume edit page

4. CLEW Webedit Service (Web Changes). The CLEW Webedit Service allows users with the administrative role to dynamically alter the HTML source code of web pages in the CLEW System. It provides an easier way of making changes by eliminating the need to download and upload HTML files to the server.



WebEdit

Instructions

- View only the content part of the file.
- Edit the source code of the files in a textarea.
- ⇩ Download the file and edit it with whatever HTML editor you choose.
- ⇧ Upload the file you edited offline and update the file on the server.

Links without the above options are directories. You can either click on these links or enter in the directory that you want to go to in the "Directory" box.

Directory: /about GO

Edit Page

Source For: indexContent.html

Title: About ICS

```
          <table width="100%" border="0" cellpadding="5">
               <tr>
                 <td><!-- #BeginEditable "content" -->
            <H1>About ICS</H1>
            <P>The Department of Information and Computer Sciences is part of
              the <A HREF="http://www2.hawaii.edu/natsci/"
TARGET="_blank">College
              of Natural Sciences</A> at the <A
HREF="http://www.uhm.hawaii.edu/">University
              of Hawai'i at Manoa</A>. ICS offers six degrees: a BA in
Information
              and Computer Sciences, a BS in Computer Science, an MS in
Information
              and Computer Sciences, an MLISc in Library and Information
Science,
              a PhD in Computer Science, and a PhD in Communication and
Information
              Sciences (interdisciplinary). There are currently 700 majors (540
              undergraduates and 160 graduate students), and over 2,000
students
              were served by the Department in the 1999-2000 academic year.</P>
            <P>We are located in the Pacific Ocean Science and Technology
(POST)
              Building, 1680 East-West Road, Honolulu, HI 96822, Room 317. Our
              department telephone number is (808) 956-7420 and you can reach
              us by FAX at (808) 956-3548 and email at <A
```

Figure 11 - CLEW webedit edit page

1.6 Thesis Statement

CLEW will support accurate modeling of the Department of Information and Computer Science at the University of Hawaii at Manoa:

1. Improving the ability to dynamically make changes to the website.
2. Creating a notification system to alert the community of changes to the website.
3. Improving the community's awareness of the department.
4. Creating a "kernelized" framework that will support the creation of the services.

1.7 Evaluation of Thesis Statement

The evaluation of the CLEW System is organized to evaluate each of the four thesis claims. The following section provides a brief description of the evaluations.

1.7.1 Evaluation of Thesis Claim 1

Claim 1: CLEW will improve the ability to dynamically make changes to the website.

CLEW provides several ways to dynamically change the website. The CLEW Resume Services, CLEW News Bulletin Service, and the CLEW People Listing Service were created to replace the Resume, People, and News Listings of the original UH ICS department website. To evaluate if CLEW improves the ability to make dynamic changes, I have conducted an evaluation of the usage of the CLEW Services against the versions provided in the original website. The evaluation includes monitoring the number of “postings” to the original UH ICS department website’s interactive features to the CLEW Services.

1.7.2 Evaluation of Thesis Claim 2

Claim 2: CLEW will notify CLEW’s users of changes in the department website by creating a notification system to alert the community of changes to the website.

CLEW provides email notifications to CLEW’s users when changes occur to the website. Notifications allow CLEW’s users to play a passive role in becoming aware of the changes in a department. To evaluate email notifications, I have conducted an evaluation of the usage of the email notifications. The evaluation methodology includes comparing the number of CLEW’s users who receives email notifications against the total number of CLEW’s users.

1.7.3 Evaluation of Thesis Claim 3

Claim 3: CLEW will improve the community’s awareness of the department.

One of CLEW’s overall goals is to improve the department community’s awareness of the department. I have conducted an evaluation using a pre-post release questionnaire methodology, to determine how the community views their own awareness levels before and after the release of CLEW. The pre-release questionnaire was administered while the original UH ICS department website was still in operation. Then, CLEW replaced the original UH ICS department website. After 3 weeks of using the CLEW Website, the post-release questionnaire was administered to the same participants.

1.7.4 Evaluation of Thesis Claim 4

Claim 4: CLEW will improve the websites ability to provide services to the department community by creating a “kernelized” framework that will support the creation of such services.

Another of CLEW’s overall goals is to provide a “kernelized” framework for the creation of services. I have conducted an evaluation using a questionnaire, to determine the CLEW Service developers’ experiences in creating CLEW Services using the CLEW Kernel framework.

1.8 Results

The results of the CLEW collegiate department website model evaluations provide evidence that CLEW improves the ability to dynamically change the department website, CLEW can notify CLEW’s users of changes in the department website by creating a notification system, and CLEW can improve the community’s awareness of the department.

Feedback about the CLEW Website was obtained from the website’s users. Results of the feedback evaluations provide evidence that the CLEW System can be used as a department website without significant problems.

The developers of the CLEW Services were asked to express their concerns about developing the CLEW Services in a questionnaire. The results from the developer evaluations show that the developers enjoyed and learned a lot about various software technologies and Software Engineering. They highly recommend CLEW as a project to provide educational value for Software Engineering courses.

1.9 Conclusion

The results of the evaluations provide evidence that CLEW supports accurate modeling of the Department of Information and Computer Sciences at the University of Hawaii at Manoa. The results also provide evidence that the ICS department has successfully adopted the use of the system. Therefore, the CLEW System has been, to this point, successful. CLEW can still serve as the department website and development can continue on the CLEW System.

CLEW has been a valuable learning device for many of the developers that worked on CLEW. The results of developer evaluations show that the development of CLEW Services has added educational value to the developers' Computer Science education. If for no other reason, development of CLEW should continue in Software Engineering courses to provide students with hands on software development experience.

1.10 Structure of the proposal

In the second chapter, I will present work related to CLEW. I will discuss two research projects that relate to collegiate department websites.

In the third chapter, I will discuss the history, design, and implementation of CLEW. To provide an overview of the development of CLEW I will discuss several details in each of the three areas.

In the fourth chapter, I will present the methodology that have used to evaluate CLEW. The steps needed to gather information, and how I will interpret that data.

In the fifth chapter, I will provide the results of the evaluations done on CLEW. I will also conclude what claims of my thesis statement are supported by the results.

In the sixth and final chapter, I will present the future directions of CLEW development and research.

Chapter 2. Related Work

2.1 The users' needs of a collegiate department website

Collegiate department websites are a valuable resource to its users. The website must support users' needs and provide the appropriate information to them. But, what exactly do users need from a department website? The following section presents an example of a user study to find what users need from a department website.

Ritter, et al. [3], state that users of a department website expect to "find particular pieces of information." Regardless of a wide divergence of the presentation and the actual content of the website, there should be a common set of tasks that the website should support. A user study using task analysis on department websites was conducted by Ritter, et al, to find a common set of tasks that should be supported in a department website. The goal was to create a "check list" of tasks that other departments could use to create, evaluate, and improve their department websites.

The first step in this user study was to define the set of users who use department websites. The set of users that Ritter, et al., created can be found in Appendix E.

Tasks analyses were created using department hard copy handout materials, search queries, and interviews to find what users look for on department websites. Task analysis is a HCI technique used to identify needs and establish requirements and usually focuses on investigating an existing situation.

One of their findings in gathering information on hardcopy materials is that, website printouts were included in the hardcopy material. Based on that observation they concluded, “that handout designs may be influenced by the website,” and that “hardcopy materials and department websites should be designed to work together.”

The set of tasks that Ritter, et al., defined can be viewed in Appendix E. Using this set of tasks, Ritter, et al., evaluated five department websites. The five department websites averaged having 70 percent of the tasks that Ritter, et al. defined. They did not provide the significance of this number. Ritter, et al, state that collegiate departments can use the set of tasks as a guideline and decide on whether or not each of the tasks are appropriate for their specific department.

Ritter, et al., also suggests that regular maintenance of the website is just as important as creating it. “Websites that become outdated decline in quality, so you should protect your investment by spending time to maintain it and keeping it up to date.” “This is particularly true for dynamic organizations like university departments.” They also state that they believe “the most practical way to keep a website updated is to devolve the maintenance of the website from the webmaster to those who create or manage the information directly.” This approach places the responsibility on the department community to make the changes that they desire to occur.

Ritter, et al., briefly mention two features that could help in the use of a department website. The first is email notifications. Email notifications can be used to keep users informed of changes to the website. The second feature is adding search functionality within the website. It provides an alternate way to find information if the design of the navigation links fails. Also, users might find searching a quicker way to find certain information or if they are uncertain where the information is located.

Unfortunately, they provide no hard evidence that these two features will be useful to the users of the website.

The most profound statement that Ritter, et al, provides is "the largest lesson that we continue to relearn, is that the online world parallels the real world." They suggest that the physical workings of a department website should be mirrored in the department website.

2.1.1 How the Ritter, et al, user study relates to this CLEW research

The set of tasks that a department website should support created by that Ritter, et al, strongly resembles the traditional collegiate department website. It resembles the notion of an information repository that the users of a department website can go to find information. In fact, Ritter, et al., expressed that the department websites had a very strong correlation between hardcopy department information handouts (e.g. information booklets and flyers), which are physical information repositories. They provide a lot of examples of how the handout information can and should be presented in a department website. In fact, I would claim that the main purpose of the traditional collegiate department website is to provide this sort of information, which almost always tends to be static, non-changing information.

Ritter, et al., also expresses that the department website should mirror the actual physical department. This introduces an idea that department websites should be dynamic and be able to change in accordance to the changes of the actual physical department. This idea is the primary goal of CLEW. The CLEW Services provide interactive services to all users to dynamically change the department website. Of course, the users must have the initiative to use the services to mirror the actual department, and CLEW's goal is to provide a means to do this. In fact, Ritter, et al., claims that the responsibility of updating the website should go to the actual community. Unfortunately, Ritter, et al. did not explain or provide examples on how the department website is to mirror the actual department.

The user study also suggests that some features such as email notifications and search functions will be useful to department websites. This validates, to some degree, CLEW's use of email notifications to allow its users to play passive roles in becoming aware of the changes to the department website. Furthermore, it also suggests that there are other features, or services, that can enhance the users experience with department websites.

2.2 Collegiate department websites used to create a sense of community

In Andrada's master's thesis [4], "Building Community Through The World Wide Web," she claims that a collegiate department website should also be able to support a "community" feeling. "Community," as defined by Andrada, is "based on feelings people have about themselves to their organization and on their knowledge of other members and the group as a whole". In Andrada's masters thesis she investigated how a department website could support building a sense of community in a department. Building a sense of community within a department is important because it can build morale and help the department function better as a whole.

A collegiate department is a community based on interest, which means that "interests and issues are what gathers people together with great inspiration and motivations". As with any other subjective matter, the sense of community is not a binary entity. Communities can exist in varying degrees. Furthermore, the sense of community are created from various parts, they include: each group member's sense of significance, their sense of solidarity, and their collective sense of self-awareness. Andrada's research questioned if a department website can improve the sense of community by improving members' feelings of importance, feeling of belonging, and their self-awareness of the people and issues in the department.

The finding of Andrada's research is that a department website can contribute to some degree in building a community in a collegiate department. Also, there are multiple communities in a department; the faculty and staff community, graduate student community, and the undergraduate student community. The specific findings of this research are as follows:

- The faculty & staff felt importance and belonging and had a high sense of collective self-awareness both before and after the introduction of the information system.
- Initially, the graduate and undergraduate students did not feel important in the department but did feel a sense of belonging. Their collective self-awareness was poor.
- At the end of the study, graduate students did not feel important but still felt belonging in the department. However, their collective self-awareness seemed to increase.
- At the end of the study, undergraduate students neither felt importance nor felt belonging in the department. But, their collective self-awareness seemed to increase.

2.2.1 How the Andrada community research relates to this CLEW research

The CLEW research relates to Andrada's research on a meta-level. CLEW and Andrada's implementation of the department website both have the intention to support various aspects of the department. The following describe the similarities and differences between my CLEW research and Andrada's research.

Andrada's community building website and the CLEW Website provides some similar functionalities. The similarities are:

- Allows users of the website to post unedited items to the website.
- Allows users to post a link to there personal homepage
- Provides a mechanism to show what is new or changed on the website.

These functionalities allow the department to interact with the website and become active contributors to the actual content of the department website. However, there are probably differences between the actual functional workings of the two websites. The department website that Andrada built no longer exists, so, no detailed differences can be made.

CLEW research and Andrada's research differs in what the department websites supports. Currently, CLEW does not try to create a "sense of community," as Andrada defines it. CLEW makes the assumption that the department already has a sense of community and CLEW's job is to be able to support better communication and better modeling of the department through its website. However, the Andrada study validates my assumption that there are senses of community in a collegiate department.

Chapter 3. The CLEW System

3.1 History of CLEW

The development of the CLEW System has been an iterative process. This chapter illustrates CLEW's beginnings, the evolution of CLEW, and the various technical aspects to the CLEW System.

3.1.1 Individual Systems

The idea for CLEW was conceived by Professor Philip Johnson in the Spring 2002 semester as individual group projects for Software Engineering courses ICS 413 and ICS 613 at the University of Hawaii at Manoa The overall goal of these projects was to create a set of individual web applications that collectively provided a useful collegiate department website. The individual systems developed in the classes, for example, the Textbooks Exchange System, the Resume System, and the Technical Reports System were separate systems and were connected using open interfaces, or "service API's" (Application Interfaces – pictured in Figure 12). In Java web application terminology, there was a war file associated with each of the systems. This means that the session object could not be used to share information among the different systems, as do usual Java web applications. The Individual Systems were successful as individual systems; that is they were able to provide a usable, useful, and interactive services. However, there were some technical problems with the interaction. The system, as a whole, was only usable if these individual systems interact with each other using the application interfaces.

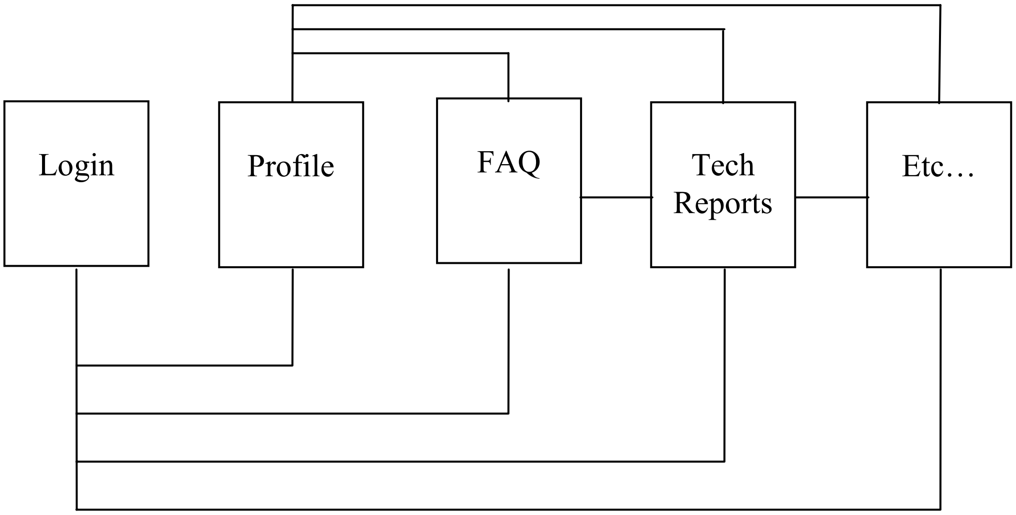


Figure 12 - The Individual System architecture
The boxes represent the individual systems (web applications, each containing their own war file). Each system implemented an application interface that allows them to share information, which is represented in this figure by the lines.

As shown in Figure 12, each system can be connected to other systems via the application interface. For example, every system that needs login capabilities needs to use the Login System. Another example is, if the Tech Report System needs to obtain information about a user who is logged in, the Tech Report System needs to access that information from the Profile System's application interface.

The primary problem in this architecture is creating, maintaining and implementing the application interfaces. The systems' application interfaces and the dependency of other systems' using these interfaces were tightly coupled. Very often changes on one system's application interface caused the other systems using it to fail.

Also, if an individual system became unavailable, due to compilation or runtime errors, all systems depending on the unavailable system's application interface also failed.

Another problem with this architecture was that it provided no standardization or constraints on how the interface was designed and implemented. Therefore, the individual systems suffered from an uncommon interface design.

The modularization which allowed each system to exist independently of each other caused major problems. Thus, by the end of the semester the individual systems as a whole were largely unusable.

3.1.2 CREST framework

To solve the problems of the previous semester, Jesse Tom and I created CREST during the Summer 2002 semester as an ICS 499 Computer Project under the guidance of Professor Philip Johnson. The goal of the project was to create a framework for the individual systems created in the Spring 2002 semester (Figure 13). The redesign created one war file crest.war, which contains all of the services. This provided a framework, which allows the individual systems, now called services, to interoperate more reliably and easily.

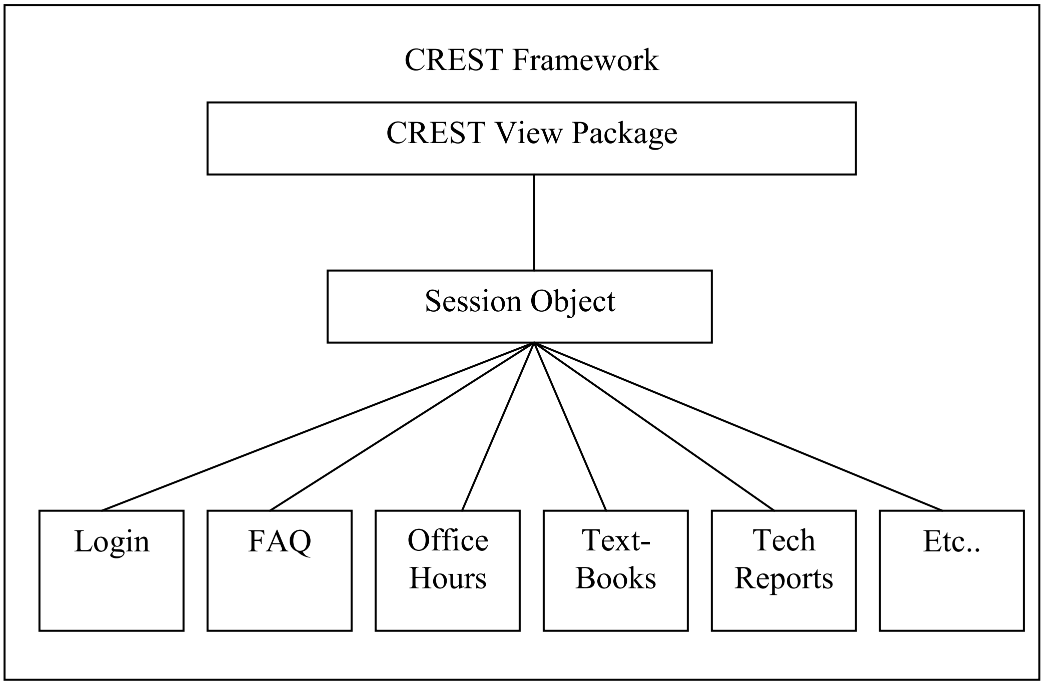


Figure 13 - The CREST framework
The outer large box represents the CREST framework. The smaller boxes represent the services that are included in CREST. CREST View Package provides the services with a mechanism to provide the services with a standardized interface. The Session Object is used to transfer data between different services.

CREST was designed to be a framework to provide its services the environment in which they can interact. These services are the "content" of CREST, meaning that the services are the interactive features that CREST relies on to interact with its users.

The main advantage of a single very large web application (CREST) as opposed to multiple small web applications (Individual Systems) is that the Java web application Servlet's Session object can be used to transfer information to all services, making the unsuccessful Individual System's application interfaces obsolete.

Another advantage that CREST provided was a standardization of the interfaces for all of the services. CREST provided a View Package that handled the display of every web page for all of the services. Therefore, interface design and implementation was no longer a requirement for the services.

In the Fall 2002 semester, CREST became the class project of ICS 414 taught by Professor Philip Johnson. Jesse Tom and I served as project managers for this project, overseeing all aspects of the project. The main goal of CREST was to create a better model of a collegiate department to increase the awareness of the department through a website. Groups of two to three students were created to work on some of the services in CREST. The ICS 414 class was able to create a FAQ, News Bulletin, Office Hours, Poll, Resume, Tech Reports, and Textbooks services under the new framework.

The CREST framework suffered from one major problem. The CREST framework required the services to be contained in the same Java web application, which caused the web application to become very large. The CREST system, which includes the CREST framework and the CREST services, contained over 31 thousand lines of code. To build CREST into one war file all thirty-on thousand lines of code must compile successfully. However, due to the collaborative aspect of this project, often a service would not compile successfully causing CREST to be completely unavailable for use.

3.1.3 CLEW Kernel architecture

In the Spring 2003 semester, the CREST framework got a complete makeover by first changing the name of the system and redesigning the framework. Since the name CREST was not a acronym, the name CLEW was created. CLEW stands for the **C**o**L**laborative **E**ducational **W**ebsite.

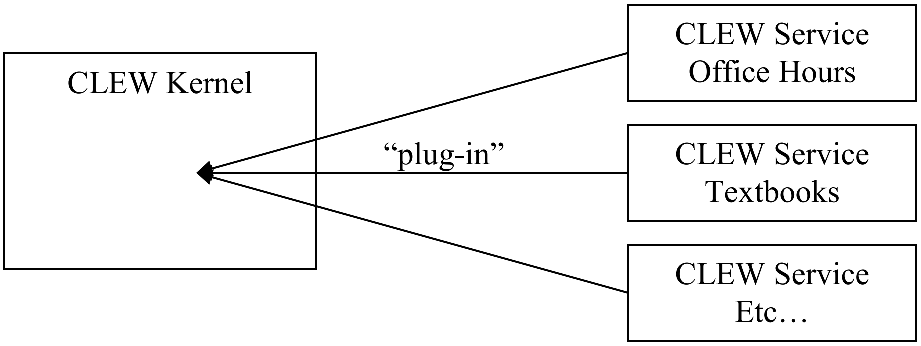


Figure 14 - The CLEW Kernel and CLEW Services framework
The CLEW Kernel provides an architecture that allows services to be plugged into it. The separation of the implementation of the services allows collaborative development.

It was my decision to redesign the existing CREST framework. The goal of this redesign was to create a "kernelized" architecture, which I name the CLEW Kernel. This allows the services (CLEW Services), to be developed outside of the system and then "plugged" into the CLEW System (Figure 14). This greatly improves the potential of CLEW to be developed in an open source collaborative environment. The introduction of the CLEW Kernel allows the CLEW Service to communicate in a specific way, thus solving the problem with the Individual Systems application interfaces. The development of CLEW has come full circle in that the problem with the Individual System's modularity, which was solved by CREST framework, now becomes CLEW's main goal.

3.2 Implementation of CLEW

The implementation of the CLEW System (CLEW Kernel and the existing CLEW Services) involves the Extreme Programming software engineering process and several software technologies, Apache Jakarta Ant, Apache Jakarta Struts, JUnit, HttpUnit, and JBlanket. The following section introduces the processes and technologies used in building the CLEW Kernel and CLEW Services.

3.2.1 Software Engineering Processes

Extreme Programming:

Extreme Programming is a specific software engineering discipline based on several principles: rapid feedback, assume simplicity, incremental change, embracing change, and quality work [5]. Based on these principles, a set of 12 practices was made to help guide software development using XP [6]:

- Planning Game
- Small Releases
- Metaphor
- Simple Design
- Write Tests First
- Refactor
- Pair Programming
- Collective Ownership
- Continuous Integration
- 40 Work Week
- On-Site Customer
- Coding Standards

Table 2 - The 12 XP practices

CLEW's implementation utilized a sub-set of the 12 XP practices: Small Releases, Simple Design, Refactor, Pair Programming, Collective Ownership, and Coding Standards. We also strongly utilized unit tests to improve the quality of the software.

3.2.2 Software Technologies

Apache Jakarta Ant:

Apache Jakarta Ant [7] is a Java-based build tool. It is very similar to the traditional Make build tool. However, Ant is entirely Java based with XML configuration, which allows cross-platform building. Ant allows developers to build and test the system without the use of an Interactive Development Environment (IDE).

CLEW utilizes Ant not only to build the CLEW Kernel and CLEW Services but also to "plug-in" the service to the Kernel. Detailed explanation is provided in Section 3.3: CLEW Build Process.

Apache Jakarta Struts:

Apache Jakarta Struts [8] is a framework to support Java Web Applications using the Model-View-Controller (MVC) design paradigm. The goal of Struts is to provide a standard implementation of the MVC design paradigm.

CLEW utilizes this framework to ensure that the implementation of CLEW conforms to the standard implementations of Java Web Applications. Also, Struts provides the necessary tools for CLEW to function. Further information is provided in Section 3.5: CLEW's use of the Struts Framework.

JUnit / HttpUnit / JBlanket:

JUnit [9] is Java framework used to write and run repeatable unit tests on Java code. HttpUnit [10] is Java framework used to write and run repeatable unit tests on Java Web Applications. JBlanket [11] is a tool that is used with JUnit and HttpUnit for assessing and improving method-level coverage of unit test cases. JUnit and HttpUnit were used to check the quality of the code in CLEW by using test cases and JBlanket was used to check the quality of the test cases. Further information is provided in Section 3.6: CLEW's use of JUnit, HttpUnit, and JBlanket.

3.2.3 CLEW Kernel Implementation

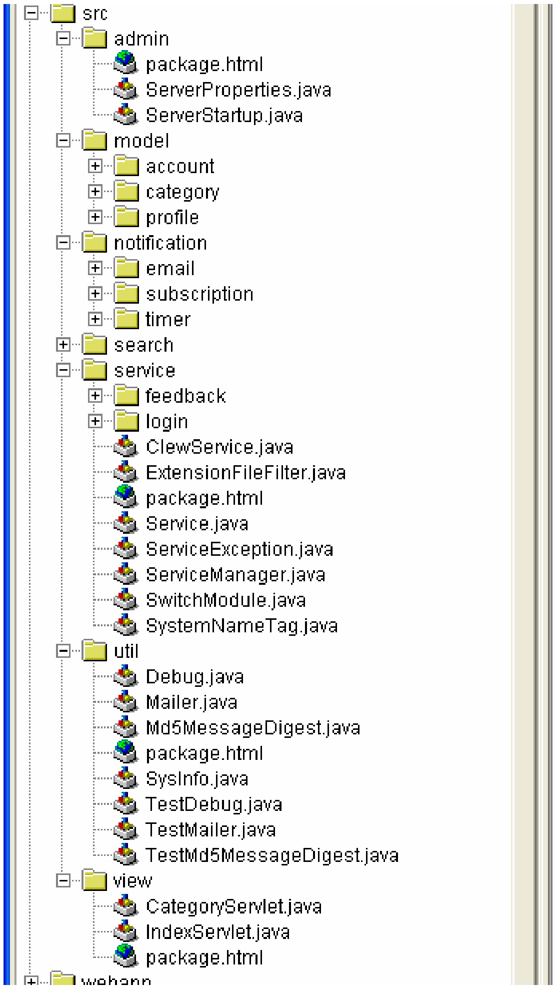
The CLEW Kernel is designed to support easy extendibility of CLEW. It provides the services with "tools" that are needed to operate in CLEW. The following section explains the various Java packages that are included in the CLEW Kernel.

3.2.3.1 CLEW Kernel Java Packages

The CLEW Kernel consists of six main Java packages:

- edu.hawaii.clew.admin
- edu.hawaii.clew.model
- edu.hawaii.notification
- edu.hawaii.clew.service
- edu.hawaii.util
- edu.hawaii.view

Each package contains Java classes that the services can use to plug-into CLEW. The following is an explanation of the uses of each of the six main packages.



```
src
  admin
    package.html
    ServerProperties.java
    ServerStartup.java
  model
    account
    category
    profile
  notification
    email
    subscription
    timer
  search
  service
    feedback
    login
    ClewService.java
    ExtensionFileFilter.java
    package.html
    Service.java
    ServiceException.java
    ServiceManager.java
    SwitchModule.java
    SystemNameTag.java
  util
    Debug.java
    Mailer.java
    Md5MessageDigest.java
    package.html
    SysInfo.java
    TestDebug.java
    TestMailer.java
    TestMd5MessageDigest.java
  view
    CategoryServlet.java
    IndexServlet.java
    package.html
webapp
```

Figure 15 - The CLEW Kernel package structure

This figure aims at providing a glimpse at the package structure of the CLEW Kernel. Each package contains functionality that the CLEW Services can utilize to create services in the CLEW System. This package structure is presented in the JBuilder IDE [12], which is the standard IDE for CLEW development.

3.2.3.2 Package edu.hawaii.clew.admin

This package provides administrative configurable access classes. For example, the ServerProperties class contains the values of server specific information that the services can use. The ServerStartup class allows CLEW to initialize objects that it needs on the server in order to function correctly.

3.2.3.3 Package edu.hawaii.clew.model

This package contains the representations of the various models that the kernel and possible services rely on. For example, this package contains the model of a user's account which allows CLEW to store information about users. Each model contains a package called “persistent”, which utilizes Jdom to persist the model’s information to an XML file.

3.2.3.4 Package edu.hawaii.clew.notification

This package implements the email notifications and subscriptions of CLEW. It provides abstract classes, which the services must use to utilize email notifications. The services must simply instantiate (or extend when necessary) the CLEW Kernel classes provided in this package to provide email notifications.

3.2.3.5 Package edu.hawaii.clew.service

This package serves two purposes. First, several classes in the package serve as a “hook” into the CLEW Kernel, which notifies the kernel that it has been extended and services need to be “plugged-in”. The CLEW Kernel then allows the service to be “plugged-in”. Other sub-packages provide default services that the kernel provides. For example, the CLEW Kernel comes with a default Login Service, which can be used by all services that need this facility. However, this default Login Service can be overridden by another login service, if needed. As an example, if LDAP is available, a service could be written to provide LDAP user authentication.

3.2.3.6 Package edu.hawaii.clew.util

This package contains classes that CLEW uses to do various utility work and may be used by any service. For example, a Mailer class is provided to process and send emails by either the CLEW Kernel or CLEW Services.

3.2.3.7 Package edu.hawaii.clew.view

This package is a mechanism to provide the CLEW Kernel and CLEW Services with a standardized web interface.

3.2.4 CLEW Service Implementation

The CLEW Services utilize and extend the CLEW Kernel to create interactive services like the Tech Reports service. The following section explains the various Java packages that are included in the CLEW Service.

3.2.4.1 CLEW Service Java Packages

The CLEW Service consists of six main Java packages:

- edu.hawaii.clewx.<service>.admin
- edu.hawaii.clewx.<service>.controller
- edu.hawaii.clewx.<service>.model
- edu.hawaii.clewx.<service>.notification
- edu.hawaii.clewx.<service>.util
- edu.hawaii.clewx.<service>.view

Note the string "<service>" represents a generalized naming convention that can be replaced by any specific name of the CLEW Service. The following is an explanation of the uses of each of the six main packages. Also, note that this package structure is the recommended package structure but, the individual CLEW Services can implement their service in whatever structure they choose.

3.2.4.2 Package edu.hawaii.clew.clewx.<service>.admin

This package contains a very important class called ServerStartup. This class is essential for the CLEW Service to be "plugged-in" to the CLEW Kernel. Contained in the ServerStartup are "hooks" into the CLEW Kernel specifying what the Kernel needs to do for the service. The CLEW Kernel using Java Reflection which is managed by the edu.hawaii.clew.service.ServiceManager class in the CLEW Kernel calls this class dynamically.

3.2.4.3 Package edu.hawaii.clew.clewx.<service>.controller

This package processes the users' interactive requests through the web interface of the CLEW Services. Each type of request that can be taken by a user for a specific CLEW Service must be implemented in this package. The specific implementation of the controller package relies on the Apache Jakarta Struts framework.

3.2.4.4 Package edu.hawaii.clew.clewx.<service>.model

This package contains the representations of the objects that the CLEW Service relies on to function as a service. For example, the CLEW Service Textbooks Service's model package contains Java classes that model how a textbook is manipulated in the service. Additionally, each model contains a package called "persistent" which utilizes Jdom to persist the model's information to an XML file.

3.2.4.5 Package edu.hawaii.clew.clewx.<service>.notification

This package extends the CLEW Kernel’s implementation of the notifications in the edu.hawaii.clew.notification package. The service’s job is to provide the CLEW Kernel with the specific implementation that the service wishes to use in the notification process. However, most of the processing occurs in the CLEW Kernel’s notification package which allows the services to easily take advantage of CLEW’s email notification feature.

3.2.4.6 Package edu.hawaii.clew.clewx.<service>.util

This package contains utility classes that the CLEW Service uses to do various specific processing.

3.2.4.7 Package edu.hawaii.clew.clewx.<service>.view

This package contains the view component of the Model-View-Controller design paradigm. The view components are implemented using Apache Jakarta Struts framework.

3.2.5 Results of the new CLEW architecture

The CLEW architecture has one major advantage of the architecture of the previous CREST version. It has reduced the amount of code needed to implement the CLEW System. This was accomplished by creating a “kernelized” architecture and having the CLEW Kernel provide functionality that the CLEW Services can use instead of implementing themselves.

The CLEW "kernelized" architecture reduced the total lines of code in the system by 7.7 percent, a drop from 31,275 to 28,870 lines of code. Most of the reduction of the total lines of code came from a major reduction in test case lines of code (which are the lines of code that are needed to implement unit test cases). The CLEW architecture was able to create a 21.4 percent decrease in the amount test case lines of code, while at the same time not affecting the method-level coverage calculated by JBlanket [11]. This was achieved by providing a mechanism in the CLEW Kernel to support the test cases of the CLEW Services. Also, by adding more functionality to the CLEW Kernel, the CLEW Services were able to use the centralized functionality and reduce the amount of code by 12.4 percent, down from an average of 3,847 to 3,432 lines of code.

The reduction of code in the CLEW architecture allows the CLEW Services to be implemented with simpler code. Also, the CLEW architecture supports easier unit test case creation.

3.3 CLEW build process

The CLEW Kernel utilizes Apache Jakarta Ant [7] for its build process. The build process for this system breaks down into two specific build processes, the kernel build process and service build process.

The kernel build process has the basic targets of a standard web application and special mechanisms to "plug-in" CLEW Services. The first step that the CLEW Kernel takes to provide a "plug-in" type build process for services is to specify a directory structure. The services build process places their files into this directory structure. The kernel build process then incorporates the services' files into the build of the CLEW Kernel.

The build processes for both the CLEW Kernel and CLEW Services are relatively simple. The following Code Snippet 1 and Code Snippet 2, provides an example of the "plug-in" type build process that the CLEW Kernel and CLEW Services use.

```
<!-- ******************************************************************* -->
<target name="plugin" depends="package"
 description="Puts CLEW Service into the Clew Kernel.">

  <!-- Copy the html and jsp over with filtering. -->
  <copy todir="${kernel.ext.webapp.dir}/textbooks" filtering="yes">
    <fileset dir="${webapp.dir}">
      <include name="*.html"/>
      <include name="*.jsp"/>
    </fileset>
  </copy>

  <copy todir="${kernel.ext.web-inf.dir}"
      file="${web-inf.dir}/struts-textbooks-config.xml" />
  <copy todir="${kernel.ext.lib.dir}"
      file="${ant.lib.dir}/textbooks.jar" />
  <copy todir="${kernel.ext.classes.dir}"
      file="${classes.dir}/textbooks_ApplicationResources.properties" />
  <copy todir="${kernel.ext.service.dir}"
      file="${service.dir}/service.textbooks.xml" />
  <copy todir="${kernel.ext.persistent.dir}/textbooks"
      file="${persistent.dir}/textbooks/textbooks.xml" />
  <copy todir="${kernel.ext.persistent.dir}/textbooks"
      file="${persistent.dir}/textbooks/textbooksSubscriptions.xml" />

</target>
```

Code Snippet 1 – An Example "plugin" build target for a CLEW Service. This code snippet illustrates how a CLEW Service "plugs-in" to the CLEW Kernel after the CLEW Service has been properly compiled and packaged. The "kernel.ext.<something>.dir" properties contain the paths to the CLEW Kernel's special service "plug-in" directories where all services must place their respective files before the CLEW Kernel build process.

```
<!-- ****************************************************************** -->
<target name="war" depends="package"
 description="Creates the clew war file.">

...

  <!-- Copy the html and jsp over with filtering. -->
  <copy todir="${build.webapp}" filtering="yes">
    <fileset dir="${ext.webapp}">
      <include name="**/*.html"/>
      <include name="**/*.jsp"/>
    </fileset>
   </copy>

  <!-- Now copy everything else without filtering. -->
  <copy todir="${build.webapp}" >
    <fileset dir="${ext.webapp}">
      <exclude name="**/*.html"/>
      <exclude name="**/*.jsp"/>
    </fileset>
  </copy>

 <war warfile="${lib.dir}/${name}.war" webxml="${webapp.dir}/WEB-INF/web.xml">
   <fileset dir="${build.webapp}">
     <exclude name="WEB-INF/**"/>
   </fileset>
   <webinf dir="${build.webapp}/WEB-INF" >
      <exclude name="web.xml"/>
   </webinf>
 </war>
</target>
```

Code Snippet 2 – The "war" target for the CLEW Kernel. This code snippet illustrates how the CLEW Kernel incorporates the CLEW Services into the CLEW System, by simply copying the service's files into the CLEW Kernel before creating the clew.war file.

The build processes for both the CLEW Kernel and the CLEW Services have been greatly simplified from the previous version of the CREST build process. The main advantage that the CLEW build process provides is that the CLEW Services will only be integrated into the CLEW System, if and only if, the CLEW Service compiles and packages correctly. Therefore, this process eliminates faulty services at build time.

3.4 CLEW Interface

The CLEW Interface is designed using the same interface design issues of the original University of Hawaii at Manoa Department of Information and Computer Science website. This is a very important design choice. The intention is to be able to evaluate CLEW not based on the usability, usefulness, and other aspects associated with the normal evaluation of websites, but rather to evaluate the added value of the services and functionality that CLEW provides. However, the HCI issues of usability are not ignored in the actual design of the CLEW Interface. When new interface design issues that were not faced in the previous version of the UH ICS department website arose, we dealt with them with our best knowledge of usability, usefulness, and interactivity (all issues associated with interface design in HCI). As a whole the CLEW Interface is very similar to the original UH ICS department website. Additionally, the CLEW Interface is designed to be independent of CLEW implementation. Therefore, many interfaces can work with CLEW.

The CLEW Interface is implemented using Java Server Pages (JSP) [14], JavaServer Pages Standard Tag Library (JSTL) [14], Apache Jakarta Struts Tiles [8], and Apache Jakarta Struts Tag Library [8]. The key technology is the Apache Jakarta Struts Tiles, which provides templates that allow a standardized look and feel for all CLEW's web pages. This technology allows the CLEW Kernel to provide a header (Figure 16) and sidebar section (Figure 17) to all web pages in the CLEW System. Tiles utilizes these two JSP files to create a "template" that CLEW Service can use to adhere to CLEW's "look and feel." An example of a CLEW Service utilizing CLEW's interface template is provided in Code Snippet 3.

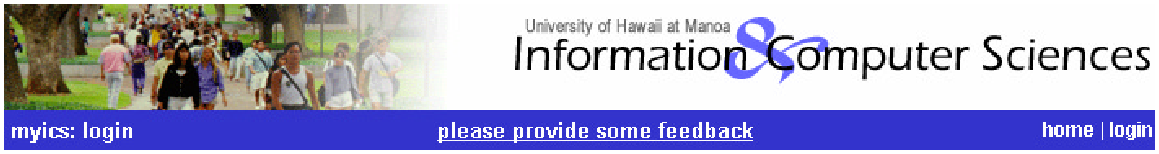

University of Hawaii at Manoa
Information & Computer Sciences
myics: login
please provide some feedback
home | login

Figure 16 - The CLEW Interface header

The header is a single JSP file that contains JSTL tags that allows the CLEW Kernel to provide the necessary information for all web pages included in CLEW to use this header.

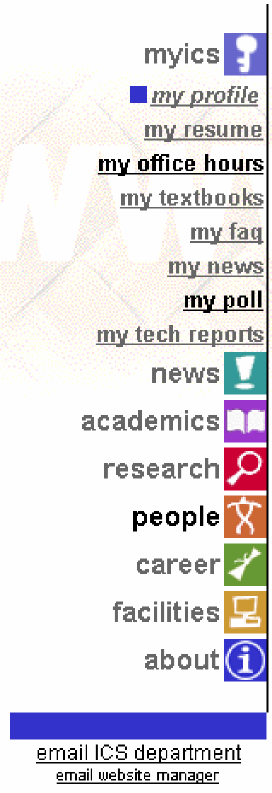


Figure 17 - The CLEW Interface sidebar

The sidebar is a single JSP file that contains JSTL tags that allows the CLEW Kernel to provide the necessary information for all web pages included in CLEW to use this sidebar. The exciting aspect of the sidebar is that it is generated dynamically by the CLEW Kernel, which allows the CLEW Services to place links to the service specific web pages.

```
<%@ taglib uri='/WEB-INF/struts-tiles.tld' prefix='tiles' %>
<%@ taglib uri='/WEB-INF/struts-bean.tld' prefix='bean' %>
...
<head>
<title><bean:message key="textbooks.page.index.title"/></title>
...
</head>
...
<tiles:insert page="/layouts/icsTileTemplate.jsp">
    <tiles:put name="header"  value="/header.jsp" />
    <tiles:put name="foot"    value="/footer.jsp" />
    <tiles:put name="sidebar" value="/sidebar.jsp" />
    <tiles:put name="content" value="/textbooks/indexContent.jsp" />
</tiles:insert>
```

Code Snippet 3 – Example use of the CLEW template. This code snippet illustrates the CLEW Textbook Service's use of the CLEW template implement using Apache Jakarta Struts Tiles. The template allows CLEW Kernel to manage the header, sidebar, and footer of the web pages, while the CLEW Textbooks Service only manages the /textbooks/indexContent.jsp web page.

3.5 CLEW's use of the Struts Framework

The Apache Jakarta Struts framework is the backbone of the CLEW Kernel and CLEW Services. Struts [8] is an open source framework designed to support the building of Java web applications using, Java Servlets [13] and JavaServer Pages (JSP) [14] technology. Struts is used as a standardizing implementation of the Model 2 or Model-View-Controller design paradigm.

CLEW relies on Struts to standardize the implementation of the Model-View-Controller design paradigm and without this framework CLEW cannot guarantee that the CLEW Services will be compatible with the CLEW Kernel.

CLEW utilizes the Struts implementation of Application Modules which create modularization within web applications. Application Modules enable developers to create web applications, which have specific duties, independent of other Application Modules. The Struts framework allows each Application Module to be added into a larger web application. This feature of Struts is ideal for implementing the CLEW Services. Struts allows CLEW Services to be developed entirely outside of the CLEW Kernel and when completed the service can be "plugged into" the CLEW Kernel for deployment.

3.6 CLEW's use of JUint, HttpUnit, and JBlanket

JUnit [9], HttpUnit [10], and JBlanket [11] are an integral part of the CLEW System. The use of these tools ensures a high level of software quality. This section illustrates how CLEW uses these software quality tools.

JUnit and HttpUnit are used in CLEW to create unit test cases for the various parts of the CLEW System. The use of unit test cases ensures that the various aspects of CLEW behave as the developers intend. JUnit is used to ensure that the Model classes (from Model-View-Controller) work correctly. HttpUnit is used to ensure that the View and Controller classes work correctly with the corresponding JSP web pages. The terms, "working correctly," generally has the definition that the system behaves appropriately with a variety of input values.

The software development process of CLEW demands that all unit test cases pass. Using tools such as these, unit tests are built into the build process of CLEW so that the unit tests are run on each build of the system. This ensures that the developers will be aware of software quality problems and that they can be addressed at "build time" as opposed to "run or deployment time." Currently in the CLEW System, there are 289 separate unit tests all of which pass.

JBlanket is a tool that is used with JUnit and HttpUnit for assessing and improving method-level coverage of unit test cases. Essentially, JBlanket ensures that JUnit and HttpUnit are being used correctly and that the unit test cases are effectively testing the system.

The software development process of CLEW aims to achieve 100% coverage of all methods in the system. The goal is to effectively test every method in the system by JUnit and HttpUnit unit tests. Currently, the CLEW System has unit tests that cover 1,093 methods of the possible 1,203 testable methods, giving 90.9 % test coverage.

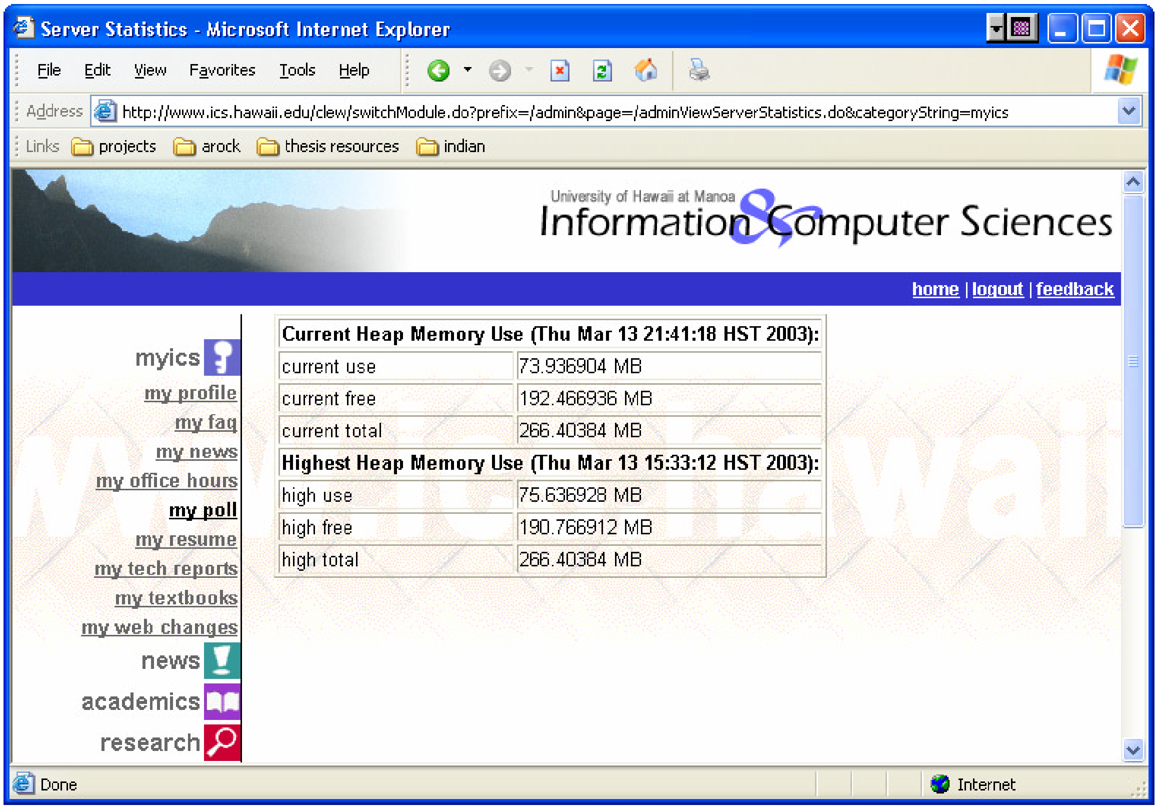
These three tools and the software development process allow the monitoring of software quality of the CLEW System. Using them has caught thousands of bugs and problems at "build time."

3.7 CLEW heap memory management

Upon the initial deployment of the CLEW System for use by faculty and staff of the UH ICS department the CLEW System ran out of heap memory in a period of 24 hours. Heap memory was never a problem until this point of development. Therefore, the CLEW System had no mechanisms to fix or monitor the memory situation. This presented a potentially major problem which could jeopardize the release of the system to the department.

The initial diagnosis of the problem was that the Java code must be allocating memory, reserving space, and never being reclaimed by Java's garbage collection process. A tool called JProfiler [16], which is tool to find threading issues, memory leaks, and performance bottlenecks in Java code, was used to profile the CLEW System running on the Apache Tomcat Server [15]. The initial results of the profiling provided no indication of where the memory leaks were located, thus the out of memory problem could not be solved by profiling.

The next solution was to find the exact size of the heap during operation on the server. A CLEW Administrator Service was created to allow users with the administrator role to view the size current heap memory that is used, free, and the total. Also, this service provides the largest heap size that has occurred during the time period that the server was running. This allows administrators to monitor the current heap size and the times when heap size is the largest in order to understand the movement of the heap size. The information that is presented by using this service is pictured in Figure 18.



Current Heap Memory Use (Thu Mar 13 21:41:18 HST 2003):	
current use	73.936904 MB
current free	192.466936 MB
current total	266.40384 MB
Highest Heap Memory Use (Thu Mar 13 15:33:12 HST 2003):	
high use	75.636928 MB
high free	190.766912 MB
high total	266.40384 MB

Figure 18 - The CLEW Administrator Service - Heap Monitoring

This presents a screen shot of how an administrator can monitor the heap memory size.

With this service and the web logs that are created by Apache Tomcat, we can determine what events took place, if an out of memory exception occurs again. Using this service, the CLEW System was load tested and stress tested while monitoring the heap size. The findings were that the system did not have a memory leak. Rather the initial and maximum sizes of the heap memory were set too small. In Apache Tomcat, there are configuration files which let developers set the initial size and maximum size of the heap memory. Using this configuration, the heap memory was changed to use 256 megabytes from its default of 64 megabytes. This solved the problem. Average values of the heap memory are 70 megabytes. It is obvious that the default value of 64 megabytes is not enough storage for a Java web application of this size.

Using the CLEW Administrator Service it would be possible to email users with the administrative role when a heap memory value above a certain threshold occurs; then the administrators can take the appropriate actions. This threshold has not been determined yet. Also, a valuable addition to this service would be to provide the viewing of events that have recently occurred and the events that have occurred during the highest value of the heap size.

3.8 CLEW Developers

The CLEW System would not have been possible without the many developers that contributed to the system. The developers came from multiple software engineering computer science classes in the UH ICS department. Appendix D gives recognition to all the developers that have contributed to the Individual Systems, CREST System, and the CLEW System.

Chapter 4. Evaluation of CLEW

I have evaluated CLEW to provide evidence regarding the four claims in my thesis statement. There are three main evaluations: an evaluation of the CLEW collegiate department website model, an evaluation of user feedback about the CLEW Website, and an evaluation of how the developers feel about implementing the CLEW Services. This chapter presents the methodologies I used in these evaluations.

4.1 CLEW Model evaluations

The first assessment of CLEW focuses on whether it provides an accurate and useful model of a collegiate department. To conduct this evaluation, CLEW replaced the original University of Hawaii at Manoa's Information and Computer Science department website in March of 2003. Evaluations took place in the following three areas illustrated in Figure 19.

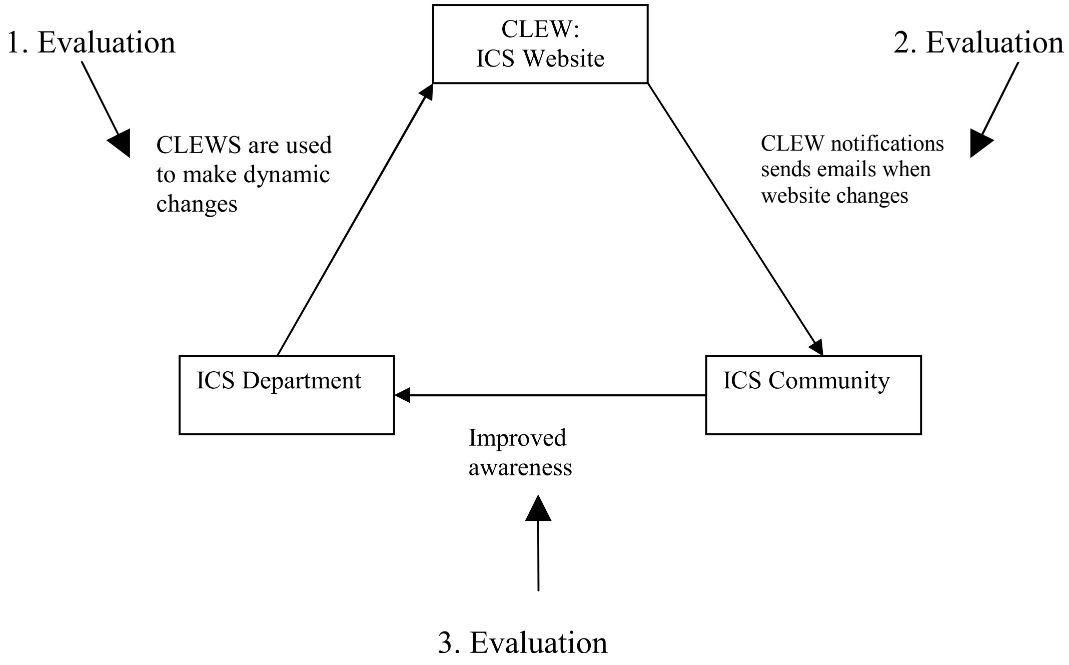


Figure 19 - Evaluations of the CLEW Model

The CLEW Model evaluations occurred in the three key areas of the CLEW Model: 1) CLEW Services, 2) CLEW Email Notifications, and 3) the improved awareness of the department. The following sections describe each evaluation in more detail.

4.1.1 CLEW Model evaluations: CLEW Services

CLEW Services play an important role in providing dynamic changes to the website. If the CLEW Services are used often to add dynamic content to the department website, then it will provide evidence that these services are useful in helping to model a collegiate department and that the department successfully adopts them. The original UH ICS department website had several services of its own, including: a resume listing, people listing and a news listing. Likewise, CLEW contains new versions of these services. I have conducted an evaluation of the advantages and disadvantages between the two versions of the services.

The methodology used in this evaluation compares the amount of interactions the users have had with these services, by measuring the number of “postings” on the website. This evaluation will indicate whether or not CLEW improves the ability of the students to dynamically make changes to the website. This evaluation provides evidence for Claim 1 of my thesis statement.

Specifically, a calculation of the number of postings over the allotted evaluation period gives an average value of postings per week. An example calculation of the value of postings/weeks is presented below.

If 60 people posted their resumes on the current UH ICS department website and period of operation of the resume service is 20 weeks. That yields the following equation and value: 60 / 20 weeks = 3 postings per week.

On November 12, 2002 the exact number of the postings of the resume, people, and news listing were collected from the original UH ICS department website. Then, on February 12, 2003, the exact numbers of postings were collected from the same listings on the same website. The evaluation period was 12 weeks, November 12, 2002 to February 12, 2003.

On March 4, 2003 the CLEW System replaced the original UH ICS department website. The initial values of the postings was zero at that point. Then, on March 31, 2003 the exact number of postings were collected from the CLEW Resume, People, and

News Bulletin Service. The evaluation period was 4 weeks, March 4, 2003 to March 31, 2003.

4.1.2 CLEW Model evaluations: CLEW Email Notification

CLEW Email Notification plays an important role in alerting the community of changes that occurred in the website. This feature allows the community to play a passive role in becoming aware of the changes in a department. This is exactly the opposite of the traditional department website were active searching is needed to become aware of changes in the department website. Initially, the CLEW System automatically subscribed CLEW's users to all email notifications. However, they may have chosen to unsubscribe from one or more email notifications if they felt that the notifications did not provide useful information to them regarding changes to the department. Also, users might have wanted to unsubscribe if they felt they did not care about how the department is changing. An evaluation of the number of subscribed users was conducted to determine the usefulness and importance of CLEW Email Notifications.

Each registered user's account information is stored in an XML file. The following is an example of the CLEW Textbook Service's subscription XML:

```
<subscription>
  <userName>kagawaa@hawaii.edu</userName>
  <items>
    <item>ICS 413</item>
    <item>ICS 414</item>
    <item>ICS 451</item>
    <item>ICS 463</item>
    <item>ICS 491</item>
  </items>
</subscription>
```

This is an example subscription XML in which the user can “subscribe” to be notified of a posting of a book on a specific category (which is represented in the XML as an “item”) on the CLEW Website. This allows for an easy count of the total number of registered users. Every CLEW Service manages its own subscriptions and has their own XML file to store the subscribed users’ information.

To provide evidence regarding Claim 2 of my thesis statement, I compared the numbers of total registered users with the number of the users who receive email notifications to determine if CLEW’s users find the notification process helpful.

4.1.3 CLEW Model evaluations: Improved awareness

The operational definition of awareness that this research is concerned with is the qualitative responses of CLEW users regarding their awareness of the department through the CLEW Website. Anonymous questionnaires measure this awareness using a pre-release and post-release questionnaire method. The following is an explanation of the questionnaires.

The pre-release questionnaire contains questions that attempts to describe the awareness that the original UH ICS Department website provides to students. Section 1.3 describes various issues with the original UH ICS department website. The goal of this questionnaire was to find the level of awareness and the level of usage that students have of the UH ICS department website. The level of awareness and usage are measured on a point scale from 1 through 5. The value 1 represents responses of strong awareness and high usage. 5 represents the opposite response, low awareness and low usage. The pre-release questionnaire is presented in Appendix A.

The post-release questionnaire contains questions concerning the CLEW version of the UH ICS Department website. This questionnaire is very similar in nature to the pre-release questionnaire, which enables me to compare and contrast the differences in the users' awareness and usage of the CLEW Website. In addition the questionnaire will try to evaluate the new services that have been implemented with CLEW. However, some of the CLEW Services replace the services in the original UH ICS department website. Thus, some of the questions of the pre-release questionnaire apply to the post-release questionnaire. The post-release questionnaire is presented in Appendix B.

The questionnaires were administered in a single UH ICS class; ICS 313 Spring 2003 Section 1. This class is an intermediate level ICS class with 40 registered students. The pre-release questionnaire was administered on February 26, 2003 while the original UH ICS department website was still in use. The CLEW UH ICS department website replaced the original UH ICS department website on March 3, 2003. The post-release questionnaire was administered on March 31, 2003, approximately four weeks after the CLEW Website became operational.

4.2 CLEW Feedback evaluations

Another source of evaluation data comes from the CLEW Feedback Service. This service provides a simple form for users to fill out with feedback about the system.

The users' feedback was gathered and interpreted. The interpretation of the feedback includes categorizing the feedback based on its content. I have created two categories of feedback: (1) functionality feedback and (2) usability feedback. The goal of this categorization is to find what areas (functionality and usability) the users are concerned with. The methodology of this evaluation will include comparing and contrasting the number of the feedbacks in these two categories.

Table 3 presents a matrix that I will use to organize these feedbacks.

	Usability Feedbacks	**Functionality Feedbacks**
CLEW System		
FAQ		
Feedback		
Login		
News Bulletin		
Office Hours		
Poll		
Resume		
Textbooks		
Webedit		
All Services		
Static Pages		
	Total:	Total:

Table 3 - Feedback evaluation matrix

By gathering the feedbacks and categorizing them into the matrix, presented in Table 3, I was able to evaluate the issues that CLEW's users are concerned with. This information provides evidence for Claim 1 and Claim 2 of my thesis statement.

4.3 Developer evaluation

One of CLEW's design goals is to provide a framework for CLEW Service development so they can easily be "plugged-in" to the system. There are two motivations for this framework design. First, to allow CLEW to be easily adopted by other collegiate departments. The second motivation is to serve as an educational goal as a "real world" application for future development and enhancement by other Software Engineering classes.

The goal of this evaluation is to learn how developers of the CLEW Services feel about the system at different levels. First, how easy is it to create a service for the CLEW System? Second, has working on the services allowed the developers to learn software processes and software technologies? The developer questionnaire is presented in Appendix C.

This evaluation differs from the pre and post questionnaire methodology of the awareness questionnaire, due to the lack of opportunity to conduct a pre questionnaire. I administered the questionnaire to 7 of the CLEW Service developers. Their responses will provide evidence regarding Claim 4 of my thesis statement.

Chapter 5. Results

A “soft release” of the CLEW System occurred on March 4, 2003, to the faculty and staff members of the UH ICS department to allow these members to become familiar with its functionality. This initial release also allowed the faculty members to register for an account and post their office hours to prepare the website for department wide deployment. On March 13, 2003, the CLEW System officially replaced the UH ICS department website. The CLEW System currently resides on http://www.ics.hawaii.edu.

This chapter describes the results of the evaluations that were explained in Chapter 4.

5.1 Results of the CLEW Model evaluations

The results of the CLEW collegiate department website model evaluations provide evidence that CLEW improves the ability to dynamically change the department website, CLEW can notify CLEW’s users of changes in the department website by creating a notification system, and CLEW can improve the community’s awareness of the department.

5.1.1 Results of the CLEW Model evaluations: CLEW Services

To compare CLEW Services to the prior ICS Services, I compared the number of “postings” in the resume, people, and news listings of the original UH ICS department website to the “postings” in CLEW’s versions of those services. Table 4 provides the number of postings during the original UH ICS department website and CLEW UH ICS department website evaluations.

Service	**Original website (11/12/02 – 2/12/03)**	**Original website (total life)**	**CLEW website (3/4/03 – 3/31/03)**
People – Student	0	66	90
People – Alumni	1	28	20
Resume	0	80	18
News	0	32	1

Table 4 - The number of postings during evaluation periods.

The numbers of postings to the resume, people, and news listing were collected from the original UH ICS department website from November 12, 2002 to February 12, 2003. The allotted evaluation period was 12 weeks. Table 4 provides the number of postings to the original website during this evaluation period.

The original website's low number of postings indicates a problem with its ability to allow interaction with the website. This problem maybe due to two factors. First, the services were not explicitly advertised to the users, therefore, most users did not know of its existence. Second, the services did not provide an obvious benefit to the users, so they did not bother to use them.

The original UH ICS department website evaluation of the postings in the resume, people, and news listing during the evaluation period of November 12, 2002 to February 12, 2003 has not provided a good view of how the department community interacted with the website. Therefore, the total numbers of the postings during the total life of the original UH ICS department website is presented in Table 4. The number of postings during the total life of the original UH ICS department website provides evidence that the original website did support a large amount of interaction. However, comparing the total number of the original UH ICS department website postings to the 845 declared major students and around 1,200 alumni that currently are associated with the UH ICS department show that only a very small percentage of the community interacted with the department website.

On March 31, 2003 the exact numbers of “postings” to the resume, people and news listings were collected from the CLEW UH ICS department website. During the time period from March 4, 2003 to March 31, 2003 the number of postings to CLEW is presented in Table 4.

An immediate observation of Table 4, is that the number of postings in the student listing has already, in 4 weeks, surpassed the total number that were generated in the original UH ICS department website that was in operation for 116 weeks, (2 years and 5 months). Also, the number of postings in the alumni listing is very close to the number generated by the original website. However, there have been technical problems with supporting alumni listings in the CLEW System. For example, only alumni having a valid UH ITS account are able to register to the CLEW System and provide their information to the alumni listing. It has already been reported that at least one alumnus does not have a valid ITS account and was not able to register. Thus, the potential adoption of this service is hindered by this problem.

Using the formula to calculate the average value postings per week (postings / weeks) we have the following values:

Service	Average Postings per Week
People – Student	17.25
People – Alumni	5
Resume	3.75
News	0.25

Table 5 - The average number of postings per week on the CLEW UH ICS department website during March 4, 2003 to March 31, 2003

The averages of the postings per week presented in Table 5 are outstanding. If the posting continues with these averages, then this would provide undisputable evidence that CLEW allows dynamic changes to the website. Also, it supports that the UH ICS department successfully adopts the CLEW System.

There is one serious problem that this methodology does not account for. The average values of postings per week could be overly high, due to the novelty of the CLEW Services. It could be possible that the interactions with the CLEW System are driven by curiosity. After the novelty of this system declines, then the number of postings will decline also. Further monitoring of the systems postings over a longer period of time is needed.

The CLEW News Bulletin Service has been used to post only one news bulletin. This service allows faculty members and administrators to post news articles to the website. The very low use of this service indicates three possibilities, the department has no news articles to post, the faculty and administrators have not seen the advantage of the news bulletin, or that the news bulletin does not provide a usable and useful service. However, until this service is investigated more thoroughly, no determination can be made about its utility to interact with the department website and its adoption by the department community.

Regardless of the previous problems, the number of postings using the CLEW Services provides evidence that CLEW can allow dynamic changes to the website; supporting Claim 1 of my thesis statement. Also, this provides evidence that the UH ICS department community can adopt the use of these CLEW Services.

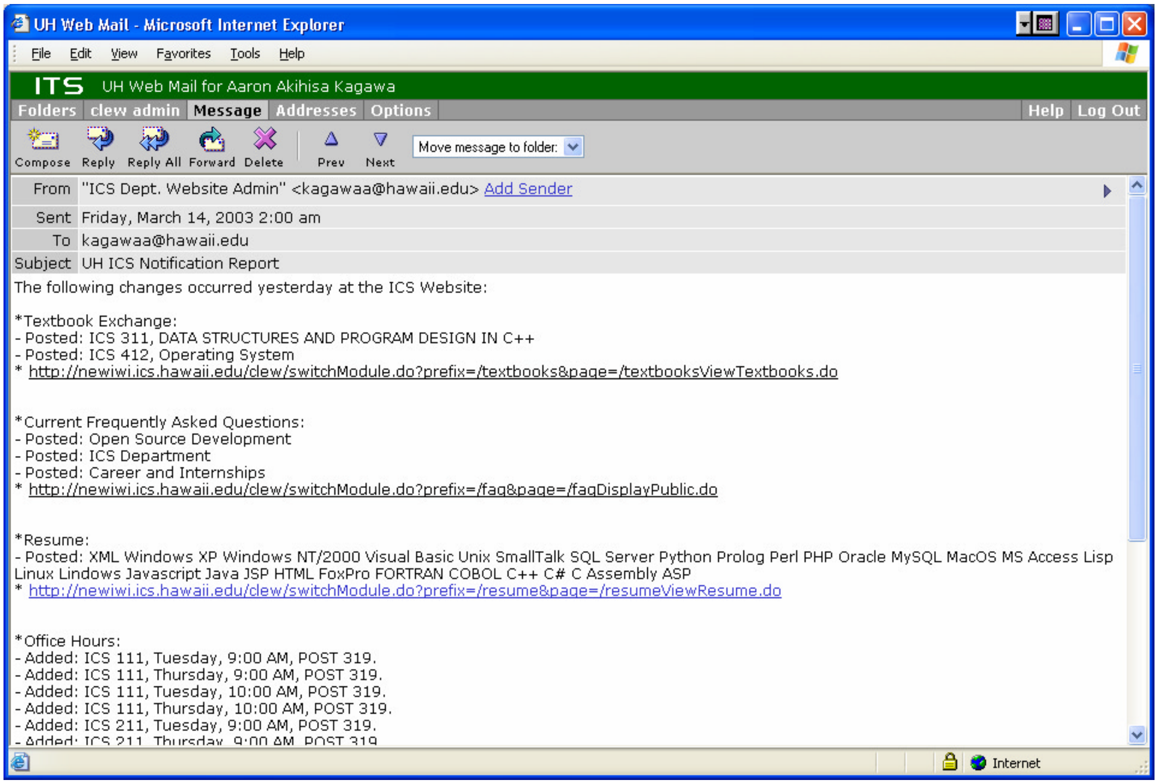
5.1.2 Results of the CLEW Model evaluations: CLEW Email Notification

To evaluate the CLEW Email Notifications, I compared the numbers of total registered users to the number of the users who receive email notifications. This evaluation determines if CLEW's users find the notification process helpful.

The current implementation of the CLEW Email Notification automatically subscribes the users of CLEW to all services' email notifications. This is an "opt-out" (as opposed to an "opt-in) subscription strategy. Unfortunately, several of CLEW's users have considered this as a "spam," type tactic. However, I felt an "opt-out" strategy was the best way to evaluate email notifications. If I left the decision to subscribe to email notifications up to the CLEW users, it might have caused serious problems. First, users might not have known that email notifications existed. Second, the use of email notifications are not traditionally supported by websites and users might be unfamiliar as to what the service actually provides, thus reluctant to subscribe to them. By making the subscriptions automatic, I guaranteed that all users of CLEW would have the opportunity to use email notifications. Then they could choose to unsubscribe, if they so desired.

I have realized that email notifications should have an “opt-in” subscription strategy under normal operation. Development is under way to make automatic subscription to email notifications an option during the registration process. Additionally, the “Edit Profile” page will provide the functionality to unsubscribe and subscribe to all of the email notifications with a click of a button. This added functionality will be added to CLEW after the evaluation period is over.

There have been 15 email notifications during this evaluation time period, March 4, 2003 to March 31, 2003, a period of 27 days. This shows that the CLEW Services have been used approximately once every two days during the evaluation period, which provides further evidence that CLEW can improve the ability to dynamically make changes to the website; supporting Claim 1 of my thesis statement. Figure 20 is an example of a typical email notification.


UH Web Mail - Microsoft Internet Explorer
File Edit View Favorites Tools Help
ITS UH Web Mail for Aaron Akihisa Kagawa
Folders clew admin Message Addresses Options Help Log Out
Compose Reply Reply All Forward Delete Prev Next
Move message to folder:
Add Sender
Internet

From "ICS Dept. Website Admin" <kagawaa@hawaii.edu>
Sent Friday, March 14, 2003 2:00 am
To kagawaa@hawaii.edu
Subject UH ICS Notification Report

The following changes occurred yesterday at the ICS Website:

*Textbook Exchange:
- Posted: ICS 311, DATA STRUCTURES AND PROGRAM DESIGN IN C++
- Posted: ICS 412, Operating System
* http://newiwi.ics.hawaii.edu/clew/switchModule.do?prefix=/textbooks&page=/textbooksViewTextbooks.do

*Current Frequently Asked Questions:
- Posted: Open Source Development
- Posted: ICS Department
- Posted: Career and Internships
* http://newiwi.ics.hawaii.edu/clew/switchModule.do?prefix=/faq&page=/faqDisplayPublic.do

*Resume:
- Posted: XML Windows XP Windows NT/2000 Visual Basic Unix SmallTalk SQL Server Python Prolog Perl PHP Oracle MySQL MacOS MS Access Lisp Linux Lindows Javascript Java JSP HTML FoxPro FORTRAN COBOL C++ C# C Assembly ASP
* http://newiwi.ics.hawaii.edu/clew/switchModule.do?prefix=/resume&page=/resumeViewResume.do

*Office Hours:
- Added: ICS 111, Tuesday, 9:00 AM, POST 319.
- Added: ICS 111, Thursday, 9:00 AM, POST 319.
- Added: ICS 111, Tuesday, 10:00 AM, POST 319.
- Added: ICS 111, Thursday, 10:00 AM, POST 319.
- Added: ICS 211, Tuesday, 9:00 AM, POST 319.
- Added: ICS 211, Thursday, 9:00 AM, POST 319

Figure 20 - An example email notification

The particular email notification shows that posting occurred in 4 services and provides a link to view the postings.

On March 31, 2003 there are 123 registered users in the CLEW System. On the same day, 116 users received an email notification. This shows that 94% of CLEW's users are subscribed to receive email notifications. This provides evidence that CLEW can notify CLEW's users of changes in the department website by creating a notification system to alert the community of changes to the website; supporting Claim 2 of my thesis statement.

5.1.3 Results of the CLEW Model evaluations: Improved awareness

The methodology of this evaluation states that the pre-release and post-release questionnaires was administered to a single UH computer science class to evaluate the general community's change of awareness of various aspects of the department.

The pre-release questionnaire was administered to the UH ICS 313 class on February 26, 2003. A total of 33 students answered the pre-release questionnaire. On March 13, 2003 the CLEW System replaced the original ICS website. The post-release questionnaire was administered to the same ICS 313 class on April 2, 2003. A total of 25 students answered the post-release questionnaire. The following are the result of the two questionnaires.

Question 1: I am aware of recent news that concerns the ICS department.

(Value) Level of Agreement	Pre-Release	Post-Release
(1) Strongly Agree	0	1
(2) Agree	5	5
(3) No Opinion	9	11
(4) Disagree	14	4
(5) Strongly Disagree	5	3

Table 6 - The improved awareness questionnaire results of question 1

The goal of this question is to compare the effectiveness of the original UH ICS News Listing service and the CLEW News Bulletin Service to dynamically change the website. This question averaged a response of 3.6 (Disagree) for the pre-release questionnaire and 3.1 (No Opinion) for the post-release questionnaire. The results provide evidence that the CLEW News Bulletin Service increases the awareness of the department community.

Question 2: I am aware of recent technical reports that have been published in the ICS department.

(Value) Level of Agreement	Pre-Release	Post-Release
(1) Strongly Agree	0	0
(2) Agree	1	2
(3) No Opinion	6	11
(4) Disagree	15	5
(5) Strongly Disagree	11	5

Table 7 - The improved awareness questionnaire results of question 2

The original UH ICS department website did not provide an interactive service to list technical reports. Also, it did not have any static web pages displaying any information about the technical reports. Therefore, the goal of this question is to evaluate the effectiveness of the CLEW Technical Report Service to increase the community's awareness of technical reports that have been published by the department community. This question averaged a response of 4.1 (Disagree) for the pre-release questionnaire and 3.4 (No Opinion) for the post-release questionnaire. The results provide evidence that the awareness of the department community's published technical reports has improved using the CLEW Technical Reports Service.

Question 3: I am aware of the office hours of the people in the ICS department.

(Value) Level of Agreement	Pre-Release	Post-Release
(1) Strongly Agree	1	1
(2) Agree	8	7
(3) No Opinion	6	9
(4) Disagree	12	7
(5) Strongly Disagree	6	1

Table 8 - The improved awareness questionnaire results of question 3

The original UH ICS department website did not have a service to list the office hours of the department members. The UH ICS department relied on personal homepages of the department members to list their own office hours. The CLEW Office Hours Service provided a consolidated view of all department members' office hours. The goal of this question is to evaluate the effectiveness of the CLEW Office Hour Service to increase the community's awareness of the department members' office hours. This question averaged a response of 3.4 (No Opinion) for the pre-release questionnaire and 3 (No Opinion) for the post-release questionnaire. The results of this question provide evidence that the CLEW Office Hours Service increases the awareness of the department community.

Question 4: I use the resume listing in the ICS website.

(Value) Level of Agreement	Pre-Release	Post-Release
(1) Strongly Agree	1	1
(2) Agree	1	1
(3) No Opinion	4	7
(4) Disagree	12	7
(5) Strongly Disagree	15	7

Table 9 - The improved awareness questionnaire results of question 4

I have made a mistake in the formulation of this question. This question tends to be a yes / no type question, which does not match the set of answers the participant can choose from. I should have kept the theme of the previous questions, in asking the participants awareness.

Regardless of the phrasing problem, the goal of this question is to compare the effectiveness of the original UH ICS Resume Listing service and the CLEW Resume Service to dynamically add resumes to the department website. This question averaged a response of 4.1 (Disagree) for the pre-release questionnaire and 3.8 (Disagree) for the post-release questionnaire. The results of these questionnaires show that the CLEW Resume Service provides service that the community can use. This can be indirectly interpreted that the community has a better awareness of the service because of their higher use.

Question 5: I use the student email/website listing in the ICS website.

(Value) Level of Agreement	Pre-Release	Post-Release
(1) Strongly Agree	3	0
(2) Agree	3	1
(3) No Opinion	4	8
(4) Disagree	11	8
(5) Strongly Disagree	11	7

Table 10 - The improved awareness questionnaire results of question 5

I have made a mistake in the formulation of this question. This question tends to be a yes / no type question, which does not match the set of answers the participant can choose from. I should have kept the theme of the previous questions, in asking the participants awareness of the particular information.

Regardless of the phrasing problem, the goal of this question is to compare the effectiveness of the original UH ICS People Listing service and the CLEW's ability to list personal information to the department website. This question averaged a response of 3.6 (Disagree) for the pre-release questionnaire and 3.9 (Disagree) for the post-release questionnaire. The results of these questionnaires show that the CLEW list of the personal information listing was not used as much as the original UH ICS People Listing service. However, this result also provides evidence that the CLEW's ability to list personal information might be superior to the original UH ICS People Listing service. CLEW changed the users' process to add themselves to the People listing. The original UH ICS People listing required the users to explicitly add themselves to the people listing, however, CLEW now does this automatically once a user obtains an account with the CLEW Website.

Question 6: The ICS website is an important resource for me to gather information concerning the ICS department.

(Value) Level of Agreement	Pre-Release	Post-Release
(1) Strongly Agree	2	6
(2) Agree	15	7
(3) No Opinion	7	8
(4) Disagree	7	2
(5) Strongly Disagree	2	2

Table 11 - The improved awareness questionnaire results of question 6

For CLEW to be successful, the department community must feel that a department website is an important source of information. The results of this question provide evidence that the department community thinks a department website is an important resource. This question averaged a response of 2.8 (No Opinion) for the pre-release questionnaire and 2.4 (Agree) for the post-release questionnaire. After CLEW was introduced the participants indicated that the importance of a department website increased.

Question 7: How often do you visit the ICS website?

(Value) Frequency	Pre-Release	Post-Release
(1) More than daily	0	0
(2) Daily	1	2
(3) Weekly	6	6
(4) Monthly	21	12
(5) Never	4	4

Table 12 - The improved awareness questionnaire results of question 7

The goal of this question is to find the frequency that the community visits the UH ICS department website. This question averaged responses of 3.9 (Monthly) for the pre-release questionnaire and 3.8 (Monthly) for the post-release questionnaire. The results show that regardless of the version of website the community rarely visits the department website. This also indicates that by providing new information on the website is an ineffective way to communication new issues to the community. An alternative way to notify the community of changes to the department is needed, and CLEW provides a possible answer with the CLEW Email Notification process (for more information about CLEW Email Notifications see Section 1.4.2).

Question 8: How often do you think the ICS website gets updated with new information?

(Value) Frequency	Pre-Release	Post-Release
(1) More than daily	0	1
(2) Daily	1	2
(3) Weekly	1	12
(4) Monthly	19	6
(5) Never	9	3

Table 13 - The improved awareness questionnaire results of question 8

The goal of this question is to find how often the community thinks that the UH ICS department website is updated with new information. This question averaged a response of 4.2 (Monthly) for the pre-release questionnaire and 3.3 (Weekly) for the post-release questionnaire. This result shows that a majority of the participants now think that the UH ICS department website is updated with new information on a weekly basis.

The change of the users' opinion could have resulted from two features of CLEW. First, the CLEW Services (e.g. Office Hours Service, Resume Service, Textbook Service) were presented to the user directly on the home page of CLEW UH ICS department website, therefore, the users knew the website had the potential to be updated on a regular basis. Second, the CLEW Email Notifications provided the users with daily emails of the updates of new information to the website, therefore providing the users with a direct indication that information has been added to the department website.

Question 9: What is the URL of the ICS website?

URL	Pre-Release	Post-Release
Correct	32	23
Blank	1	1
Incorrect	0	1

Table 14 - The improved awareness questionnaire results of question 9

There were no particular goals of this question. Rather, I thought it would be interesting to see how many ICS students knew what the URL is for the department website. This question gave a percent correct of responses of 97% for the pre-release questionnaire and 92% for the post-release questionnaire. The responses to this question do not have any dependence on what version of the department website is currently in operation. Therefore, provides no evidence to support any of my thesis claims.

Question 10: What is the primary purpose of the ICS website?

Pre-Release Sample Results
To provide information about the ICS department such as news, technical reports, office hours, resume listings, and student email/website listing.
A repository of useful information.
Don't know
Right now, it has no purpose.

Table 15 - The improved awareness questionnaire results of pre-release question 10

Post-Release Sample Results
Provide Information about the department
To inform about the ICS department.
Disseminate information on program offerings and provide point of contact to professors via email adding and home pages
Provide information for ics students

Table 16 - The improved awareness questionnaire results of post-release question 10

The results of this open ended question show that most of the participants feel that the purpose of a department website is to provide information about the department. This question does not provide any evidence for any of my thesis claims.

However, there is one interesting finding. All responses to the post-release questionnaire mention "providing information about the department." This differs from the pre-release questionnaire, where there were a couple of responses indicating that either the participants "did not know what the purpose of a department website is for" or that the ICS department website has "no purpose".

5.2 Results of the CLEW Feedback evaluations

The methodology of this evaluation states that the feedback provided by CLEW's users was categorized into two categories to understand where the users' concerns are with the use of the CLEW System. The feedback was obtained by utilizing the CLEW Feedback Service, which by filling out a form, the users can send an email to the developers with their feedback.

As of March 31, 2003 there have been 58 feedbacks provided by the CLEW users. The following table presents the results of the users' feedback.

	Usability	**Functionality**
CLEW System	3	3
FAQ	1	0
Feedback	1	0
Login	8	4
News Bulletin	3	1
Office Hours	4	2
Poll	4	1
Resume	1	6
Technical Reports	0	1
Textbooks	2	1
Webedit	2	0
All Services	1	5
Static Pages	4	0
	Total: 34	Total: 24

Table 17 - Matrix of results used to organize user feedback evaluation

Table 17 shows that there has been a significant amount of feedback from the users of the system. This also shows that the users have a vested interest in the functionality and usability of their department website.

The majority of the feedbacks, 34 out of 58, are concerned with usability issues. This result was anticipated, because the CLEW Interface was not designed or evaluated with Human Computer Interaction usability techniques. Therefore, usability issues in the CLEW UH ICS department website is a definite problem and needs further investigation.

The number of feedbacks concerned with the functional workings of CLEW, accounts for 24 of 58 user feedbacks. This result provides evidence that the CLEW System can function as a department website without a lot of functional problems. Furthermore, of the 24 functionality concerned feedbacks only 5 of them had any serious implications. All 5 of those "serious" problems were fixed in a matter of minutes, the rest of the functional feedbacks were either so minor that "fixes" were not necessary.

5.3 Results of the CLEW Developer evaluations

The methodology of this evaluation states that the developers of the CLEW Services will be given a questionnaire to find out the developers' experiences of creating their CLEW Services.

A total of 7 developers answered the developer questionnaire. The following are the results of the questionnaire.

Question 1: It was relatively easy to create services for the CLEW environment

(Value) Level of Agreement	Developer Questionnaire
(1) Strongly Agree	1
(2) Agree	3
(3) No Opinion	2
(4) Disagree	1
(5) Strongly Disagree	0

Table 18 - The developer questionnaire results of question 1

The goal of this question was to find the level of ease or difficulty the developers of the CLEW Services had implementing their respective services. Most of the developers felt that it was relatively easy to create the CLEW Services. The ease of which the CLEW Services were created provides some evidence about the "kernelized" architecture of CLEW. In my opinion, the CLEW Kernel provides the CLEW Services with various functionality that allows for easier implementation of the Services. This result provides evidence that supports this opinion. It also supports Claim 4 of my thesis statement.

Question 2: The CLEW framework allows the services to be easily "plugged-in" to the system.

(Value) Level of Agreement	Developer Questionnaire
(1) Strongly Agree	0
(2) Agree	6
(3) No Opinion	1
(4) Disagree	0
(5) Strongly Disagree	0

Table 19 - The developer questionnaire results of question 2

The goal of this question was to find the level of ease or difficulty the developers of the CLEW Services had with integrating their services with the CLEW Kernel. Most of the developers felt that it was relatively easy to "plug-in" the services into the CLEW Kernel. This result provides evidence that the "kernelized" architecture provides a good architecture to support a modularized development of CLEW Services; supporting Claim 4 of my thesis statement.

Question 3: Creating CLEW Services has enhanced my understanding of various software technologies, for example, Ant, Jdom, JUnit, HttpUnit, etc.

(Value) Level of Agreement	Developer Questionnaire
(1) Strongly Agree	7
(2) Agree	0
(3) No Opinion	0
(4) Disagree	0
(5) Strongly Disagree	0

Table 20 - The developer questionnaire results of question 3

The goal of this question was to find the level of educational value of creating CLEW Services. All the developers felt that creating the CLEW Services has enhanced their understanding of software technologies.

Question 4: Creating CLEW Services has enhanced my understanding of software engineering.

(Value) Level of Agreement	**Developer Questionnaire**
(1) Strongly Agree	7
(2) Agree	0
(3) No Opinion	0
(4) Disagree	0
(5) Strongly Disagree	0

Table 21 - The developer questionnaire results of question 4

The goal of this question was to find the level of educational value of creating CLEW Services. All the developers felt that creating the CLEW Services has enhanced their understanding of software engineering.

Question 5: I would recommend working on CLEW Services as a form of education for students.

(Value) Level of Agreement	Developer Questionnaire
(1) Strongly Agree	5
(2) Agree	2
(3) No Opinion	0
(4) Disagree	0
(5) Strongly Disagree	0

Table 22 - The developer questionnaire results of question 5

The goal of this question was to find if the developers feel other students would have an educational benefit from working on CLEW Services. This result shows the confidence that the developers have in using CLEW as a project in the classroom setting. I felt that the developers learned valuable lessons about working in a group development setting.

Question 6: I enjoyed developing CLEW Services.

(Value) Level of Agreement	Developer Questionnaire
(1) Strongly Agree	6
(2) Agree	1
(3) No Opinion	0
(4) Disagree	0
(5) Strongly Disagree	0

Table 23 - The developer questionnaire results of question 6

The goal of this question was to determine if the developers enjoyed working on CLEW Services. The majority of the developers strongly agreed that they enjoyed developing the CLEW Services. The CLEW project has enabled this group of developers to work closely together and contribute to a worth while goal. I feel the use of Extreme Programming's pair programming had a lot to do with the enjoyment of the developers. Pair programming allows a couple of programmers to work together on a single computer. This technique increases the enjoyment and learning experience of both programmers involved.

Question 7: Please provide some additional comments about your experience developing CLEW Services.

Sample Results
I think CLEW is getting harder and harder to understand as a developer. I think the build process of the system is getting kind of out of hand. However, I've never worked on any other big applications like CLEW, so maybe the build process is always like that.
The hardest part about making a service for CLEW is getting it started. In CREST there were multiple places you needed to add your service to. In CLEW, it is much better. But, it would be extremely hard if I didn't have any other service as an example.
We have been working on this system for so long, we know the system pretty well so developing services are not so bad. If you told some one totally new to come in and do it, they will probably be stuck on it for a while if they didn't have any template to follow. CLEW itself is becoming a relatively large system, so it might start to get harder to maintain overtime.
I definitely enjoyed developing the services for CLEW. The major difference between this project and many of the other projects was the fact that it will be in use, instead of being discarded at the end of the semester.
It was a good experience working on this project. It is better than any other project out there as for something to be done in a class. It helped us to all work together. One thing that is good about this project is that we get to see it actually being used, I have a sense of accomplishment, plus I can actually show someone that I did something. Overall, it has been great working on this project.

Table 24 - The developer questionnaire results of question 7

The responses to this questionnaire clearly show that the developers enjoyed being involved with the CLEW System. The overall results of the questionnaire show that CLEW Services are somewhat challenging to develop, developers have gained a lot of knowledge about Software Engineering and various software technologies, developers strongly recommend working on this system as a form of education. Developers also commented that they appreciate that this system is actually used by real users.

Chapter 6. Future Directions of CLEW

This chapter presents potential future directions of CLEW development and experimentation.

6.1 Providing CLEW Services to allow changes of removal in the website.

Currently, CLEW provides interactive services that allow changes of additions to the department website. The evaluations have proven that the CLEW Services allow easy postings to the CLEW Website. However, as time goes by, some of these additions will become dated and require removal.

CLEW has functionality to support removal of old information from the website. However, to make users "aware" of old information, it would be useful to notify them. I have come up with a couple ideas on how this can be accomplished:

- (1) The CLEW Services add the functionality to check the dates of the postings that were made to the service. Then using some threshold of time, the system will determine if the posting could be outdated.
- (2) The CLEW Webedit Service, a service that manages the static pages in the website, adds the functionality to read in all static pages in the CLEW ystem checking for the date the page was last modified. Using some threshold of time, the Webedit will determine if the static page could be outdated.

If the information in the postings or on the static pages are determined to be outdated, then the CLEW System can send an email notification to the users who are responsible for the posting or static page. They can then take the appropriate actions to ensure that the information is up-to-date.

6.2 HCI evaluations and redesign of the CLEW Interface

During the evaluation of CLEW many of the user's feedback were aimed at pointing out usability problems of the CLEW Interface. These usability problems are valid concerns, because the CLEW System has not integrated HCI issues into its development process.

Two projects can be created from using HCI techniques to improve the CLEW Interface. First, the current CLEW Interface can be evaluated using HCI evaluation techniques to solve the usability problems of the interface. This evaluation can be aided by my collection of user feedbacks through my evaluation period of CLEW.

Second, a total redesign of the CLEW Interface could be done using HCI techniques. Currently, the CLEW Interface strongly resembles the original UH ICS department website and it contains many UH ICS specific information. By creating a generic interface, CLEW can be deployed to other departments.

6.3 CLEW as a computer science department project

CLEW has the potential to provide a computer science department with interesting opportunities to collaborate on a single project. One such possibility is integrating CLEW into a computer science department curriculum.

Currently, the two computer science fields that make the most sense for CLEW to be integrated into, are the Human Computer Interaction field and Software Engineering field. The following sections presents two proposals on how CLEW development could be integrated in the HCI and SE computer science curriculums.

The first proposal integrates the Human Compute Interaction curriculum with the Software Engineering curriculum by creating a single class. This class will use both HCI and SE techniques to work on CLEW. This approach allows both SE issues and HCI issues to be presented side by side. As a result, students learn how issues in the different techniques should influence each other.

The second proposal integrates CLEW development into separate HCI and SE classes. That allows CLEW to be a central artifact that the classes will focus on. The following is how CLEW could be used in both the HCI curriculum and in the SE curriculum

In the HCI curriculum, projects would be constrained to evaluating current CLEW Services and creating new CLEW Services. The HCI classes will be given some software requirements, such as they are restricted to design interfaces for JSP, Java Web Applications, XML database, etc. It is their job to create design and evaluation reports that will aid in the implementation of the CLEW Services.

In the SE curriculum, using the design and evaluation reports provided by the HCI classes, the SE classes will implement CLEW Services following the design issues provided by the reports.

This proposal will work best if HCI and SE courses occur in succession. So, the HCI reports can become the input for the SE implementation.

6.4 The CLEW System used as a department community building tool

The CLEW System can be evaluated following Andrada's research methodology [4] to determine how CLEW can be used to improve the sense of community in a collegiate department.

Currently, my research methodology assumes that the UH ICS department has a sense of community. However, this assumption needs to be validated. By following Andrada's research methodology one can find, if in fact the UH ICS department has a sense of community and if CLEW can improve that sense of community.

6.5 Deploying the CLEW System to other collegiate departments

The CLEW System has not been specifically designed for the UH ICS department. Rather, one of CLEW's goals is to provide a system that any collegiate department can use to support accurate modeling of their department in their website. There are two types of collegiate departments that can benefit from using CLEW.

First, a computer science department can use CLEW for both their department website and as a computer science department project (Explained in Section 6.3).

Second, any collegiate department that has low resources in creating an interactive department website. The CLEW System is entirely open source and can be obtained free of charge. Also, because the CLEW System provides interactive services the duties of a webmaster are decreased significantly.

6.6 The continued development of CLEW

The development of the CLEW System, the CLEW Kernel and the CLEW Services are in no way complete. There are many possible enhancements that have been uncovered in the evaluation of this system. A couple of possibilities are presented in the following sections.

- The default Login Service provided by the CLEW Kernel can be improved by integrating the login and registrations with a LDAP server.
- Create a CLEW Calendar Service that allows the department to dynamically manage events, seminars, activities, etc.
- Create a CLEW Course Management Service that allows students to plan out the course they wish to take.
- Create a CLEW Project Service that allows students to advertise their school projects.
- Create a better CLEW Feedback Service that allows better management of feedbacks from the website users.

There are many services that one can think of that can be added to the CLEW System, to further help the department community interact with the department website.

Appendix A. Pre-Release Questionnaire for CLEW

Thank you for your participation. As a reminder, your participation in this research is voluntary. All references to data gathered will be made anonymously.

-- Aaron Kagawa

All questions are about the University of Hawaii's Department of Information and Computer Science.

	Strongly agree	Agree	No opinion	Disagree	Strongly disagree
1. I am aware of recent news that concerns the ICS department.	1	2	3	4	5
2. I am aware of recent technical reports that have been published in the ICS department.	1	2	3	4	5
3. I aware of the office hours of the people in the ICS department.	1	2	3	4	5
4. I use the resume listing in the ICS website.	1	2	3	4	5
5. I use the student email/website listing in the ICS website.	1	2	3	4	5
6. The ICS website is an important resource for me to gather information concerning the ICS department.	1	2	3	4	5

	More than daily	Daily	Weekly	Monthly	Never
7. How often do you visit the ICS website?	1	2	3	4	5
8. How often do you think the ICS website gets updated with new information?	1	2	3	4	5

9. What is the URL of the ICS website?

http://www.____________________________________

10. What is the primary purpose of the ICS website?

Appendix B. Post-Release Questionnaire for CLEW

Thank you for your participation. As a reminder, your participation in this research is voluntary. All references to data gathered will be made anonymously.

-- Aaron Kagawa

All questions are about the University of Hawaii's Department of Information and Computer Science.

	Strongly agree	Agree	No opinion	Disagree	Strongly disagree
1. I am aware of recent news that concerns the ICS department.	1	2	3	4	5
2. I am aware of recent technical reports that have been published in the ICS department.	1	2	3	4	5
3. I aware of the office hours of people in the ICS department.	1	2	3	4	5
4. I use the resume listing in the ICS website.	1	2	3	4	5
5. I use the student email/website listing in the ICS website.	1	2	3	4	5
6. The ICS website is an important resource for me to gather information concerning the ICS department.	1	2	3	4	5

	More than daily	Daily	Weekly	Monthly	Never
7. How often do you visit the ICS website?	1	2	3	4	5
8. How often do you think the ICS website gets updated with new information?	1	2	3	4	5

9. What is the URL of the ICS website?

http://www.______________________________

10. What is the primary purpose of the ICS website?

Appendix C. Questionnaire for CLEW Developers

Thank you for your participation. As a reminder, your participation in this research is voluntary. All references to data gathered will be made anonymously.

-- Aaron Kagawa

All questions are about the development of the CLEW(and CREST) services.

	Strongly agree	Agree	No opinion	Disagree	Strongly disagree
1. It was relatively easy to create services for the CLEW environment.	1	2	3	4	5
2. The CLEW framework allows the services to be easily “plugged-in” to the system.	1	2	3	4	5
3. Creating CLEW Services has enhanced my understanding of various software technologies, for example, Ant, Jdom, Junit, HttpUnit, etc.	1	2	3	4	5
4. Creating CLEW Services has enhanced my understanding of software engineering.	1	2	3	4	5
5. I would recommend working on CLEW Services as a form of education for students.	1	2	3	4	5
6. I enjoyed working on the CLEW Services.	1	2	3	4	5

7. Please provide some additional comments about your experience developing CLEW Services.

Appendix D. The Developers of the CLEW System

This appendix provides a chronological breakdown of the specific developers who have contributed to the CLEW System. Also, the following provides a short description of each major development.

Course: ICS 413 / ICS 613 Software Engineering I

Semester: Spring 2002

Professor: Philip Johnson

System: Individual Systems (see Section 3.1.1 for more details)

Note: the ICS 413 and the ICS 613 students mirrored each others development often creating overlapping systems.

System Name	Developers	Short Description
Login	(413) – Cinthia Ha, Aaron Kagawa (613) – Christoph Aschwanden, Yihua Xie, Cedric Qin Zang, Jian Zhang	Provides the ability to register and login into a web application
Resume	(413) – Travis Morita, Jareus Sylva (613) – Sally Dunn, Hongbing Kou, Rui Xue, Weiping Yan	Provides the ability to create, post, and manage resumes.
User Profile	(413) – Everett Inamasu, George Maximo, Richard Shuen, Corey Taira (613) – Joy Agustin, Dave Burns	Provides the ability to create and manage a user profile.
Tutor Schedule	(413) – Anh Huynh, Swee Lee Jim, Steven Miyakawa, Kin Lik Wang	Provides the ability to show the tutor schedules, locations, etc.
Textbook	(413) – Spencer Au, Everett Inamasu, Preston Ma, Jason Medeiros	Provides the ability to post the sale of textbooks.
Department Office	(413) – Jason Antolin, Swee Lee Jim, Corey Taira, Kin Lik Wang	Provides the ability to show the locations of the offices in the department.
Frequently Asked Questions	(413) – Jason Antolin, Aaron Kagawa, Jesse Tom	Provides the ability to post and questions and answers.

Course Evaluation	(413) – Anh Huynh, Preston Ma, Steven Miyakawa, Richard Shuen, Jesse Tom	Provides the ability to critique a course and view critiques of courses.
News Bulletin	(613) – Xin Chen, Burt Leung, Ping Liu, Yihau Xie	Provides the ability to post and manage department news.
Poll	(613) – Christoph Aschwanden, William Doane, Jun Xu, Jian Zhang	Provides the ability to create and manage a poll.
TechReport	(613) – Dave Burns, Hongbing Kou, Yihua Xie, Weiping Yan	Provides the ability to create and manage technical reports.
Job Posting	(613) – Sally Dun, Ping Liu, Frank Tien, Jun Xu	Provides the ability to post job advertisements.

Course: ICS 499 Individual Computer Project

Semester: Summer 2002

Professor: Philip Johnson

System: CREST(see Section 3.1.3 for more details)

Development Leaders: Aaron Kagawa, Jesse Tom

Service Name	Developers
Login	Aaron Kagawa, Jesse Tom

Course: ICS 414 Software Engineering II

Semester: Fall 2002

Professor: Philip Johnson

System: CREST(see Section 3.1.3 for more details)

Development Leaders: Aaron Kagawa, Jesse Tom

Service Name	Developers
FAQ	Jason Antolin, Preston Ma

Login	Aaron Kagawa, Jesse Tom
News Bulletin	Richard Shuen, Corey Taira
Office Hours	Everett Inamasu, Jason Medeiros
Poll	Aaron Kagawa
Resume	Travis Morita, Alex Sue
Technical Reports	Spencer Au, Steven Miyakawa, Takuya Yamashita
Textbooks	Jesse Tom

Course: ICS 499 Individual Computer Project

Semester: Spring 2003

Professor: Philip Johnson

System: CLEW (see Section 3.1.3 for more details)

Development Leaders: Aaron Kagawa

CLEW Kernel	Aaron Kagawa

Service Name	Developers
Developer Information	Richard Shuen
FAQ	Jason Antolin
Login	Aaron Kagawa, Jesse Tom
News Bulletin	Richard Shuen
Office Hours	Jason Medeiros
Poll	Aaron Kagawa
Resume	Alex Sue
Search	Steven Miyakawa
Technical Reports	Aaron Kagawa, Steven Miyakawa
Textbooks	Aaron Kagawa, Jesse Tom
Webedit	Corey Taira

Appendix E. Information from Users Needs

The set of users who use a collegiate department website (Ritter, et al):

- Current Students
- Prospective students – undergraduate, graduate, local, national, international
- Faculty/Staff – in the department, at the university, at another university
- Alumni
- Parents
- Donors
- Research consumers
- Research program managers
- Press
- Prospective faculty
- State legislators (for state schools)
- Disabled users

Set of website tasks to support all users. (Ritter, et al)

- Introductory
  - Welcome message from the Dean/Head of Department
  - Message from the Dean/Head of Department
  - About the college
  - Purpose of school/Mission statement/Vision statement
  - Publications, full or sample
- People
  - Bios for faculty, support staff, administrators, alumni, graduate students, undergraduate students, associated faculty, friends of the department, teaching staff, post-doctoral fellows, visitors
  - Listings of categories (e.g. Faculty)
  - Contact information/Directory
  - Points of contact for:
    - General information
    - The web
    - Admissions
    - Research
    - Press
    - Records
    - Students Affairs/Internships
    - Study Abroad
- Programs

  - Undergraduate/Graduate
  - Research
  - Outreach
  - Study abroad
  - Internships/Cooperative Education
  - Associated Conferences
  - Institutes, Centers, and Labs
  - Distance Learning Seminars and Certificates
  - Rankings
- Policy
  - Alumni relations
  - Multicultural affairs/Diversity
  - Corporate relations
  - Administration
  - Academic
- Current events
  - Calendar of events
  - Current Issues
  - Press releases
  - News and Media
- Prospective students
  - Admission requirements
  - Advising sites
  - What graduates can/will do
  - Campus and classroom pictures
  - Research topics
  - Student organizations and clubs
  - Mentoring
  - Visiting
  - Applications
  - Summary of reasons to come
- Financial matters
  - Scholarship information for all groups of people – undergraduate, graduate, post-doctoral
  - Gifts/How to donate
  - How to be a coop partner
  - Campaign/fundraising goals
  - Job openings
- Physical location
  - Address of college (email, fax, phone)
  - Campus maps/maps for coming from a distance
  - Find a building (offices, classrooms, labs)
  - Virtual tour
  - List of campuses (from multiple campus programs)
  - Buildings(s) plans
- School resources

  - Enrollment figures (student/faculty)
  - Schools/departments
  - Labs, Centers, Institutes
  - Other resources (this may vary widely from horse barns to pre-schools to particle accelerators)
- Specific majors
  - Major requirements
  - Degree options
  - Course offerings
  - BS degree guide for first year students
  - Related degrees
  - Achievement expectations
  - Minors
- Specific courses
  - Class announcements
  - Assignments
  - Lecture notes
  - Schedule
  - Syllabus
  - Class message boards and mailing lists
  - Assignment due dates
  - Assignment solutions
  - University regulations related to the course (e.g. American with Disabilities Act)
  - Grades

- Web features and support
  - Search
  - Contact webmaster
  - Link to university homepage and other associated centers and units
  - Related links
  - Frequently Asked Questions (FAQ) for Department
  - Student Resumes
  - Merchandise
- Alternative views
  - Text-only view
  - Alternate mediate available upon request
  - Alternate language (i.e. Spanish)

Bibliography

[1] S. Krug, *Don't make me think*. Indianapolis, Indiana USA: New Riders, 2000.

[2] D. K. Van Duyne, J. Landay, J. I. Hong, *The design of sites: patterns, principles, and processes for creating a customer-centered Web experience*. Boston, MA: Pearson Education, Inc., 2003.

[3] F.E. Ritter, A.R. Freed, O.L. Haskett, *Discovering user information needs: The case study of university department websites.* Computer Science and Engineering Department, Pennsylvania State University, University Park, Pennsylvania 16802, December 2002.

[4] R.M. Andrada, *Building community through the world wide web.* Department of Information and Computer Sciences, University of Hawaii, Honolulu, Hawaii 96822, M.S. Thesis, July, 1995.

[5] Introduction to Extreme Programming (XP). <http://www.xp123.com/xplor/xp9912/index.shtml>

[6] Introduction to Agile Development Methods. <http://csdl.ics.hawaii.edu/~johnson/413s02/AgileMethods.pdf>

[7] Apache Ant. <http://ant.apache.org/>

[8] The Apache Struts Web Application Framework. <http://jakarta.apache.org/struts/>

[9] JUnit, Testing Resources for Extreme Programming. <http://www.junit.org/index.htm>

[10] HttpUnit. <http://httpunit.sourceforge.net/>

[11] JBlanket. <http://csdl.ics.hawaii.edu/Tools/JBlanket/>

[12] Borland JBuilder. <http://www.borland.com/jbuilder/>

[13] Java (TM) Servlet Technology. <http://java.sun.com/products/servlet/>

[14] JavaServer Pages (TM) Technology. <http://java.sun.com/products/jsp/>

[15] Apache Tomcat. <http://jakarta.apache.org/tomcat/index.html>

[16] Java Profiler – JProfiler. <http://www.ej-technologies.com/products/jprofiler/overview.html>