

Equipment and Help Calls Management System

By

Kevin Hsu

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Engineering Technology

University of Cincinnati
College of Applied Science

June 2003

Equipment and Help Calls Management System

By

Kevin Hsu

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Engineering Technology

© Copyright 2003 Kevin Hsu

The author grants to the Information Engineering Technology Program permission to reproduce and distribute copies of this document in whole or in part.

Kevin Hsu

Date

Dr. Hazem Said, Faculty Advisor

Date

James F. Sullivan, Department Head

Date

Acknowledgements

I would like to give special thanks to Mu-Sheng Hsu, my grandpa for making my college experience possible. I would like to give special thanks to Erma Fritsche, Head of the Library Information Technology Department, along with Cherie Shroyer-Matchan, System Administrator and Thomas Scherz, User Services of the LIT team for allowing me to do this project. I would like to give special thanks to Professor Soleda Leung of University College for teaching me how to program and giving me encouragements. I would also like to thank Eboney Kimbrough for assisting me with grammatical errors in the report. I would like to give special thanks to Wayne Sibley for assisting me with the flow of the report. Finally, I would like to give special thanks to Dr. Hazem Said for his guidance.

Table of Contents

Section	Page
Acknowledgements	i
Table of Contents	ii
List of Figures	iv
Abstract	vii
1. Statement of the Problem	1
2. Description of the Solution	1
2.1. User Profile	3
2.1.1. LIT Personnel	3
2.1.2. Librarians	4
2.2. Design Protocols	4
2.2.1. MySQL Database	4
2.2.2. Programming	7
2.2.3. Networking	8
2.2.4. Multimedia	8
3. Deliverables	10
4. Design and Development	
4.1 Timeline	11
4.2 Budget	11
4.3 Software and Hardware	12
5. Proof of Design	13
5.1. Page Layout	14
5.2. Library Staff Section	15
5.3. HTTP Authentication Login	17
5.4. Equipment Section	18
5.5. Floor Map	34
5.6. Help Calls Section	36
5.7. Main Section	45
5.8. Personnel Section	45
6. Testing Procedures	50
6.1. The User Authentication is Required to Access Main Function of Web Application	50
6.2. GUI Testing	51

7. Conclusions and Recommendations	
7.1. Conclusions	51
7.2. Recommendations	51
Appendix A. Menu Database and Table	54
Appendix B. Equipment Database and Tables	56
Appendix C. Help Calls and Tables	60
Appendix D. Apache HTTP Authentication Setup	61
References	62

List of Figures

Figure Number	Page
Figure 1. Relationship for Equipment Database	5
Figure 2. Help Call Database Tables	6
Figure 3. PHP Script Request	7
Figure 4. Type Left Sided Menu and Right Side of the page for menu	8
Figure 5. Project Timeline	10
Figure 6. Hardware and Software Budget	11
Figure 7. LIT Web Pages Layout	13
Figure 8. Page Layout	14
Figure 9. Home Page	15
Figure 10. Add a New Help Call Page	16
Figure 11. Submitted Help Call	17
Figure 12. HTTP User Authentication Login Screen	17
Figure 13. Failed Authentication request Page	18
Figure 14. Add Computer Page	19
Figure 15. List of All Computers Page	20
Figure 16. Add a Monitor Page	21
Figure 17. List of All The Monitors Page	22
Figure 18. Add a Printer Page	23
Figure 19. List of All The Printers Page	24
Figure 20. Add a Phone and Data Jack Page	25
Figure 21. List of All The Phone & Data Jacks Page	26

Figure 22. Assign Equipment to Employee Page	27
Figure 23. List of Assigned Equipment	28
Figure 24. Full Detail Page	29
Figure 25. Equipment User Access Log Page	30
Figure 26. Workstation Lookup Page	31
Figure 27. Error Search result for Computer Page	32
Figure 28. Search Result for Computer Serial Matching Page	33
Figure 29. 4Spine Floor Map Page	34
Figure 30. 5 th Floor Spine Floor Map Page	35
Figure 31. Info. Commons Floor Map Page	35
Figure 32. Add a New Help Category Page	36
Figure 33. Edit Help Category Page	37
Figure 34. Delete Help Category Page	37
Figure 35. Open Calls Page	38
Figure 36. Help Calls By Engineer Page	39
Figure 37. Edit Open Help Call Page	40
Figure 38. Help Call Update Page	41
Figure 39. Added Record shown in List Page	41
Figure 40. Open Call Delete Page	42
Figure 41. Create Help Call Report Page	43
Figure 42. Generated Help Call Report Page	44
Figure 43. Help Call User Log Page	45
Figure 44. Add New Employee Page	46

Figure 45. JavaScript Error Message Alert Box	47
Figure 46. List of All The Employees Page	47
Figure 47. Employee Added Page	48
Figure 48. Added Employee Shown in List of All Employee Page	48
Figure 49. Update Employee Page	49
Figure 50. Update Successful Page	50
Figure 51. Employee Deleted Successful Page	50

Abstract

The Equipment and Help Calls Management System was created to give the support personnel of the Library Information Technology Department a way to manage their daily help calls and track computer equipment. The system allows quick access to information on computer equipment and opened help calls/trouble tickets online. The system also provides a Web form for library staff to submit help calls through the Web. The Equipment and Help Calls Management System is a user-friendly application developed with Hypertext Preprocessor (PHP) server side scripting language, JavaScript, MySQL, and Apache Web Server. The system allows administrators to search on computer equipment and add, edit, delete, and view user logs to table access, as well as access to opened help calls.

Equipment and Help Calls Management System

1. Statement of the Problem

Recently, the Library Automation Department and the Library Information Systems Support Department, located in Langsam Library at the University of Cincinnati, merged together to create the Library Information Technology Department (LIT). LIT supports eleven College and Departmental libraries, including the Ohio Mechanical Institute - College of Applied Science (OCAS) Library's public and staff workstations. LIT currently has more than 500 workstations in production. Since, this number increases yearly, LIT would like to have a more efficient way to manage both the computer equipment and help call databases. LIT Network administrators need a database access log and custom query report to streamline the process of logging and tracking both computer equipment and help calls. Additionally, data from both equipment and help calls need to be accessible off site. LIT would like to offer additional means for library staff to submit help calls via a Web application.

2. Description of the solution

The Equipment and Help Calls Management System (EHCMS) is a database driven Web application. It is a tool used by LIT personnel to administer its daily help calls and computer equipment. EHCMS is composed of three main components:

1. Relational Database Management System (RDBMS)

Equipment database and normalized tables

- ❖ Add, delete, update functions
- ❖ Query by Workstation ID, Computer Serial, or UC Property ID
- ❖ Database access log
- ❖ Custom queries.

Help call database

- ❖ Add, delete, update help call logs
- ❖ Query by technician of unresolved help calls
- ❖ Database access log
- ❖ Custom queries.

2. Apache Web Daemon

3. Programming languages (Interface)

- ❖ PHP
- ❖ HTML
- ❖ CSS

The equipment database includes the following functionalities:

- ❖ Enable technicians to access information stored on database.
- ❖ Enable technicians to perform data entry from job site.
- ❖ Enable technicians to search / query database by UC property number, PC serial number, or Workstation ID.

The help call management will include the following functionalities:

- ❖ Enable staff to submit a help call via a Web form.
- ❖ Enable technicians to check what's on their job list.
- ❖ Enable technicians to edit and update a job.
- ❖ Enable technicians/Administrator to generate a custom report.

Log tables were created in each database to store information on who logged in and performed add, update, or delete functions on the database by storing the user's login ID, what table he/she accessed, what action they performed on the table, and the time and date of access.

Through the use of Hypertext Preprocessor (PHP) in server side scripting to interact with the MySQL database and Hyper Text Markup Language (HTML) code embedded in PHP and vice-versa to create a user interface. Cascading Style Sheet (CSS) is linked in each document to add style, so there is a uniform look and feel. JavaScript is used to perform client-side input validation.

2.1 User Profile

The EHCMS has two user profiles. The targeted users are LIT personnel and librarians from LIT's supported libraries.

2.1.1 LIT Personnel

LIT personnel will use the LIT's Intranet site navigation links, which will allow access to both the equipment and help calls databases located in a MySQL database Server. They will have administrative privileges, such as read and write permissions. LIT personnel are very familiar with Microsoft applications, such as MS Office Suite and

Internet Explorer. Thus, little to no training is required for them to be able to navigate through the menu driven user interface.

This group has an intermediate-advanced level of technical knowledge and will be comfortable finding what they need on EHCMS. The EHCMS will make use of the UC branding initiative web template and its left sided navigation structure.

2.1.2 Librarians

This group is computer literate, yet the level varies. Librarians are trained on the Outlook Exchange Mail Client and use it daily to communicate with fellow employees. They should be familiar with filling out a web form with input text boxes and clicking on the submit button when completed. Access for this group will be restricted to a help call submission Web form to report any computer related problems.

2.2 Design Protocols

The project's focus is mainly on the programming and database aspect of IT, with minor focuses on networking and multimedia.

2.2.1 MySQL Database

MySQL is an open source RDBMS that was used to create the equipment and help call databases. The equipment database contains the following tables (See Figure 1.):

- ❖ Computers – stores data for both hardware and software installed on each workstation.
- ❖ Monitors – stores make, model, size, serial, and etc.
- ❖ Printers – stores data such as make, model, serial, and etc.
- ❖ Voice and data jack - stores voice and data jack number and its location.

- ❖ Departments – stores department name and what building it's located and its mail location and such.
- ❖ Users table – stores data about the user.
- ❖ Equipment detail – an objectified relationship of the aforementioned tables.

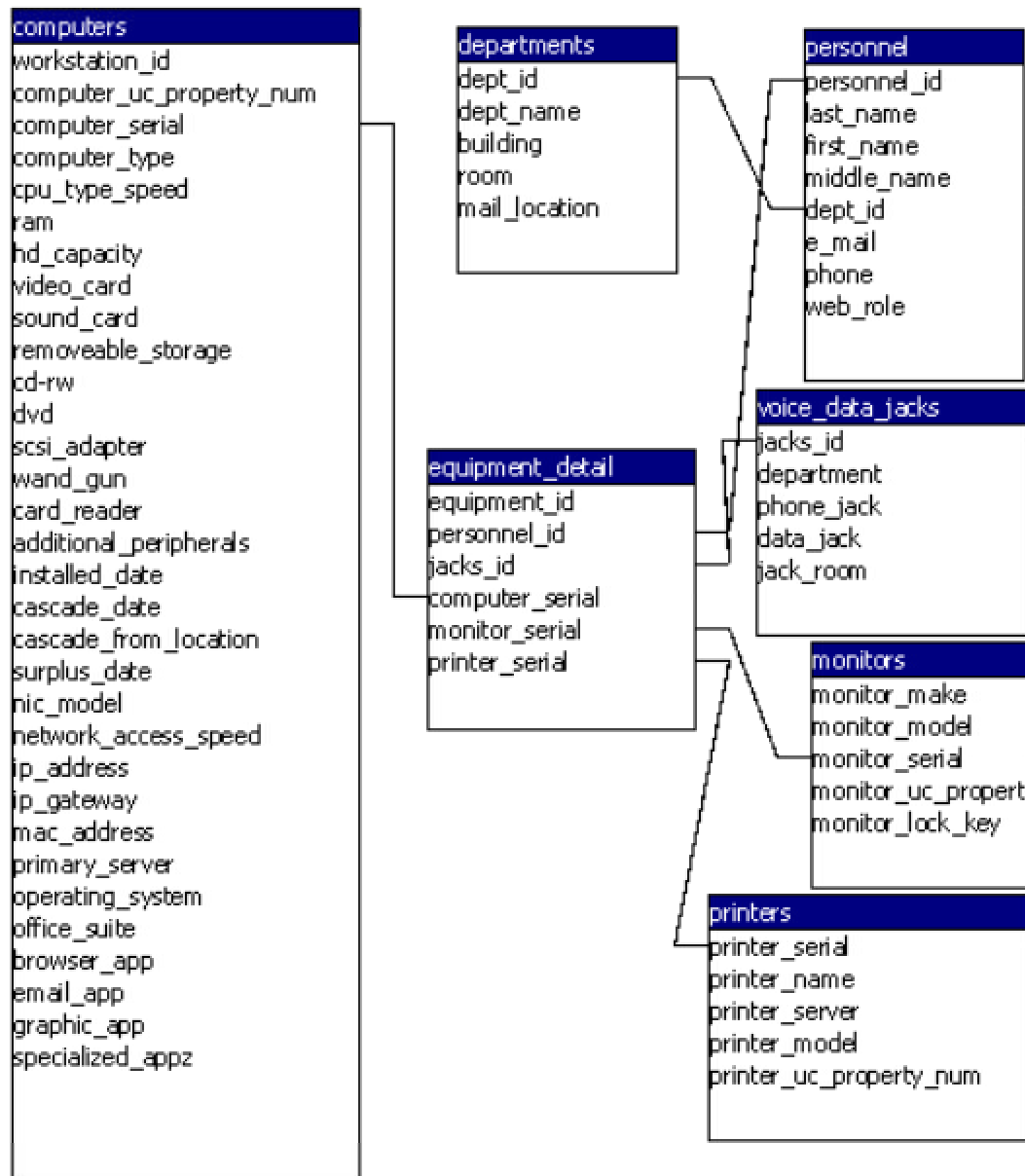


Figure 1. Relationship for Equipment database.

The help call database consists of the following three tables:

- ❖ Log – stores data related to a help call.
- ❖ Help categories – stores the types of problems users might encounter.
- ❖ Techs – stores names of help call staff.

log
help_id
date_arrived
completed
date_completed
help_catagory
help_desc
help_user
help_dept
help_phone
help_tech
help_notes
help_resolution
help_email

help_categories
help_type_id
help_type

techs
tech_id
tech_name

Figure 2. Shows Help Call database tables.

2.2.2 Programming

PHP, a server side scripting language is used to access the database along with HTML. CSS is used for formatting the presentation of the HTML page. JavaScript is used to handle the client side error checking. Since, PHP pages are stored on the web server; PHP script coding will not be viewable through the client's browser.

Figure 3 shows the process of a browser requesting a PHP page on the server to gain data stored on the database.

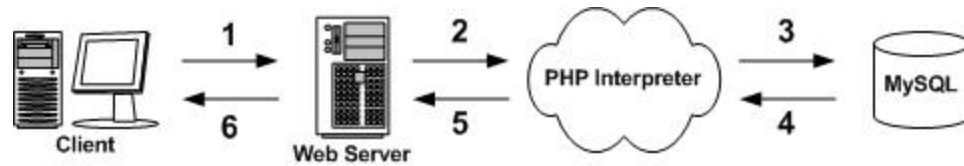


Figure 3. PHP Script request.

- ❖ Client's Web browser issues a request for a document that includes PHP.
- ❖ Web server retrieves the requested PHP page and sends it to the PHP interpreter.
- ❖ PHP engine parses the PHP script that contains commands to connect to MySQL database and query to execute.
- ❖ MySQL server executes the requested query and sends the query result back to the PHP interpreter.
- ❖ The PHP engine processes the script which includes formatting of the query into HTML and passes it back to the Web server.
- ❖ The requested page is passed back to the client's Web browser.

2.2.3 Networking

I used Apache as the Web Server application and made use of its built-in Apache user authentication Modules.

2.2.4 Multimedia

The site will focus on functionality and design layout. Color schemes will be drawn from the UC branding web template (See Figure4.).

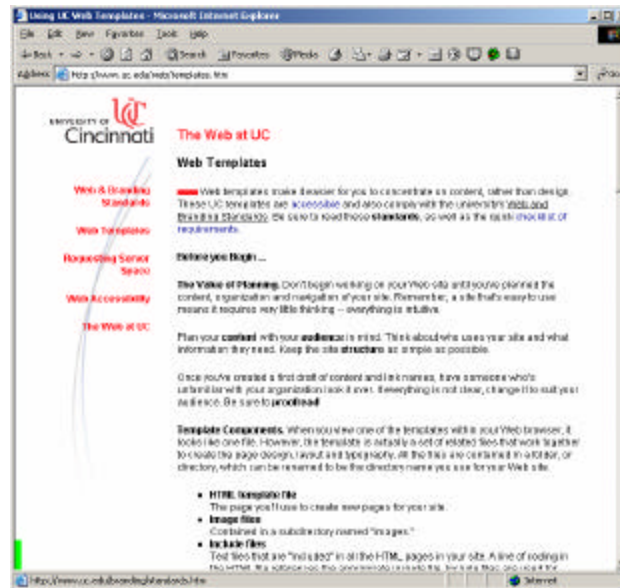


Figure 4. A typical left sided menu and right side of the page for content.

3. Deliverables

1. A Web-Based Equipment and Help Call Management System that enables inventory and help call tracking.
2. A back-end RDBMS that will facilitate the development of a dynamic front-end interface. This RDBMS will be developed on MySQL Server.
3. A dynamic front-end interface developed with HTML, PHP, CSS and JavaScript.
4. Librarians will be limited to add a new help call page.
5. Administrative users of the database are authenticated through Apache's htaccess and htpasswd.
6. A common navigation bar on the left of every page for easy navigation.
7. Unique forms and reports requested by the client.
8. Authenticated users will be able to perform the following tasks:
 - ❖ Add devices, users, and help calls into the database.
 - ❖ View devices, users and their associative information.
 - ❖ View help call logs
 - ❖ Update devices, user, and help call logs.
 - ❖ Search on computer serial, Workstation ID, or UC property tag number for devices in the database.
 - ❖ Create/Add a new call
 - ❖ Edit an existing help call
 - ❖ Assign a call to another technical support member
 - ❖ Close an existing help call once the help call is resolved.
 - ❖ User access will be logged automatically.

4. Design and Development

4.1 Timeline

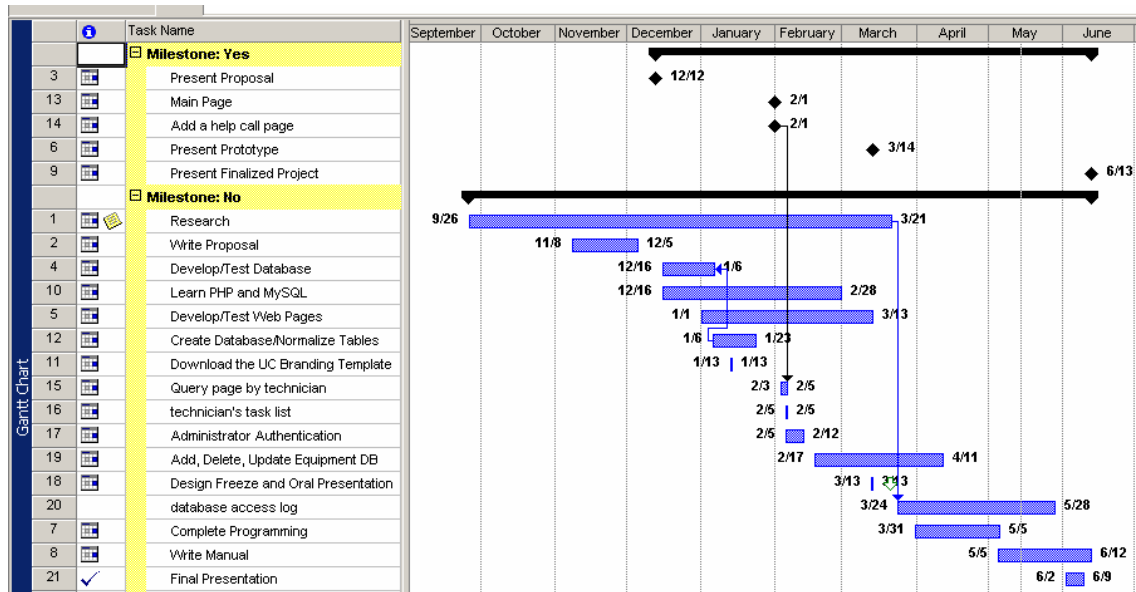


Figure 5. Project Timeline.

4.2 Budget

<i>Hardware</i>	<i>Cost</i>	<i>Source</i>
Dell Dimension 4100 Pentium III 866Mhz w/512 Mb 20Gb + 50Gb HDD (Server)	\$ 1,500.00	www.dell.com
Micron Client Pro CN Pentium III 933 MHz (Development PC)	1,500.00	www.micronpc.com
<i>Software</i>	<i>Cost</i>	<i>Source</i>
Mandrake 8.2	0.00	www.linuxiso.org
Apache 1.3.23	0.00	www.apache.org
MySQL 3.23.52	0.00	www.mysql.com
PHP Version 4.1.2	0.00	www.php.net
Macromedia Dreamweaver MX	345.99	www.buy.com
Premiumsoft Navicat 5.0	62.00	www.mysqlstudio.com
OpenSSL 0.9.6c	0.00	Included with Mandrake 8.2
mod_ssl 2.8.7	0.00	Included with Mandrake 8.2
SSH Secure Shell	116.00	www.ssh.com
Microsoft Office 2000 Professional	487.29	www.buy.com
Total	\$ 4,011.28	

Figure 6. Hardware and Software Budget

4.3 Software and Hardware Requirements

Software Requirements

- ❖ Mandrake 8.2 (included: Apache 1.3.23, PHP 4.1.2, MySQL 3.23.52)
- ❖ Macromedia Dreamweaver MX
- ❖ Microsoft Office 2000/XP Suite

Hardware Requirements

- ❖ Web/Database Server
- ❖ Development PC with internet access

5. Proof of design

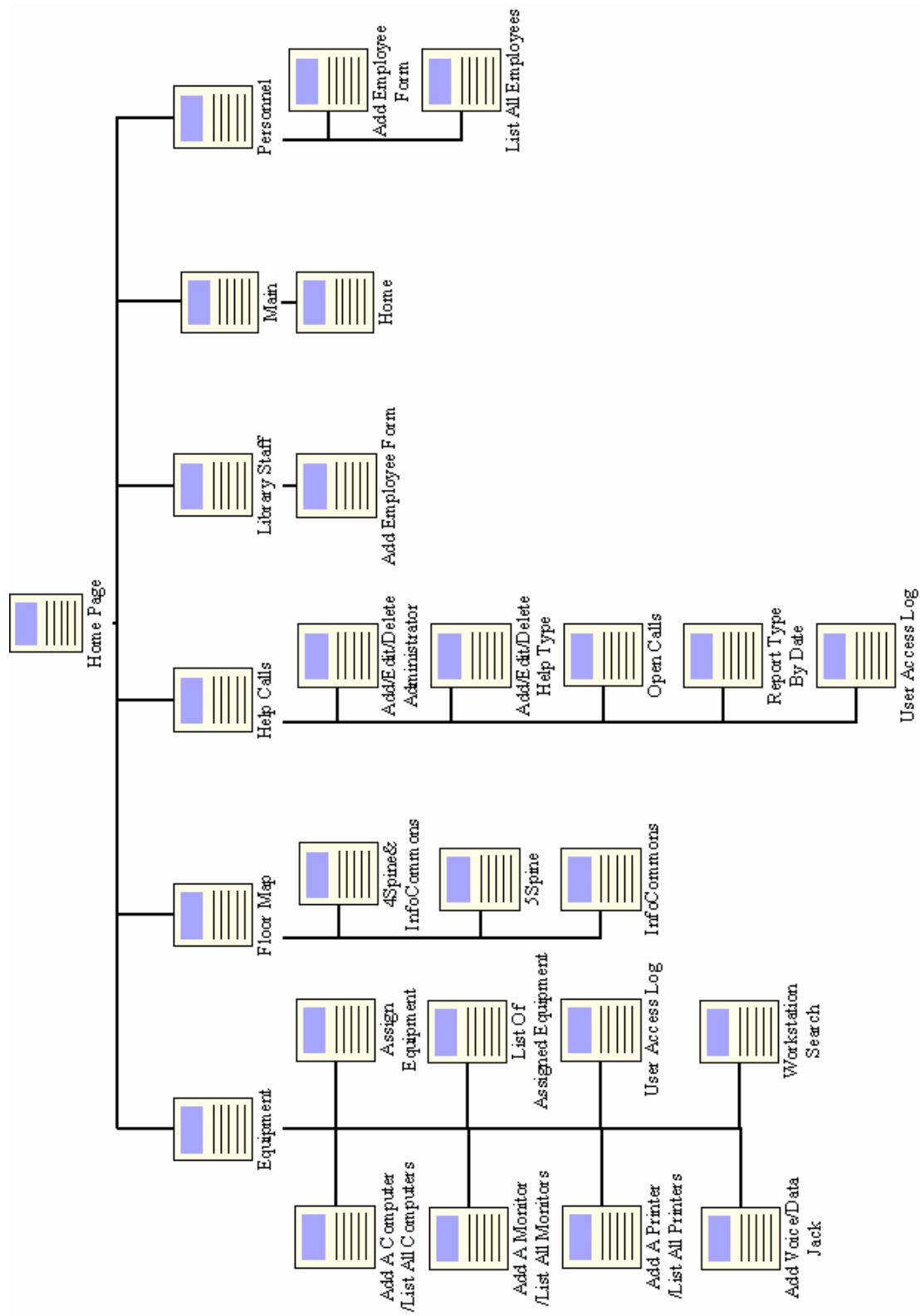


Figure 7. Library Information Technology Department Web Page Layout

5.1 Page Layout

The purpose of the Library Information Technology web site is to track and manage computer equipment inventory and help calls. I have developed a functional web application that is custom tailored to meet LIT's design needs for the amount of multimedia used on the web site. For this reason, the pages are not overly saturated with graphical layouts. Many of the pages throughout the web site have a similar page layout (See Figure 8.). Due to the size of the floor maps, the floor map layout pages have a somewhat different page layout. For example, this section is shown without the navigation menu on the left side, UC logo, and department title.

The navigational links are database driven, which makes editing them centralized. For example, if a link is added into the nodes table of the menu database then the link will appear on throughout the application.

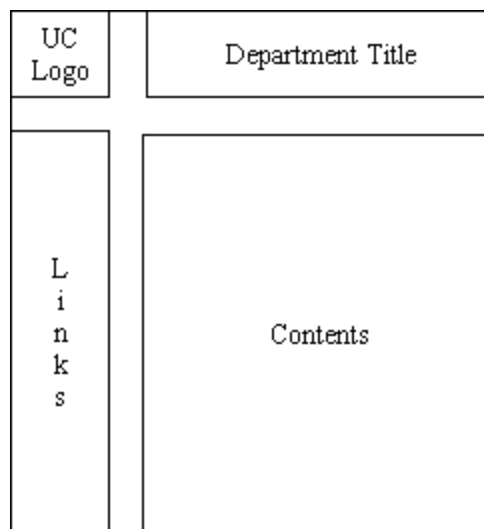


Figure 8. Page Layout

The department home page lists the department personnel and their specialized services (see Figure 9.). Each name is a hyperlink, if clicked it will launch the default email client application of that computer.

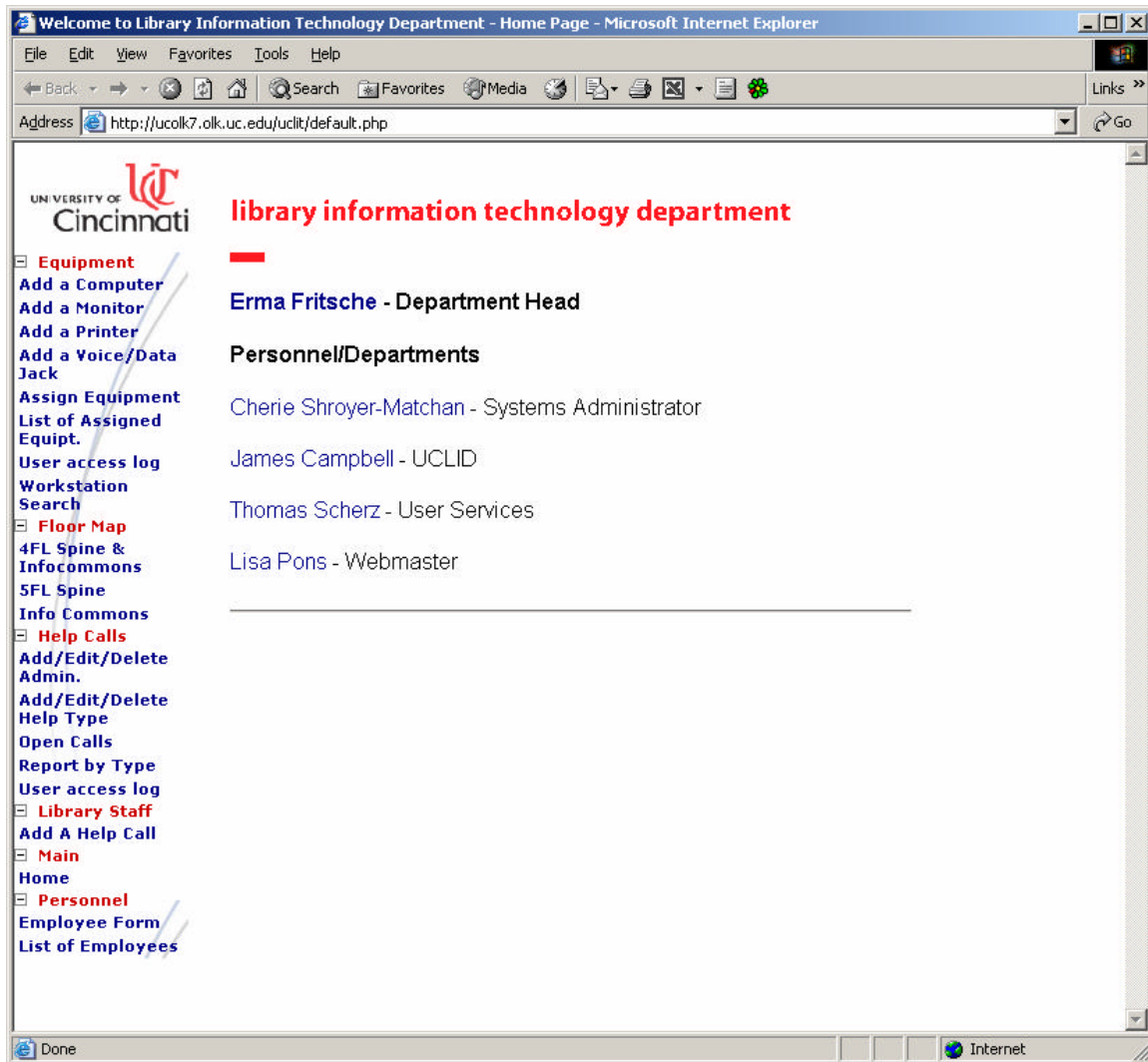


Figure 9. Home Page.

5.2 Library Staff Section

The library staff section contains a link to add a new help call. Future plans are to add a Do It Yourself (DIY) for step-by-step instruction on how to add/load paper into printer and how to change toner cartridges and other general task that requires less technical knowledge to execute.

Helpcalls Section - Add/Delete Help Call - Microsoft Internet Explorer

Address: http://ucolk7.olk.uc.edu/ucilit/show_addhelp.php

UNIVERSITY OF Cincinnati library information technology department

Add a New Help Call

* - Indicates required fields

Help Category:
Hardware *

User's Name:
Kevin Hsu *

Phone Number:
61745

Department:
Library IT *

Help Call Description:
Still yet another testing message *

Submit Help Call

Figure 10. Add a New Help Call Page.

Library staff may use the add a new help call page to report any computer malfunctions that they may have noticed or was reported by library patrons. When the “Submit Help Call” button is clicked, the form sends a post method request to `do_addhelpcall.php`, where the help calls are assigned to a full-time LIT staff member. This is done via a switch case code with hard coded full-time names and their corresponding help categories. An example of a submitted help call is shown in Figure 11.

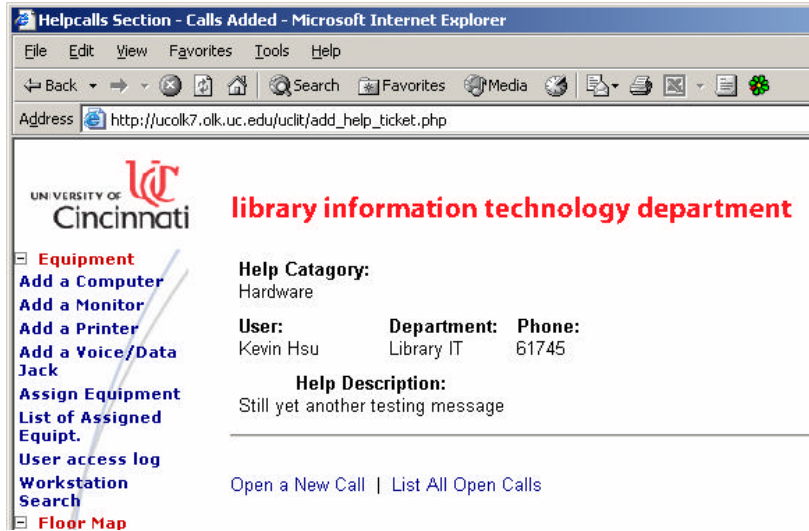


Figure 11. Submitted Help Call.

5.3 HTTP Authentication Login

Links clicked under the Equipment, Help Calls, and Personnel sections will launch the secure login screen (See Figure 12.), if the user has not been logged in. The user is authenticated via Apache's HTTP authentication. In order, for a user to be able to login, a user account must be created via a command line on the web server. Additional, details on how new users are created using htpasswd can be found in Appendix B.

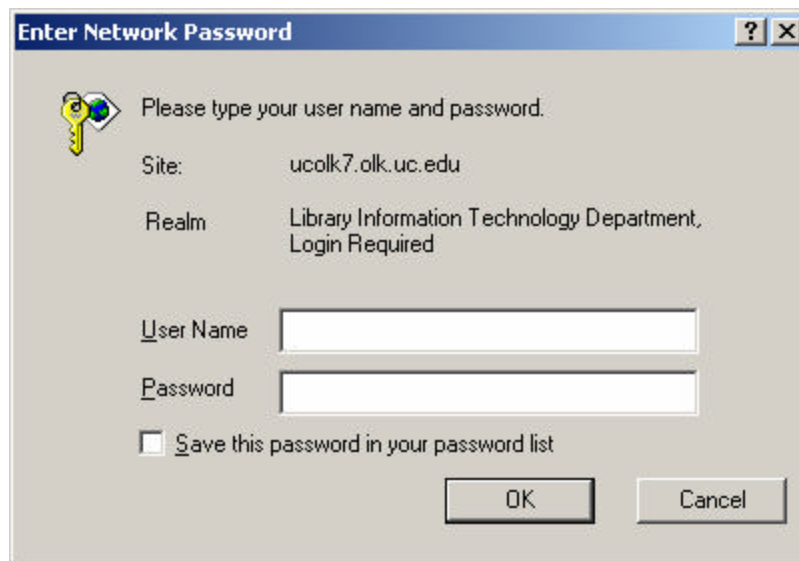


Figure 12. HTTP User Authentication Login Screen.

The users of the system will be allowed three tries to enter into the text fields the correct User name and password to gain access to the protected pages. Otherwise, users will be redirected to the login failed page (See Figure 13.).



Figure 13. Failed Authorization Request Page.

5.4. Equipment Section

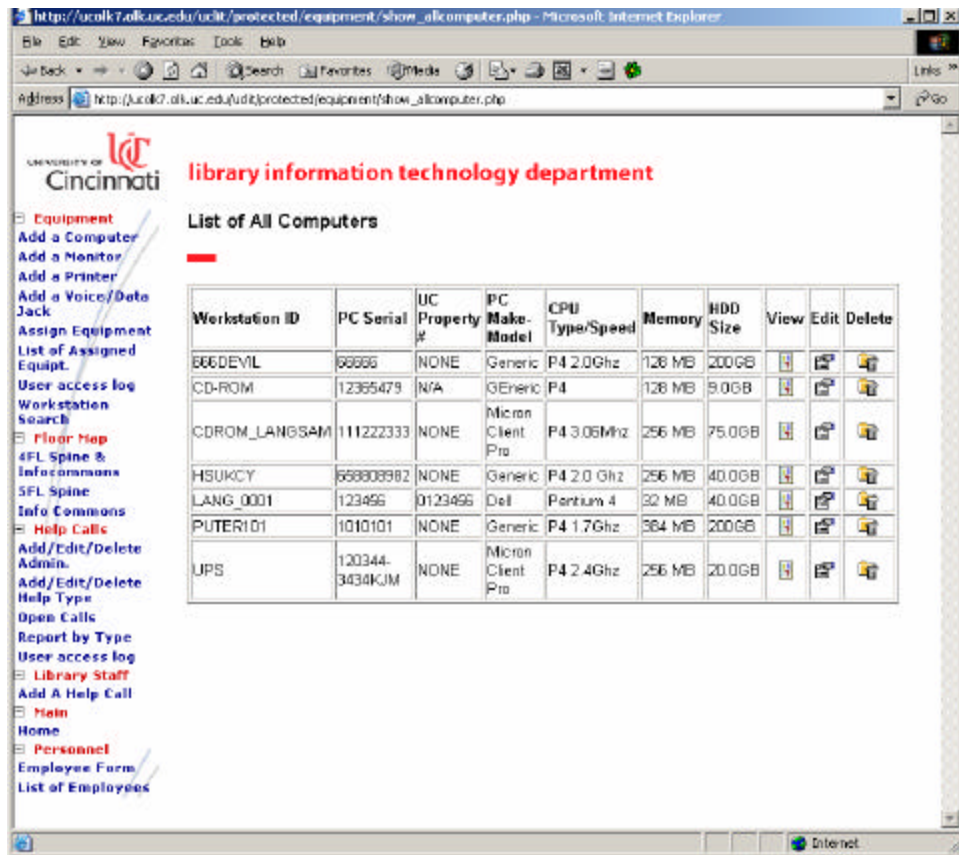
The Add a Computer page allows users to add, edit and delete. Text field validations are performed through JavaScript to ensure required fields are not left blank. Figure 45. illustrates the JavaScript popup error message. Once all required fields are filled in and the submit button is clicked, a request to “process_computer_add.php” is called. The information gathered is then processed and inserted into the computer table. Figure 14 illustrates the Add a Computer page.

Figure 14. Add Computer Page.

The “List of All Computers” page displays the entire computer table (see Figure 15.). The table is sorted by Workstation ID, which was a request of the LIT administrator. The amount of data gathered from the Add a Computer page would have created a scrolling issue. For this reason, I have added a view link to redirect the user to a detailed view of all the data gathered from the Add a Computer page.

The edit and delete columns contain gif images that are clickable links by the use of HTML anchor tags. Once a user clicks on the link, an URL with computer serial and user action is sent to the add computer page, where it performs a test to see if the

\$_GET['computer_serial'] is set. If the \$_GET['computer_serial'] was set, then the page was requested by another page.



The screenshot shows a web browser window with the address bar displaying `http://ucolick7.ollu.edu/ucit/protected/equipment/show_allcomputer.php`. The page header includes the University of Cincinnati logo and the text 'library information technology department'. The main content area is titled 'List of All Computers' and contains a table with the following data:

Workstation ID	PC Serial	UC Property #	PC Make-Model	CPU Type/Speed	Memory	HDD Size	View	Edit	Delete
666DEVIL	66666	NONE	Generic	P4 2.0Ghz	128 MB	200GB			
CD-ROM	12365479	N/A	Generic	P4	128 MB	8.0GB			
CDROM_LANGSAM	111222333	NONE	Micron Client Pro	P4 3.06Mhz	256 MB	75.0GB			
HSUKCY	668808982	NONE	Generic	P4 2.0 Ghz	256 MB	40.0GB			
LANG_0001	123456	0123456	Dell	Pentium 4	512 MB	40.0GB			
PUTERID1	1010101	NONE	Generic	P4 1.7Ghz	384 MB	200GB			
UPS	120344-3434KJM	NONE	Micron Client Pro	P4 2.4Ghz	256 MB	20.0GB			

Figure 15. List of All Computers Page.

The “Add a Monitor” page consist of four required fields marked by red asterisks (see Figure 16.). A link is provided below the submit button to list all monitors within the monitor’s table.

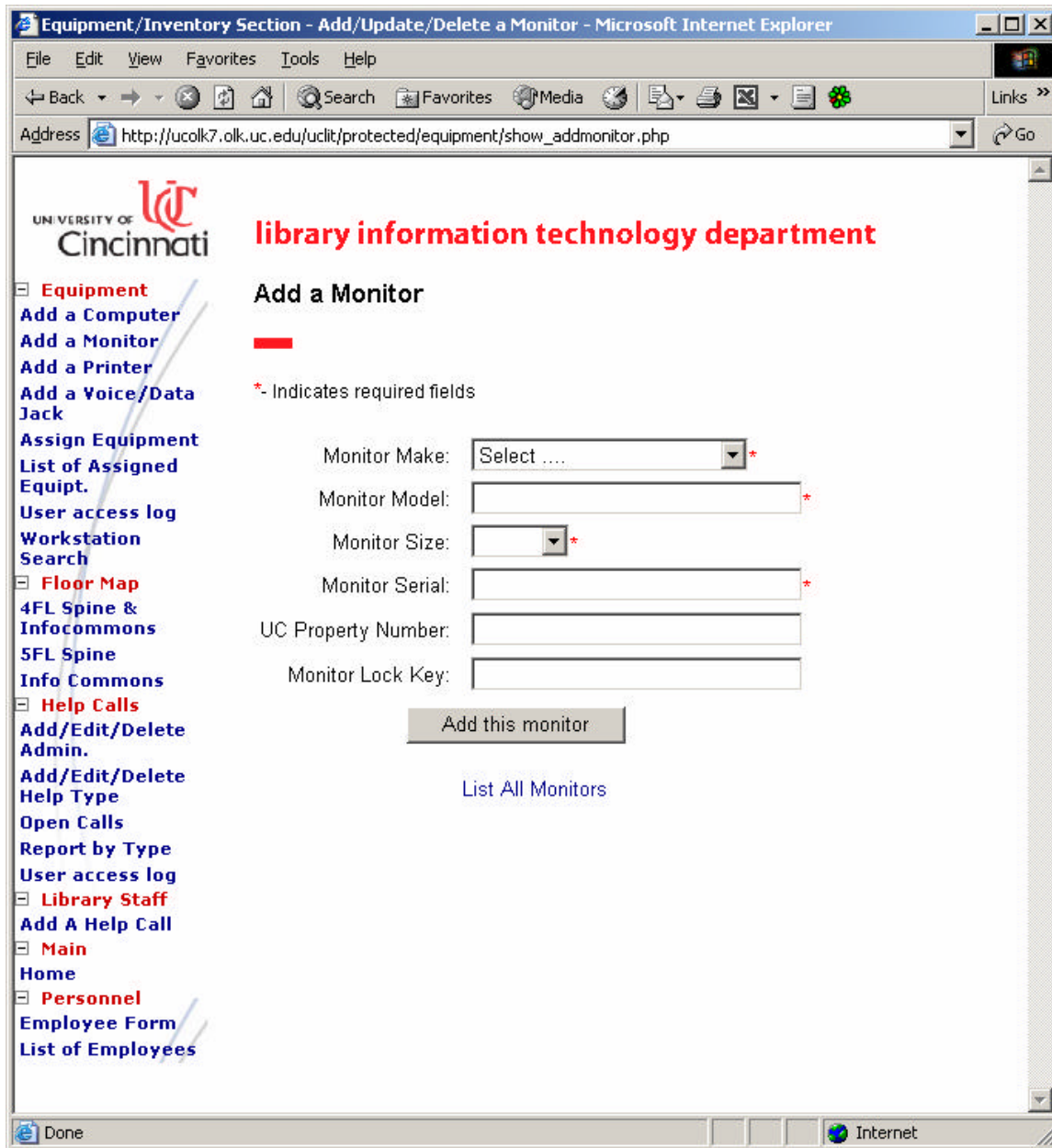


Figure 16. Add a Monitor Page.

For example, when a user clicks on the link below the submit button the “show_allmonitor.php” page will be requested. A SQL select statement is run during each request. A report is generated and sorted by the monitor size and serial number (see Figure 17.).

The screenshot shows a web browser window with the address `http://ucolk7.olk.uc.edu/ucilit/protected/equipment/show_allmonitor.php`. The page header includes the University of Cincinnati logo and the text "library information technology department". The main content area is titled "List of ALL The Monitors" and contains a table with the following data:

Monitor Size	Monitor Make	Monitor Model	Monitor Serial#	UC Property #	Lock Key	Edit	Delete
15inch	MAG	*	08229517-03	404-111520771	*		
17inch	DELL	*	1053769	NONE	*		
17inch	DELL	*	122099-0019	NONE	*		
17inch	DELL	*	17inch80-DCG3W-69	NONE	*		
15inch	MAG	*	17inch80R-A 5TNB-89	NONE	*		
17inch	DELL	*	17inch80R-A5MR-69	NONE	*		
17inch	DELL	*	17inch80R-A5TN1	NONE	*		
17inch	DELL	*	17inch80R-A5TN5-89	NONE	*		
17inch	DELL	*	17inch80R-DC9V5-69	NONE	*		
17inch	DELL	*	17inch80R-DC9V8-69	NONE	*		
17inch	DELL	*	17inch80R-DC9VD-69	NONE	*		
17inch	DELL	*	17inch80R-DC9VG-69	NONE	*		
17inch	DELL	*	17inch80R-DC9VH-69	NONE	*		
17inch	DELL	*	17inch80R-DC9VS-69	NONE	*		
17inch	DELL	*	17inch80R-DC9VV-69	NONE	*		
17inch	DELL	*	17inch80R-DCBC2-69	NONE	*		
17inch	DELL	*	17inch80R-DCBC5-69	NONE	*		
17inch	DELL	*	17inch80R-DCBCU-69	NONE	*		
17inch	DELL	*	17inch80R-DCBCX-69	NONE	*		
17inch	DELL	*	17inch80R-DCBCY-69	NONE	*		
17inch	DELL	*	17inch80R-DCBD0-69	NONE	*		

Figure 17. List of All The Monitors Page.

The “Add a Printer” page has four required fields which the user must fill out or else it will generate a JavaScript alert box (See Figure 45.).

Equipment/Inventory Section - Add a Printer - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print Links

Address http://ucolk7.olk.uc.edu/ucit/protected/equipment/show_addprinter.php Go

UNIVERSITY OF Cincinnati library information technology department

Equipment
 Add a Computer
 Add a Monitor
 Add a Printer
 Add a Voice/Data Jack
 Assign Equipment
 List of Assigned Equipt.
 User access log
 Workstation Search

Floor Map
 4FL Spine & Infocommons
 5FL Spine
 Info Commons

Help Calls
 Add/Edit/Delete Admin.
 Add/Edit/Delete Help Type
 Open Calls

Add a Printer

* - Indicates required fields

Printer Serial: *

Printer Name: *

Printer Server:

Printer Make: *

Printer Model: *

UC Property Number:

Add this monitor

[List All Printers](#)

Figure 18. Add a Printer Page.

The “List of All Printers” page provides both edit and delete functions by the way of a clickable image link. As users click on either the edit or delete functions an URL with the selected record data is appended to the end of the “show_addprinter.php”.

Equipment/Inventory Section - View All Printers - Microsoft Internet Explorer

Address: http://ucolk7.olk.uc.edu/ucilit/protected/equipment/show_allprinter.php

UNIVERSITY OF Cincinnati **library information technology department**

List of ALL The Printers

Printer Serial#	Printer Name	Printer Server	Printer Make	Printer Model	UC Property #	Edit	Delete
058WK002	LOCAL	NONE	EPSON	ACTIONLASER 1600	0302810		
1F8E351374	LOCAL	NONE	EPSON	LQ-570+	NONE		
1F8E351375	LOCAL	LANG_PER_Q	EPSON	LQ 570+	NONE		
1F8E35161	LOCAL	NONE	EPSON	LQ-570+	NONE		
1F8E497131	LOCAL	NONE	EPSON	LQ-570	NONE		
4041906532	LOCAL	NONE	EPSON	LQ-870	DA00095		
40U0050458	LOCAL	NONE	EPSON	LQ-870	NONE		
40U0050477	LOCAL	NONE	EPSON	LQ-870	NONE		
40U0050483	NONE	NONE	EPSON	LQ-870	NONE		
40U0058447	LOCAL	NONE	EPSON	LQ-870	NONE		
40U1168513	LOCAL	NONE	EPSON	LQ 870	NONE		
40U116853	LOCAL	NONE	EPSON	LQ-870	NONE		
40U118507	LOCAL	NONE	EPSON	LQ-870	NONE		
C06002015XZ8612335	LOCAL	POSEIDON	EPSON	LQ-870	NONE		
C0600211P88417119	LOCAL	LIBSTAFF	EPSON	LQ-870	NONE		
0315803	LOCAL	NONE	HEWLETT-PACKARD	LASERJET 5M	C3917A		

Figure 19. List of All The Printers Page.

The “Add a Phone and Data Jack” page only has two required fields. Because most workstations are plugged into the data jacks to gain network connectivity and in most cases public workstations do not have the phone jacks activated so therefore, it does not require that text field to be entered.

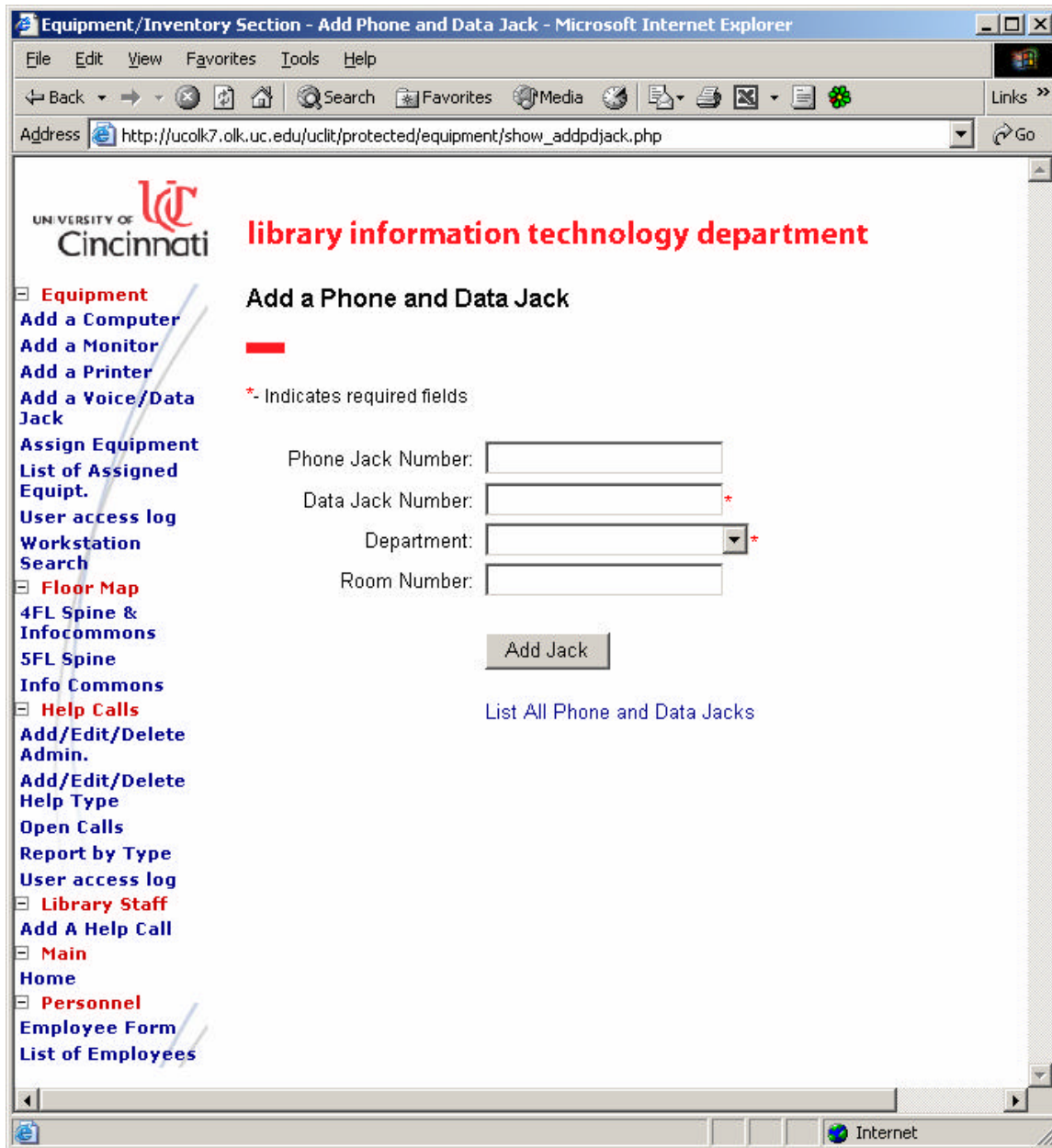


Figure 20. Add a Phone and Data Jack Page.

“List of All The Phone & Data Jacks” page is no exception to the list computers, monitors, printers, and employee pages, where required fields are validated by JavaScript conditional statements. An alert box appears when a required field is left blank. The edit and delete columns contains clickable gif images. When the images are clicked the form, “show_addpdjack.php,” is populated with the selected record (see Figure 21.).

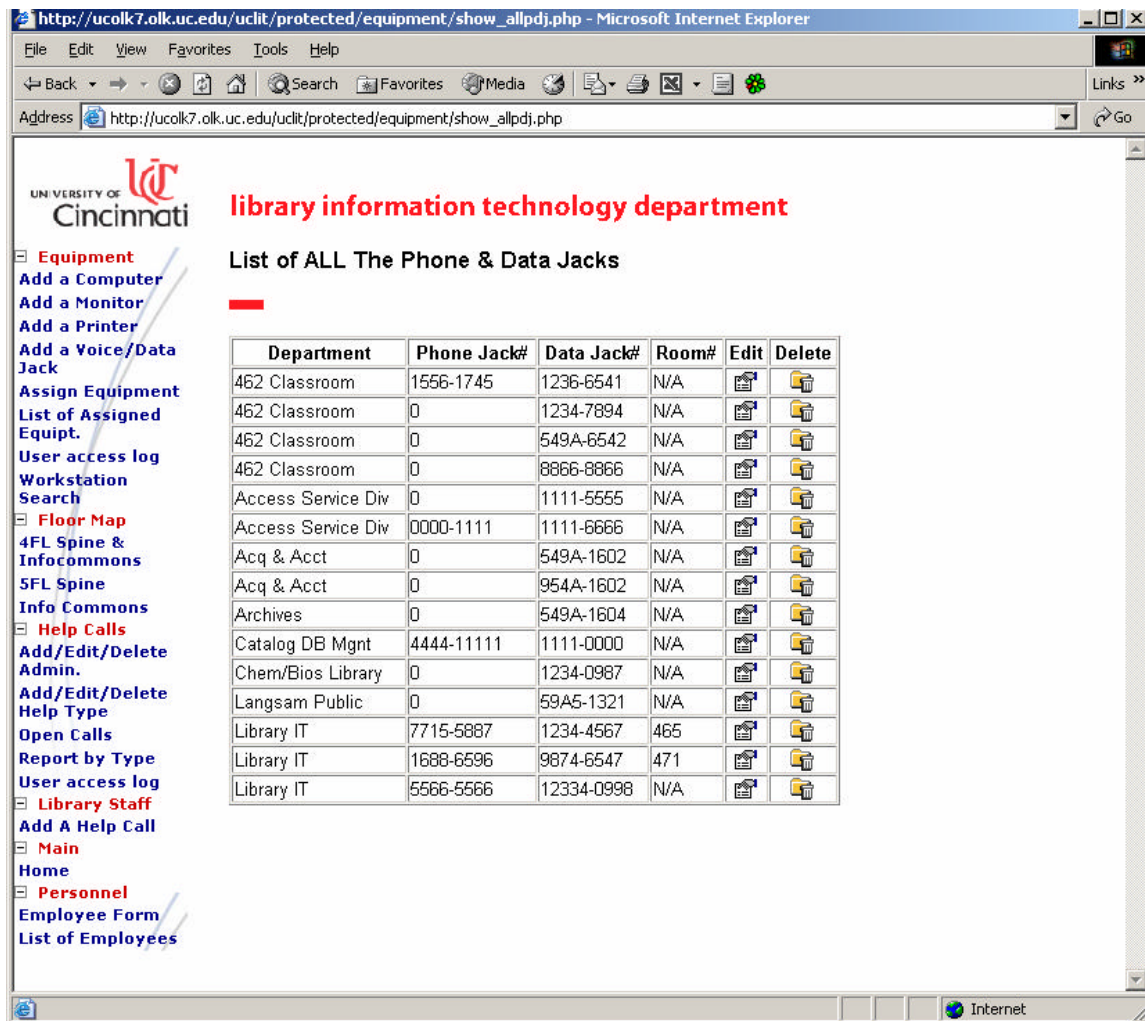


Figure 21. List of All The Phone & Data Jacks Page.

The “Assign Equipment to Employee” page consists of dropdown menu lists. The lists allow users to select personnel ID or equipment that is stored in the equipment database. If the user selects a computer or monitor that has already been assigned, then it will display the information of the computer or the monitor to whom it is assigned to in the “process_equipdetail.php” page similar to Figure 23.

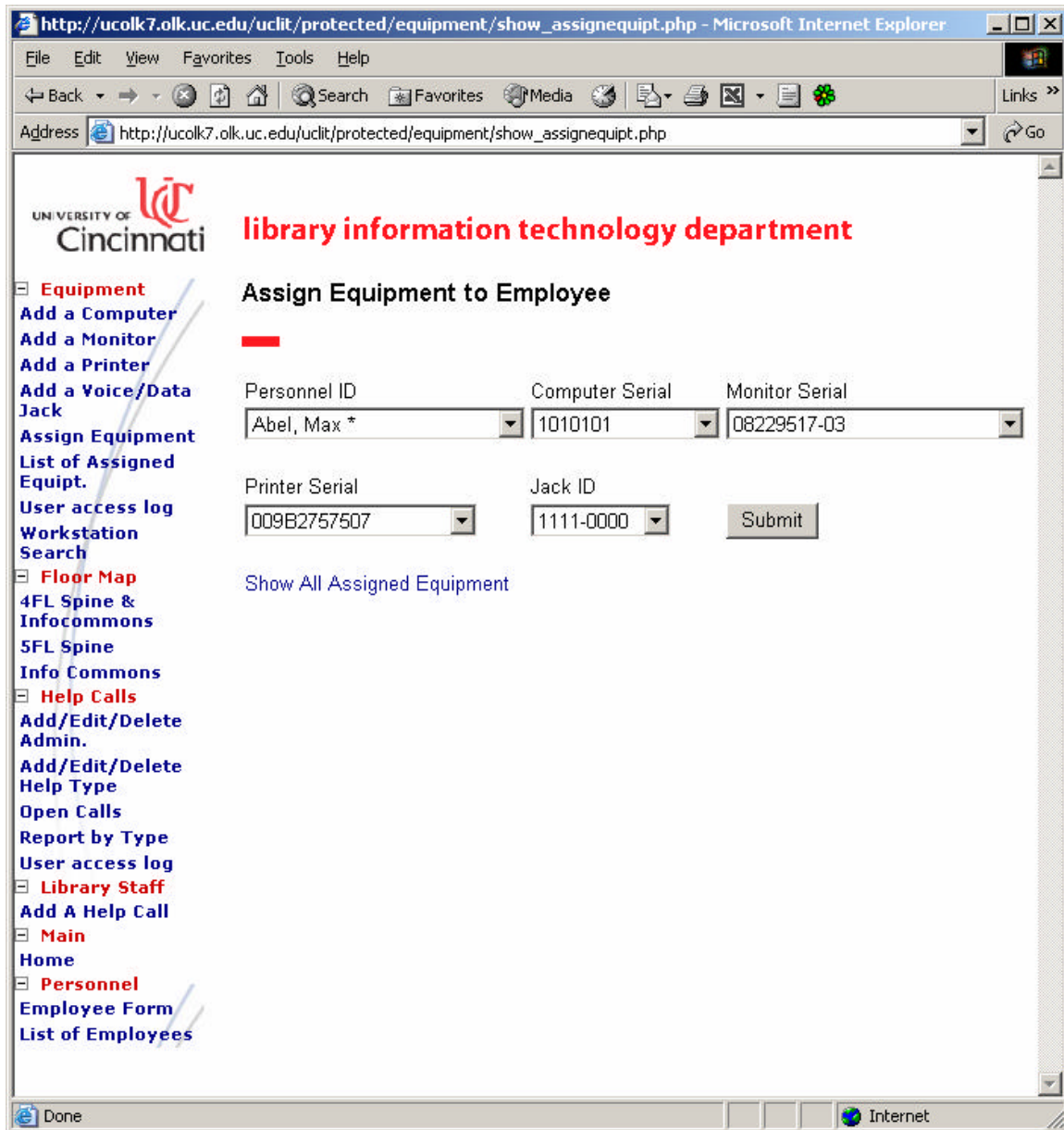


Figure 22. Assign Equipment to Employee Page.

The “Show All Assigned Equipment” link when clicked directs the user to the “List All Assigned Equipment” page (See Figure 23.). This page displays the equipment and the employee who it was assigned to. The table is sorted by “Detail ID”. All the column fields are clickable links. When clicked it sends them to the edit page.

http://ucolk7.olk.uc.edu/ucilit/protected/equipment/show_alllequiptdetail.php - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print

Address http://ucolk7.olk.uc.edu/ucilit/protected/equipment/show_alllequiptdetail.php Go

UNIVERSITY OF Cincinnati library information technology department

Equipment
 Add a Computer
 Add a Monitor
 Add a Printer
 Add a Voice/Data Jack
 Assign Equipment
 List of Assigned Equipt.
 User access log
 Workstation Search

Floor Map
 4FL Spine & Infocommons
 5FL Spine
 Info Commons

Help Calls
 Add/Edit/Delete Admin.
 Add/Edit/Delete Help Type
 Open Calls
 Report by Type
 User access log

Library Staff
 Add A Help Call

Main
 Home

Personnel
 Employee Form
 List of Employees

List ALL Assigned Equipment

Detail ID	Employee ID	PC Serial	Monitor Serial	Printer Serial	Jack ID
1	Abel, Max *	1010101	08229517-03	009B2757507	1111-0000
2	Adler, Pam *	111222333	1053769	0315803	1111-5555
3	Albrecht, Cheryl *	120344-3434KJM	122099-0019	058WK002	1111-6666
4	Allen, Frances *	123456	17inch80-DCG3W-69	1F8E351374	12334-0998
5	Bach, Pamela *	66666	17inch80R-DC9VD-69	40U0050458	549A-6542
6	Cook, Ron *	658808982	17inch80R-DCBW-69	009B2757507	549A-1604

Assign Equipment Form

Figure 23. List All Assigned Equipment Page.

Once the “Detail ID” is clicked the user will see a more descriptive detailed view of the assigned equipment (See Figure 24.). The images in each table row are clickable links, when clicked the link directs users to the edit page.

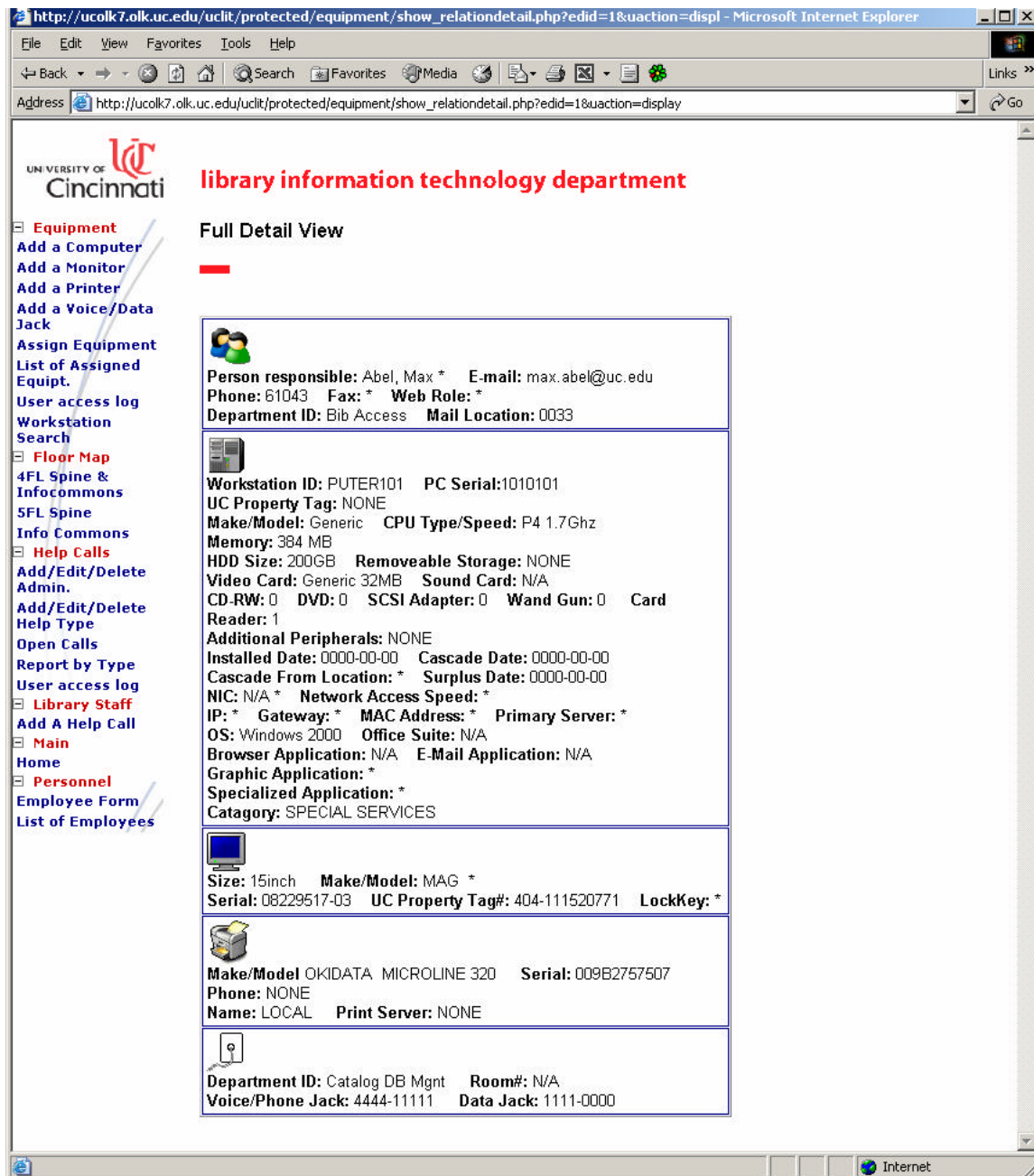


Figure 24. Full Detail Page.

The “Equipment User Access Log” page tracks logged in users add, edit, and delete actions. This was requested by the LIT’s system administrator. Figure 25. illustrates the Equipment User Access Log.

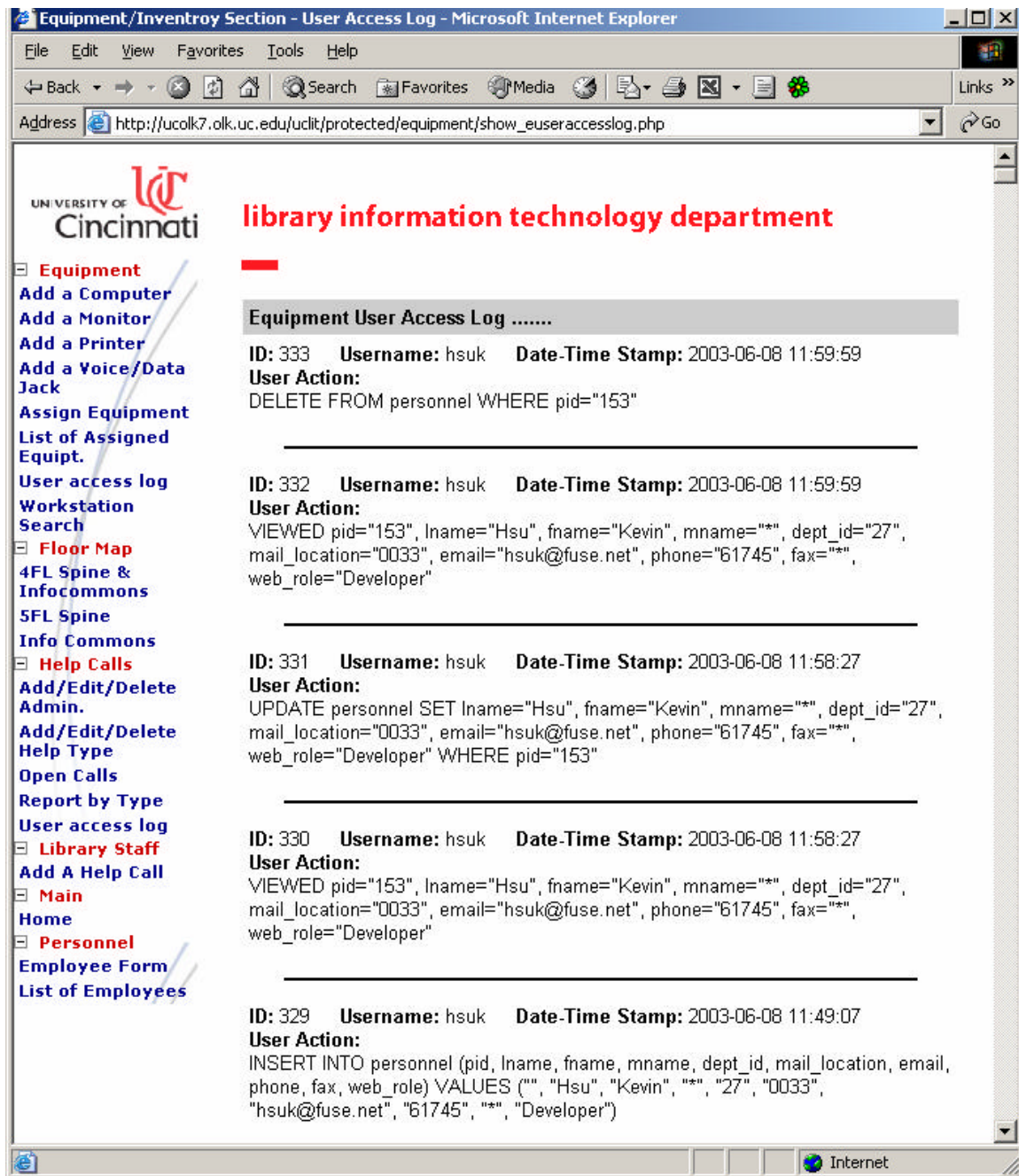


Figure 25. Equipment User Access Log Page.

The workstation lookup page consists of a text field for the user to enter a search string. The list/menu dropdown control allows the user to search on specific column field in the computer table (see Figure 26.).

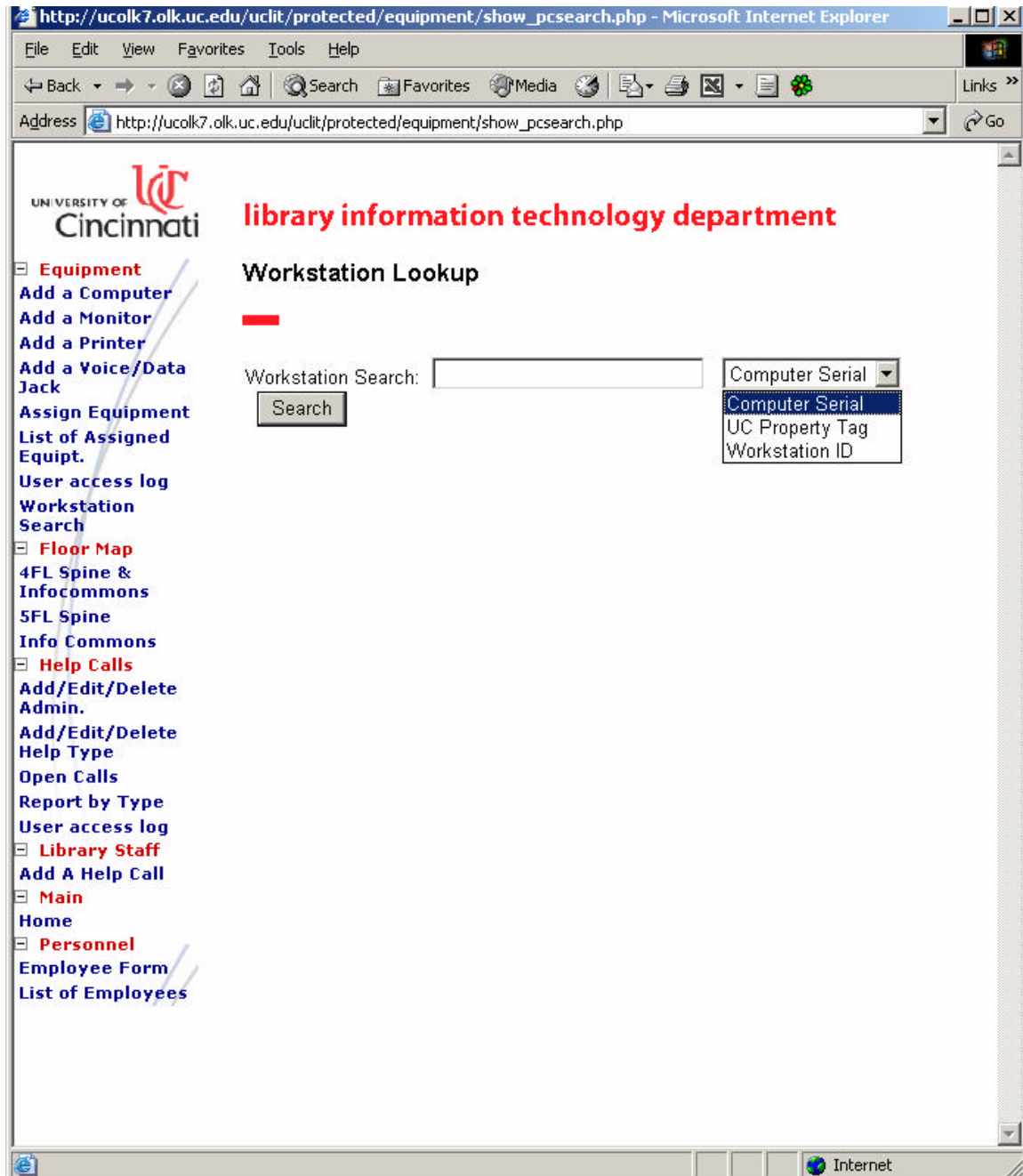


Figure 26. Workstation Lookup Page.

When the search string is processed and added to the SQL select statement, it is then queried. If no record matching the search criteria was found a descriptive error message will generate (See Figure 27.).

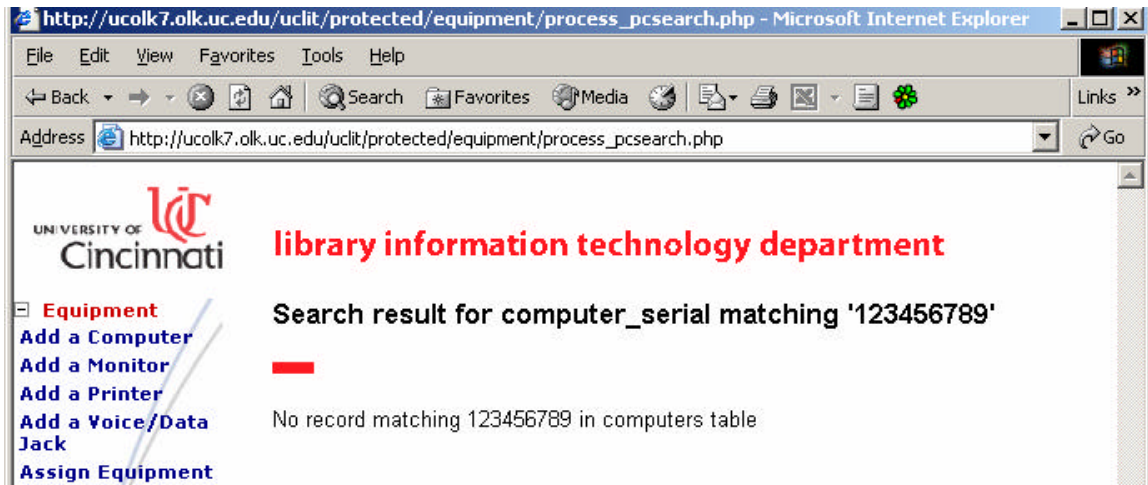


Figure 27. Error Search Result for Computer Page.

If the query returns a match then partial detail of the computer data is displayed (see Figure 28.). The PC serial is a clickable link which populates the “show_addcomputer.php” form when clicked.

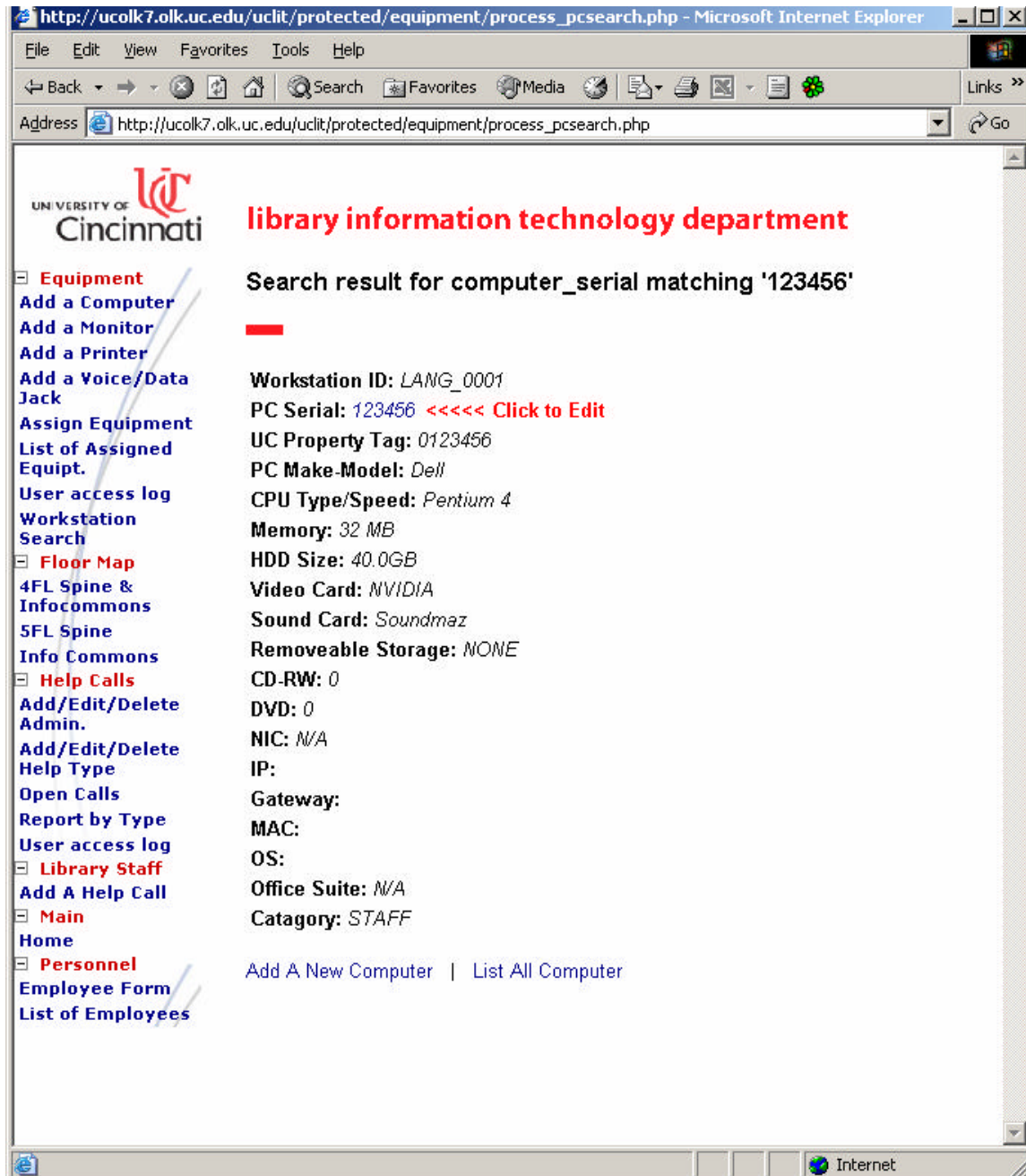


Figure 28. Search Result for Computer Serial Matching Page.

5.5 Floor Map

Floor maps were provided to give both the library staff and the administrator of the LIT support staff a fast and easy way to find where a workstation is located (see Figure 29, 30, 31.). The floor maps are of Langsam library's public computers reserved for library patrons.

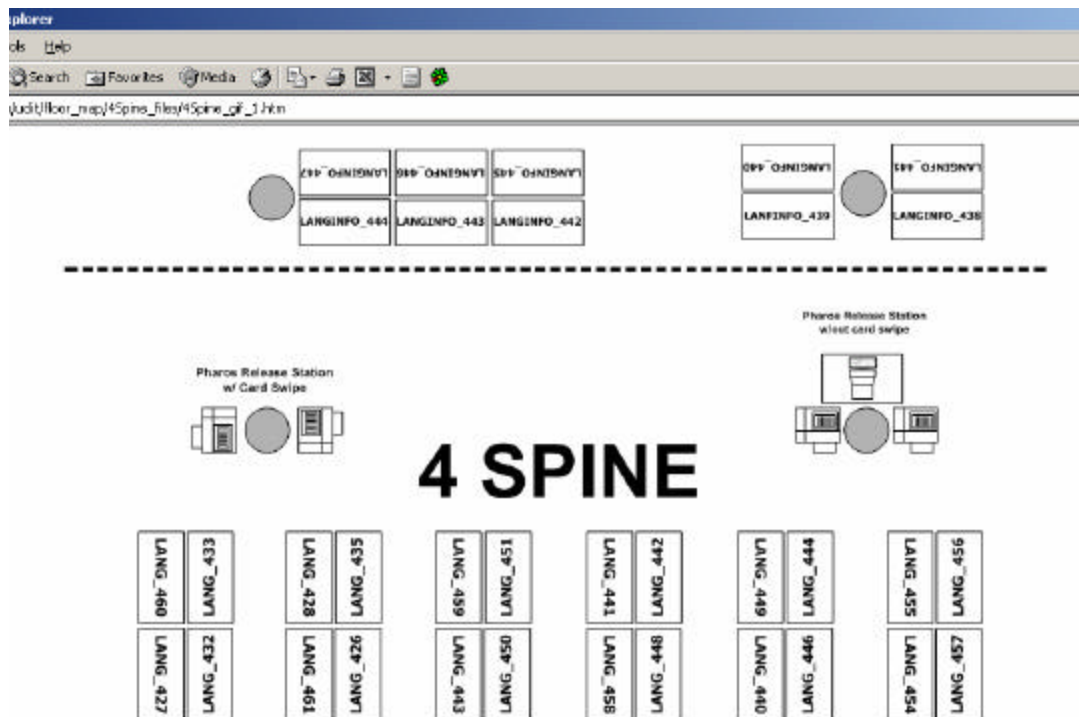


Figure 29. 4Spine Floor Map Page.

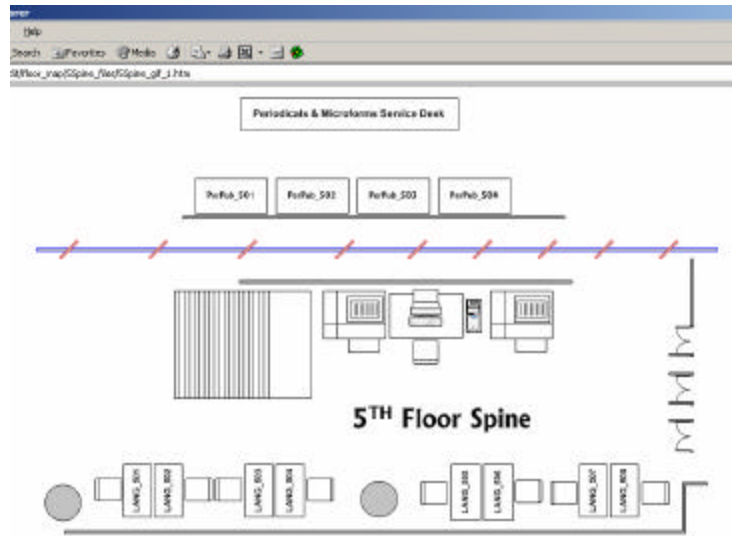


Figure 30. 5th Floor Spine Floor Map Page.

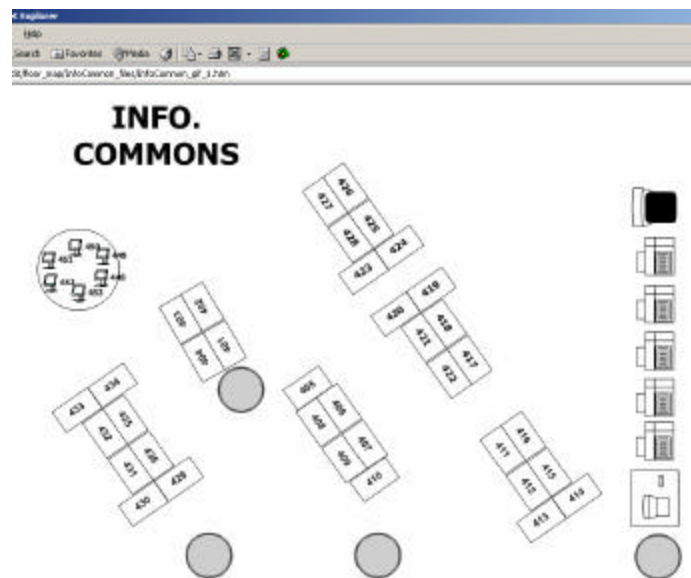


Figure 31. Info. Commons Floor Map Page.

5.6 Help Calls Section

The “Add A New Help Category” page serves multiple functions. When the page is requested it serves as an add a new help category page on the upper part of the content area and on the bottom area it serves as a list of all help category types sorted by help category.

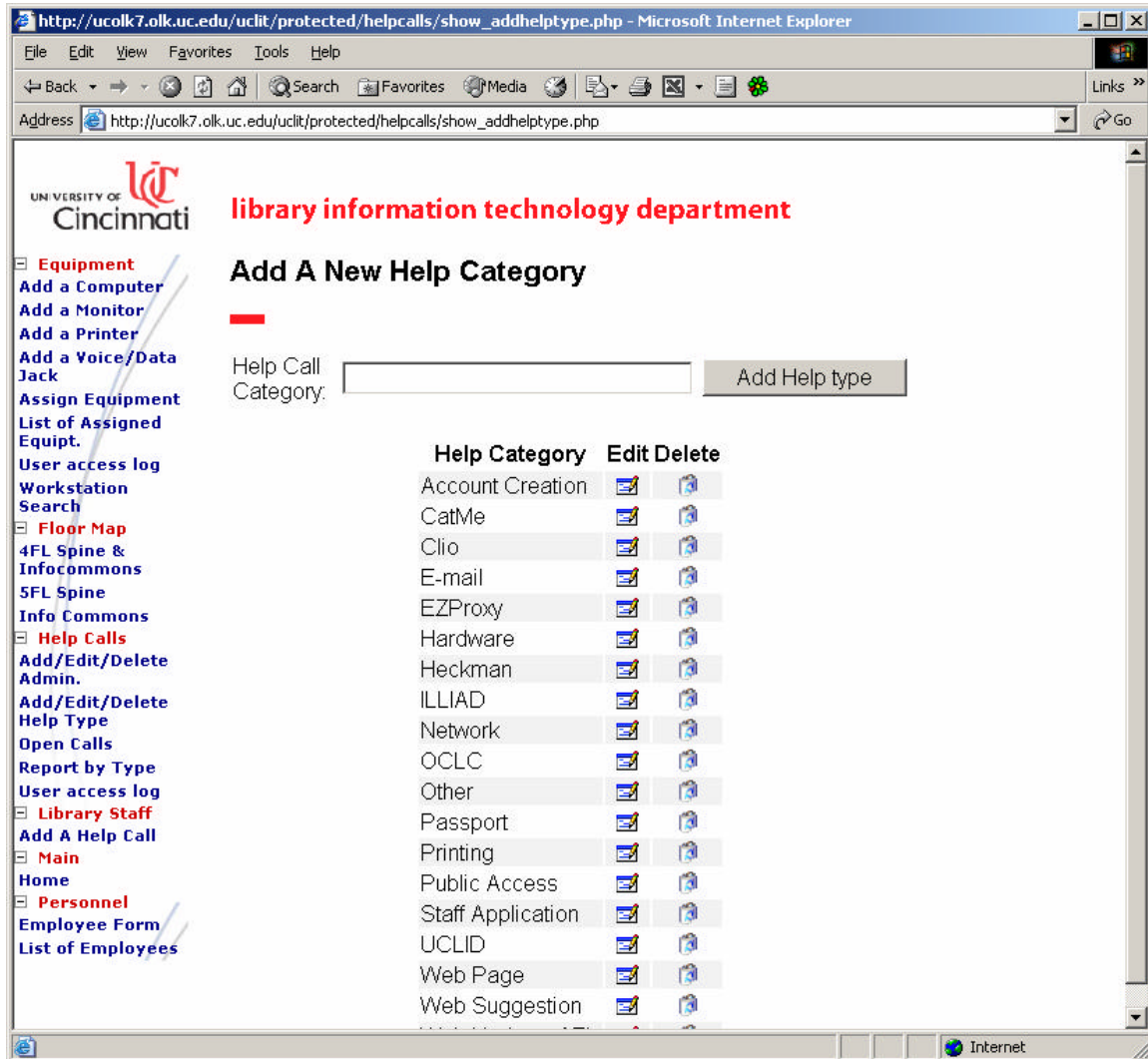


Figure 32. Add a New Help Category Page.

Administrators may add, make edits, or delete a help call type that is department specific. The edit and delete uses the same text fields and submit button to perform the requested task. “process_helptypeud.php” is requested once the submit button is clicked.

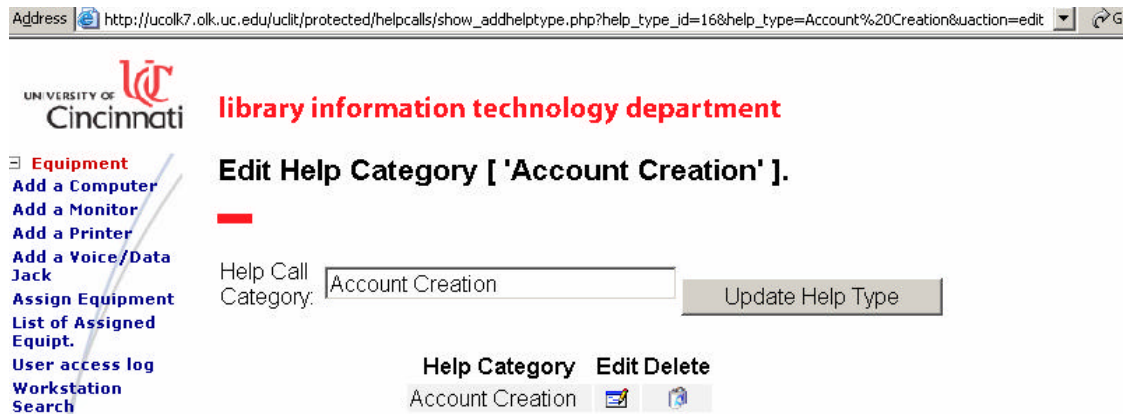


Figure 33. Edit Help Category Page.

As a user clicks on any of the delete image via clickable link in the “show_addhelptype.php” page, a URL with the selected record is appended to the end of the URL address textbox. The help category text field and submit button has been pre-assigned a value based on which link is click in the “show_addhelptype.php” (See Figure 34.).

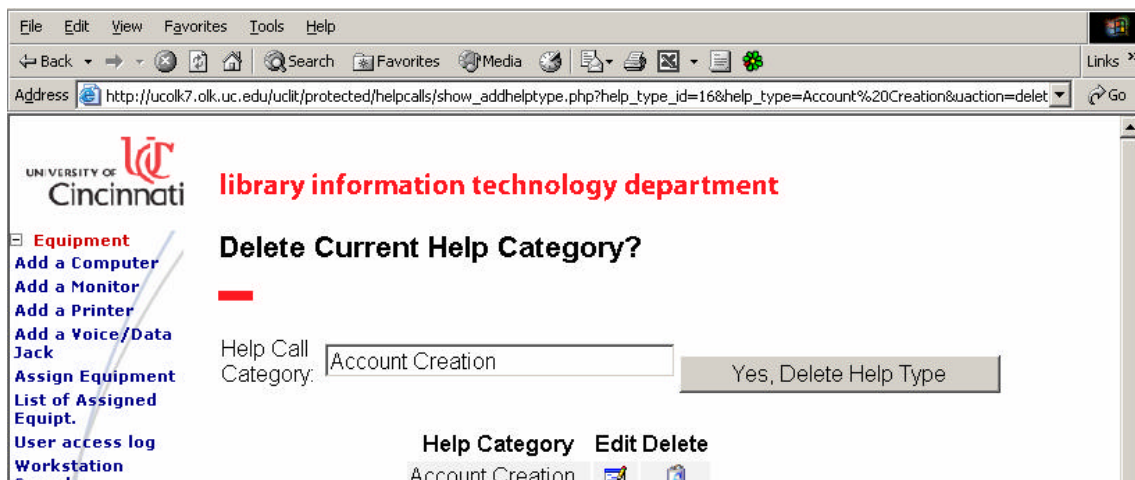


Figure 34. Delete Help Category Page.

Header and button value are dynamically assigned a value depending on what the user clicked on. When a user clicks on an edit or delete link, it sends an URL with the respected help type id, help type name and a user action as to what the user would like to do with the selected record to the “show_addhelptype.php” page. A GET method is then performed to grab the assigned value from the URL address bar. Conditional statements then check the value of user action. If user action is blank, meaning add help type to table. Second, check if help type exist in the help type table. If not exist then add to help type table. If user action equals edit then assign header value and button value then run update query. If query is successful then display an update successfully message and a link to go back to add, edit, or delete help type page.

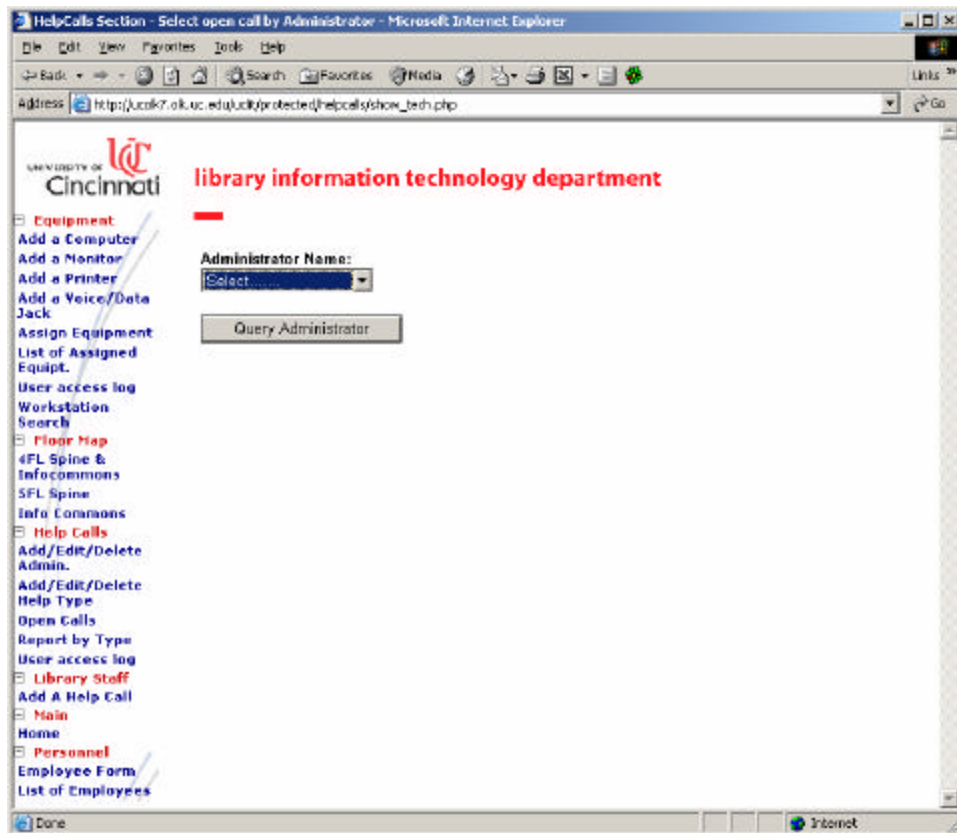


Figure 35. Open Calls Page.

As user selects an administrator from the list/menu dropdown control and “Query Administrator” button is clicked the “sel_by_tech.php” page performs a SQL select query on the log table on the “helpcalls” database where it sorts the result by date, then administrator name.

Date Arrived	Department	Assigned To	Description	Posted By	Phone	Edit	Delete
2003-05-07	Multimedia	Chetura Nagahawatts	We would like to add another hard drive to the Blue editing suite. I have the hard drive at my desk, thanks to Larry Jay Sennard Operational Services Manager Multimedia Services University Libraries University of Cincinnati Phone: (513) 556-1888 Fax: (513) 556-0001 Pager: (513) 209-4614	Sinnard, Jay (SINNARDJ@UCMAIL.UC.EDU)			
2003-05-16	Bib Access	Cherie Matchan	Dear Library Help: The clock on my computer is off by about 2 minutes compared to Kooner's time. Can you fix this for me again? Thanks in advance! Sincerely, Katrina M. Bowling-Bergman	Bowling, Katia (bowlink)			
2003-05-27	Acquisitions and Accounting	Cherie Matchan	Cherie, This morning we had three ILL requests waiting in our office email. These requests were generated using the old ILL request forms on May 23, 24, and 26. Please either redirect the old URL to our new ILLiad page or remove the link to our old request forms ASAP. Thanks! Tom	Tom White	556-1480		
2003-05-28	Engineering	Cherie Matchan	Doing network access to ones.uc.edu server. Not sure what directories are needed - for purpose of sending files to the server.	Dorothy F. Byers	556-3352		
2003-06-02	Acquisitions and Accounting	Cherie Matchan	Some, if not all, of the C&D Libraries are having problems accessing the ILLiad Client Material is beginning to arrive that needs to be processed in ILLiad, but Math, Engineering, Classics, and CRC are all experiencing problems with the ILLiad Client icons. They apparently aren't linked to ILLiad. To the inexperienced (me), this seems like a problem that should be easy to correct. I would also imagine that correcting this would be a high priority for the IT staff. Unfortunately, the lot of feedback I'm getting from staff indicates a rather relaxed response from Library IT. Please investigate and get these problems corrected ASAP. Thanks! Tom 556-1480	Thomas White	556-1480		
2003-06-06	Bib Access	Cherie Shoyat	Dear Library Help: The student worker in Bib. Access who works with OCLC CATME has had problems with the sessions locking up. She has told me that this is an on-going problem with the computers at the student work stations. She doesn't seem to have a problem using the staff terminal located in the corner by the windows. How might this problem be fixed? Sincerely, Katrina M. Bowling-Bergman	Bowling, Katia (bowlink)			
2003-06-06	Acq & Acct	Cherie Shoyat	I was able to access the ILLiad Client from our ILL Student Assistant workstation this morning, but when I tried to send pickup notification emails to our customers the following error window appeared: Cannot Open File: I:\ill\edemail\mchiora.txt. Not an emergency-- just check it out when convenient. Thanks! Tom Acq & Acct 480 Langsam	White, Thomas (white)			
2002-11-07	IDS	Ernie Fitzche	IDS Staff Workstation still on Hub.	Toby Heidtmann	61895		
2003-03-03	Classics	Ernie Fitzche	I now see a need for a generic Classics e-mail that all staff can use to send out notices -- such as announcements to pick up photocopies received on ILL. (There is no record placed in UCUD for these so we can't just send them in like the books we did at ILL.) We now have too spend time phoning people to pick these up, and ILL DEpt has advised us that they have had better luck just sending an e-mail. Actually, I think such an account may have been established a few years ago for another purpose -- and a purpose that was abandoned. We wanted to use it as a way to keep all students advised of new reminders of procedures.	Wellington, Jean	61806		

Figure 36. Help Calls By Engineer Page.

Once the edit link for the respected record is clicked, the primary id of the help call is appended to the URL where that information is used to perform a SQL select query on the “help_id” and later populate the text fields. The User, department, name Date arrived field was requested by the network administrator to be disabled.

Helpcalls Section - Edit Call - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Links

Address http://ucolk7.olk.uc.edu/ucilit/protected/helpcalls/edit_ticket.php?help_id=844 Go

UNIVERSITY OF Cincinnati library information technology department

Edit Help Call Ticket ['844']

☐ Completed
 Technician: Chatura Nagahawatte

User: Sinnard, Jay (SINNARJD) [SINNARJ]
 Department: Multimedia
 Date Arrived: 2003-05-07

Description:
 We would like to add another hard drive to the Blue editing suite. I have the hard drive at my desk. thanks no hurry Jay Sinnard
 Operational Services Manager
 Multimedia Services
 University Libraries
 University of Cincinnati
 Phone: (513) 556-1893
 Fax: (513) 556-0991
 Pager: (513) 269-4614

Resolution:

Update Call 844

Done Internet

Figure 37. Edit Open Help Call Page.

When Update button is clicked from the “edit_ticket.php” page a request to “update_ticket.php” page is sent where it processes the selected record and outputs and update successful or failed message.

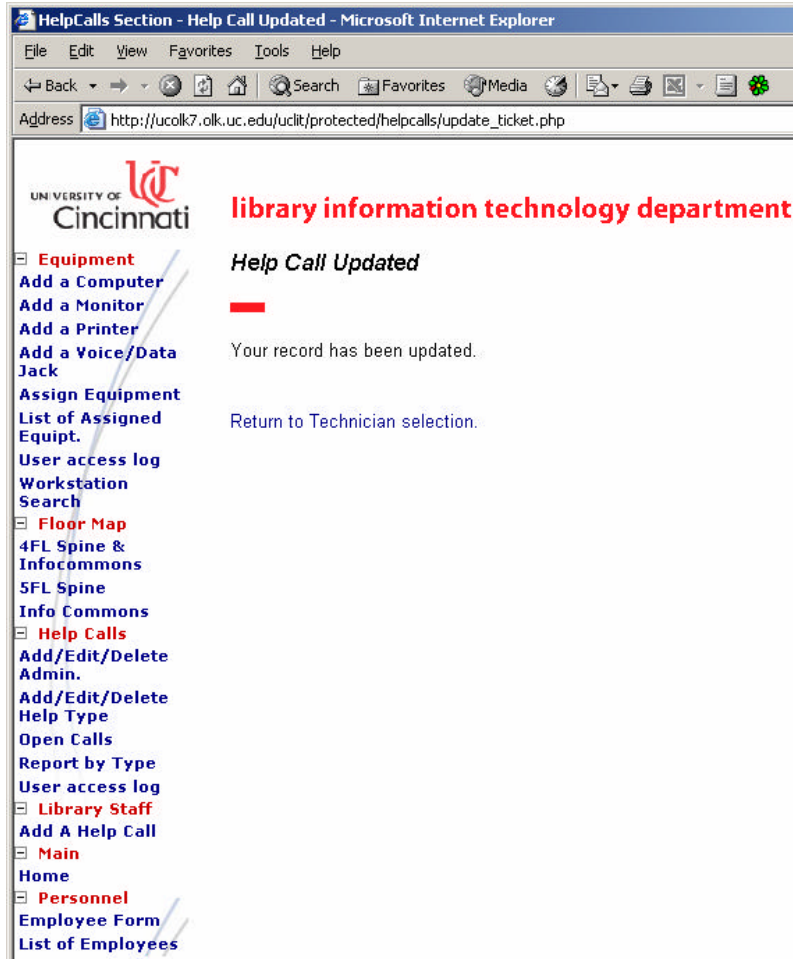


Figure 38. Help Call Updated Page.

Below the figure shows the record of a submitted help call (See Figure 10.).

2003-06-06	Info Commons	Thomas Scherz	I got the same message from "Virtual Bouncer" this morning on the pc at the Info Commons Desk next to the phone. It also appears as a program when you go to "Start". yhtak	Kinsey, Kathleen (kinseyka)	*		
2003-06-08	Library IT	Thomas Scherz	Still yet another testing message	Kevin Hsu	61745		

Open A New Call | Return to Select a Tech

Figure 39. added record shown in list Page.

The idea that a form could be reused to perform different task was thought of during the later part of this project. The “show_addhelp.php” page (See Figure 39.) shown the result of the delete button being clicked from the “sel_by_tech.php” page.

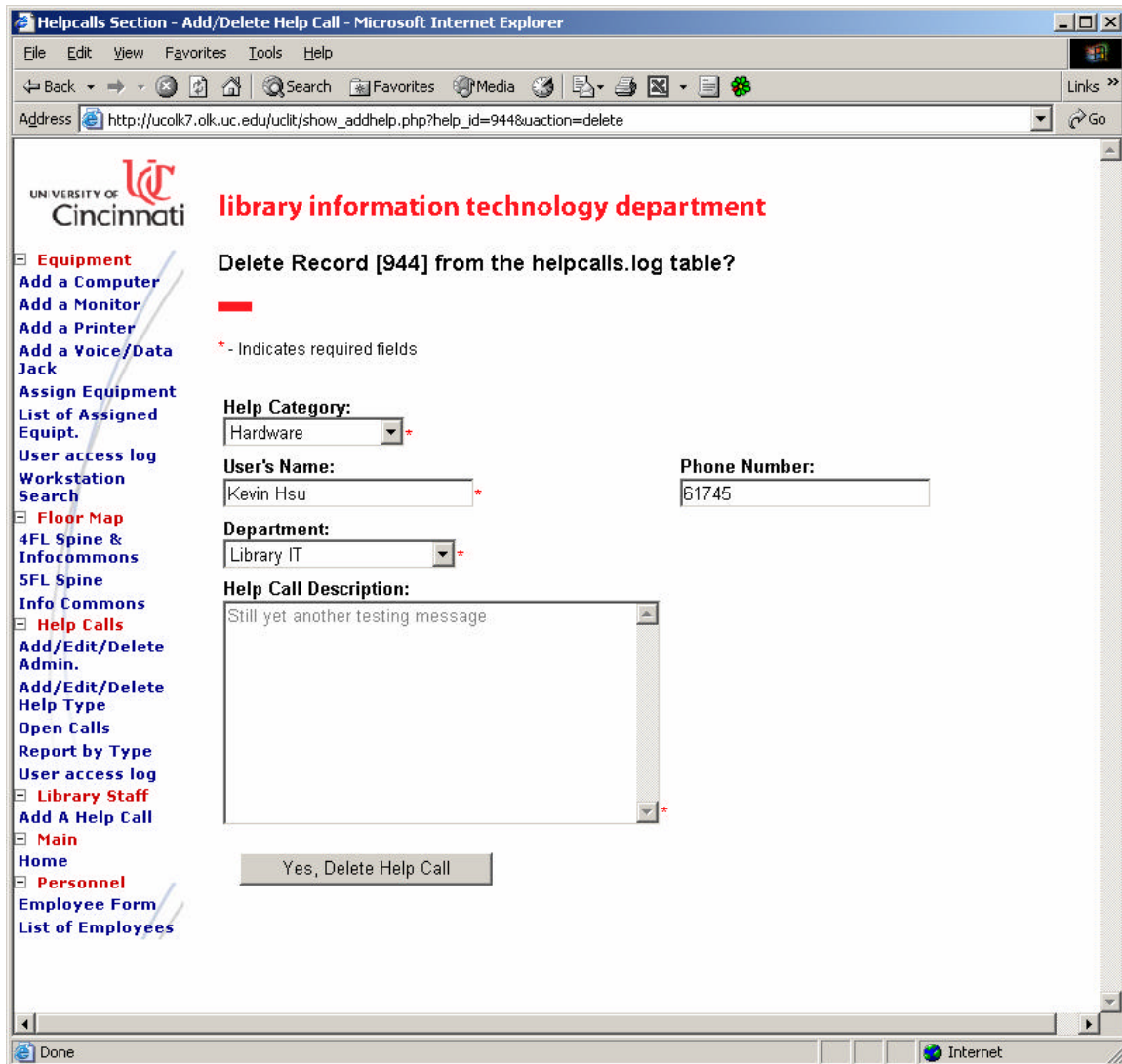


Figure 40. Open Call Delete Page.

The “Create Help Call Report” page allows user to select a beginning date and an ending date, which help type report may be generated. A JavaScript alert box will appear if the ending date is lest than the starting date. (See figure 45.) for an example of an alert box.

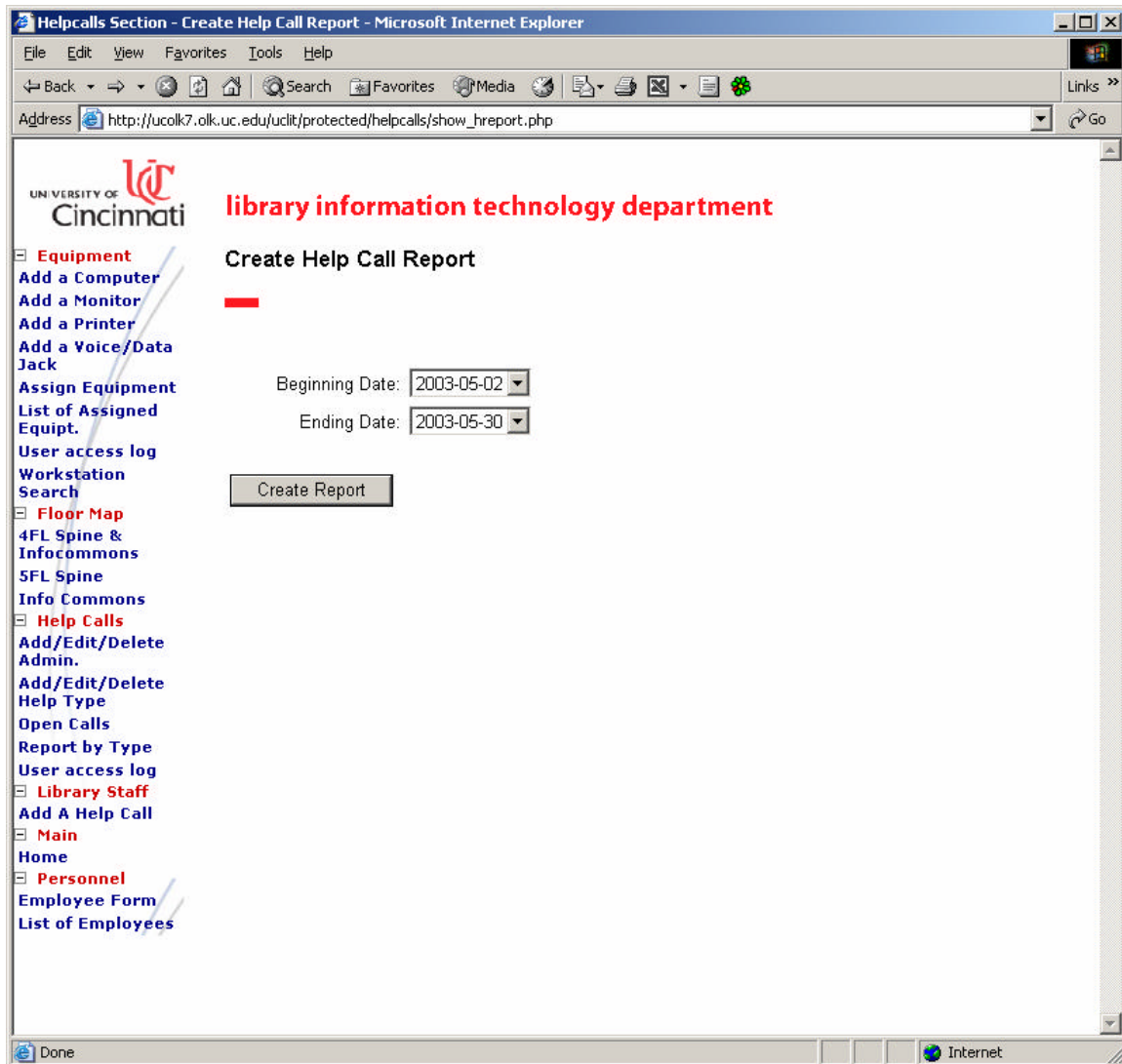


Figure 41. Create Help Call Report Page.

The “Generated Help Call Report” page uses the \$_POST action provided by PHP to get the value of the starting date and ending date then is uses that in a pre-assigned SQL statement where it counts the occurrence of each help category type from the starting date to the ending date.

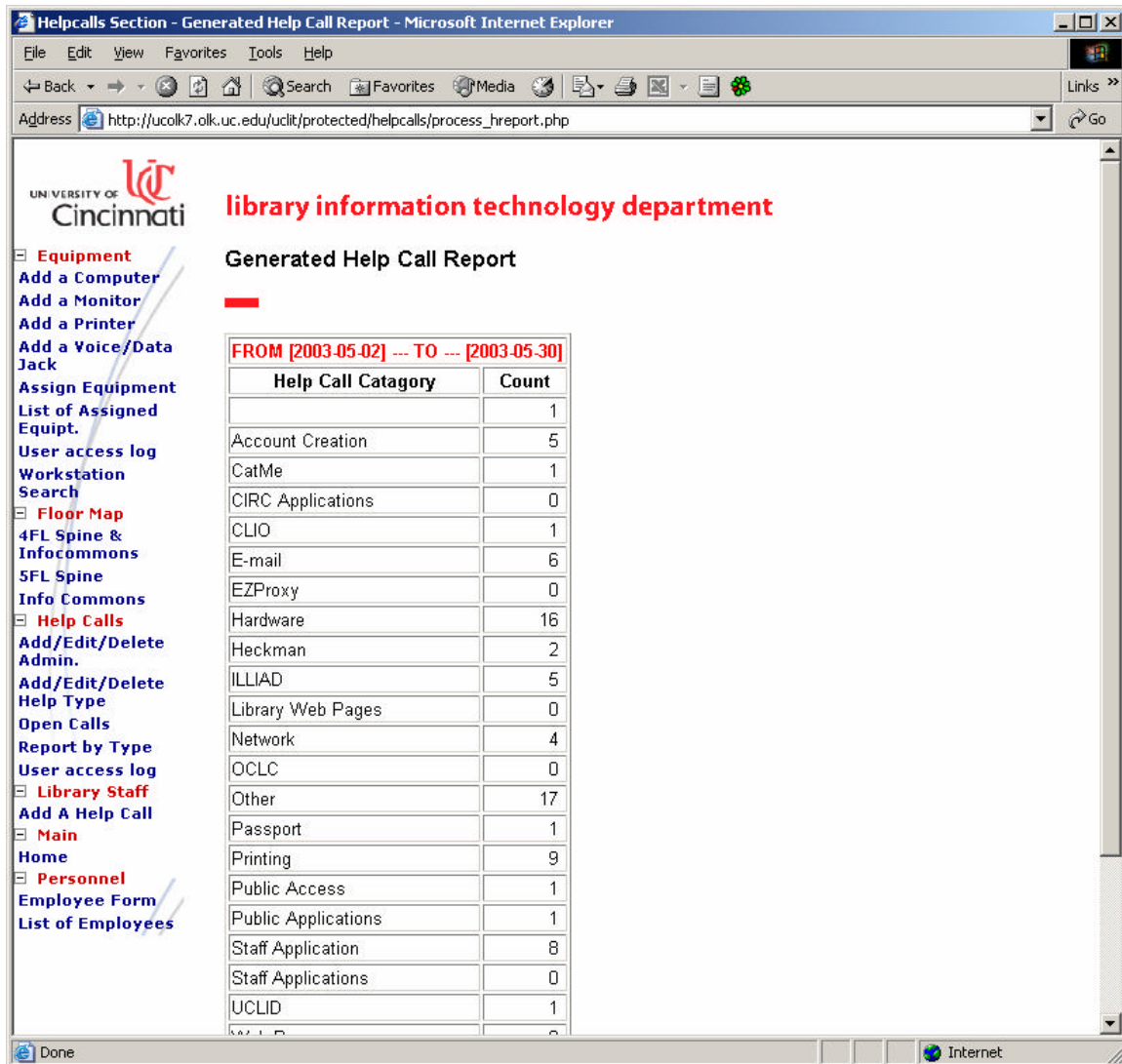


Figure 42. Generated Help Call Report Page.

The “Help Calls User Access Log” page logs add, edit, and delete from the “Help Calls” section. The “Add A Help Call” page does not required user to be logged Thus, the add action from that page was left out of the log.

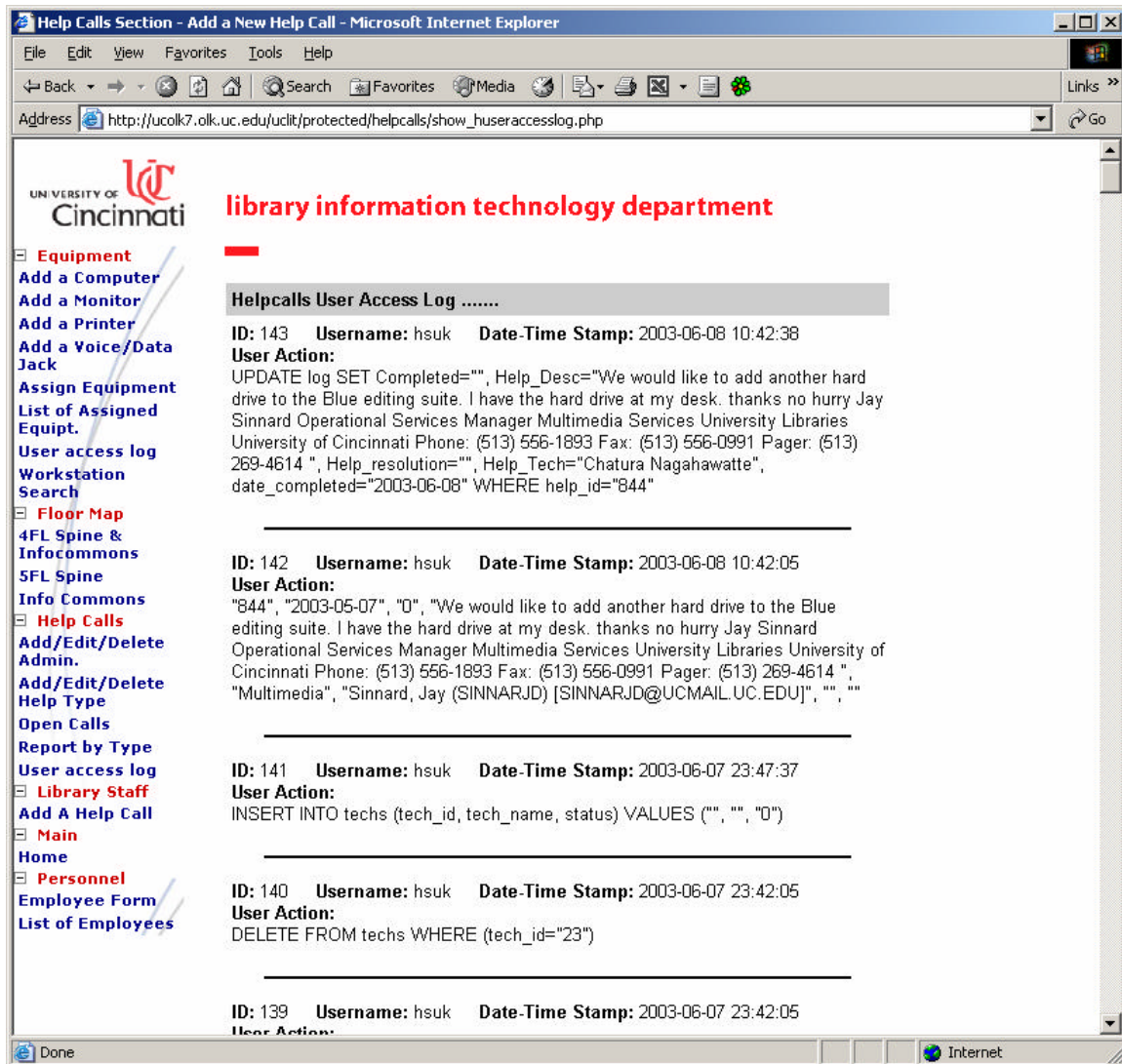


Figure 43. Help Call User Log Page.

5.7 Main Section

Main section contains a link back to the main home page, default.php (See Figure 9.).

5.8 Personnel Section

Personnel section contains two links, first a link to employee form, where user may add a new employee. The employee form also acts as an edit and delete form if an edit or delete link is clicked from the "List of Employees" page. When the list of employee page is requested, it executes a SQL query statement that pulls the data from

the personnel table in the equipment database where employees are sorted by department id.

The screenshot shows a web browser window with the address bar displaying a URL from the University of Cincinnati. The page title is 'library information technology department' and the main heading is 'Add New Employee'. A red asterisk indicates required fields.

Navigation Links (Left Sidebar):

- Equipment
 - Add a Computer
 - Add a Monitor
 - Add a Printer
 - Add a Voice/Data Jack
 - Assign Equipment
 - List of Assigned Equip.
 - User access log
 - Workstation Search
- Floor Map
 - 4FL Spine & Infocommons
 - 5FL Spine
 - Info Commons
- Help Calls
 - Add/Edit/Delete Admin.
 - Add/Edit/Delete Help Type
 - Open Calls
 - Report by Type
 - User access log
- Library Staff
 - Add A Help Call
- Main
 - Home
- Personnel
 - Employee Form
 - List of Employees

Form Fields:

- Personnel ID:
- Last Name: *
- First Name: *
- Middle Name Initial:
- Department ID: *
- Mail Location:
- E-mail:
- Telephone #:
- Fax #:
- Web Role:

Buttons:

- Add Employee
- List of All Employees

Figure 44. Add New Employee Page.

The text fields and list/menu marked with a red asterisks indicates that it is a required field and cannot be blank when the submit button is clicked. When a required field is empty when the submit button is clicked, JavaScript form validator function is executed checking to make sure the required fields are not blank. If it is an alert box will appear with the required field displayed in the alert box (See figure 45.).

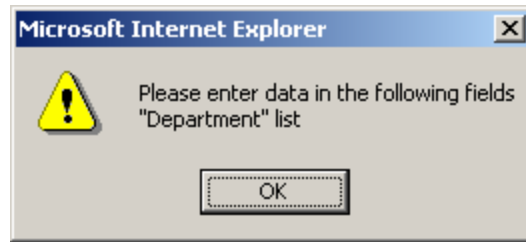


Figure 45. JavaScript Error Message Alert Box.

The “List of All The Employees” page executes an select SQL statement each time the page is requested, where the employees are sorted by department id. I ran into a problem when trying to display department names rather than displaying department id.

FID	LName	FName	MInitial	Department	Mail Location	E-mail	Phone#	Fax	Web Page	Edit	Delete
47	Good	Roger	*	0	*	roger.guod@uc.edu	95858	*		✎	✖
88	Hernandez	Stephanie	*	0	*	stephanie.hernandez@uc.edu	745 5710	*		✎	✖
72	Musata	Fred	*	0	*	fred.musata@uc.edu	733 5200	*		✎	✖
74	Musata	Dave	*	0	*	dave.musata@uc.edu	80386	*		✎	✖
113	Thomas	Virginia	*	0	*	virginia.thomas@uc.edu	83153	*		✎	✖
34	Stokius	Dore	*	1	0000	dore.stokius@uc.edu	84503	*		✎	✖
44	Corleone	Dan	*	1	0000	dan.corleone@uc.edu	84404	*		✎	✖
76	Mayfield	James	*	1	0000	james.mayfield@uc.edu	83829	*		✎	✖
77	McMinn	Mary Anne	*	1	0000	mary.mcminn@uc.edu	84502	*		✎	✖
83	Montana	Victoria	*	1	0000	victoria.montana@uc.edu	84515	*		✎	✖
96	Paul	Priscilla	*	1	0000	priscilla.paul@uc.edu	84609	*		✎	✖
98	Reynolds	Jerry	*	1	0000	jerry.reynolds@uc.edu	84504	*		✎	✖
99	Ross	Melinda	*	1	0000	melinda.ross@uc.edu	84558	*	Developer	✎	✖
96	Preckwinkle	Jelly	*	1	0000	jelly.preckwinkle@uc.edu	84509	*		✎	✖
123	Rapinier	Decky	*	1	0000	decky.rapinier@uc.edu	84551	*		✎	✖
127	Whelan	Peggie	*	1	0000	peggie.whelan@uc.edu	80866	*		✎	✖
36	Francis	Mary	*	2	0000	mary.francis@uc.edu	84441	*		✎	✖
88	Leckinger	Irene	*	2	0000	irene.leckinger@uc.edu	84441	*		✎	✖
96	Riley	Jacqueline	*	2	0000	jacqueline.riley@uc.edu	82484	*		✎	✖
26	Cordery	Melinda	*	2	0000	melinda.cordery@uc.edu	84484	*		✎	✖
42	Chickensh	Inge	*	3	0000	inge.chickensh@uc.edu	84408	*		✎	✖
95	Hutcher	Janice	*	3	0000	janice.hutcher@uc.edu	84306	*		✎	✖
110	Stokius	Judy	*	3	0000	judy.stokius@uc.edu	84702	*		✎	✖
110	Stokius	Judy	*	3	0000	judy.stokius@uc.edu	84463	*		✎	✖
129	Vilma	Tara	*	3	0000	tara.vilma@uc.edu	84408	*		✎	✖
27	Correll	Alice	*	4	0113	alice.correll@uc.edu	84353	*		✎	✖
147	Correll	Alice	*	4	0113	alice.correll@uc.edu	84353	*		✎	✖

Figure 46. List of All The Employees Page.

The “process_employee_add.php” page checks to make sure the employee does not already exist in the personnel table then it either displays the employee added (See Figure 47.) or an message that says “The employee already exist in the personnel table” in the content section of the page layout.

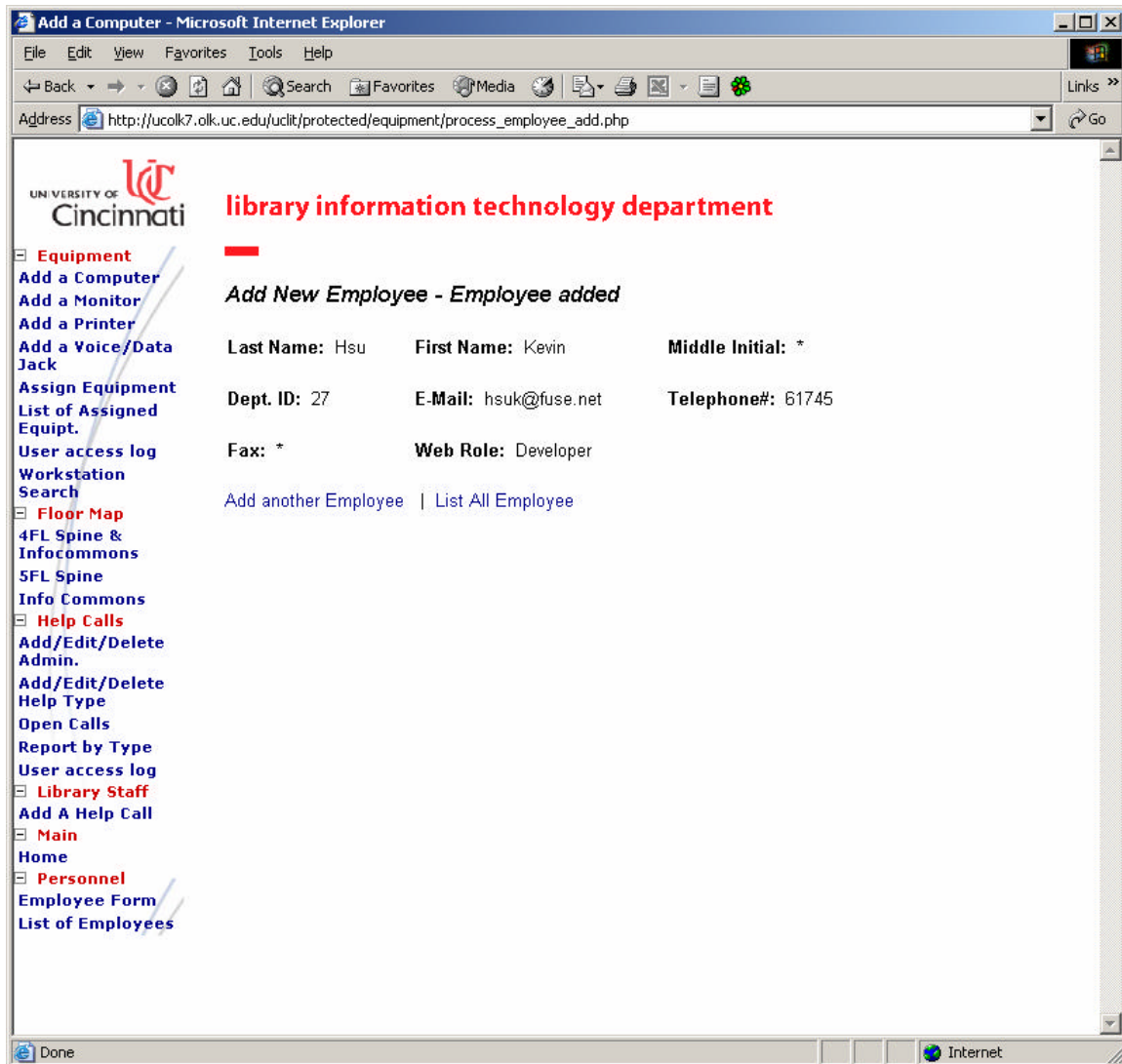


Figure 47. Employee Added Page.

A new employee has been added to the personnel table and the employee is listed in the “List of Employee” page (See Figure 48.).

37	Fritsche	Erma	*	27	0033	erma.fritsche@uc.edu	61437	*	*		
153	Hsu	Kevin	*	27	0033	hsuk@fuse.net	61745	*	Developer		
75	Matchan	Cherie	*	27	0033	cherie.matchan@uc.edu	61404	*	Developer		

Figure 48. Added Employee shown in List of All Employee Page.

The Edit and Update page of the personnel page has the “Personnel ID” disabled because the personnel id is the primary id and that if it was not disabled it may cause further database errors such as duplicated “Department ID”, or duplicated employee.

The screenshot shows a web browser window with the URL `http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_addemployee.php?pid=153&uaction=edit`. The page header includes the University of Cincinnati logo and the text "library information technology department". The main heading is "Update Record['153'] from Personnel". A red asterisk indicates required fields. The form contains the following fields:

- Personnel ID: 153 (disabled)
- Last Name: Hsu *
- First Name: Kevin *
- Middle Name Initial: *
- Department ID: Library IT *
- Mail Location: 0033
- E-mail: hsuk@fuse.net
- Telephone #: 61745
- Fax #: *
- Web Role: Developer

At the bottom of the form is an "Update Employee" button and a link "List of All Employees". The left sidebar contains a navigation menu with categories like Equipment, Floor Map, Help Calls, Library Staff, Main, Home, and Personnel.

Figure 49. Update Employee Page.

When the Update button is clicked or depressed from the “show_addemployee.php” page, the “process_employeeud.php” page will be request to perform a query on the selected record.

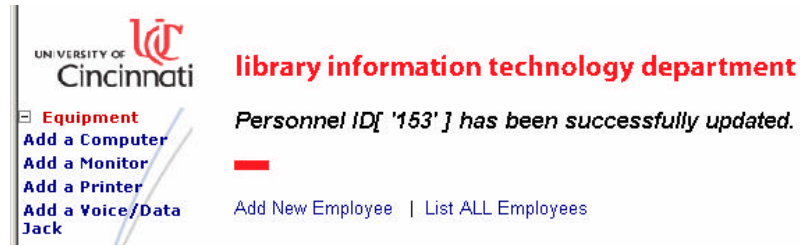


Figure 50. Updated Successful Page.

Result after a submit button with the Delete value is depressed or clicked from the “show_addemployee.php” page the “process_employeeud.php” page processes the selected record (See Figure 51.).

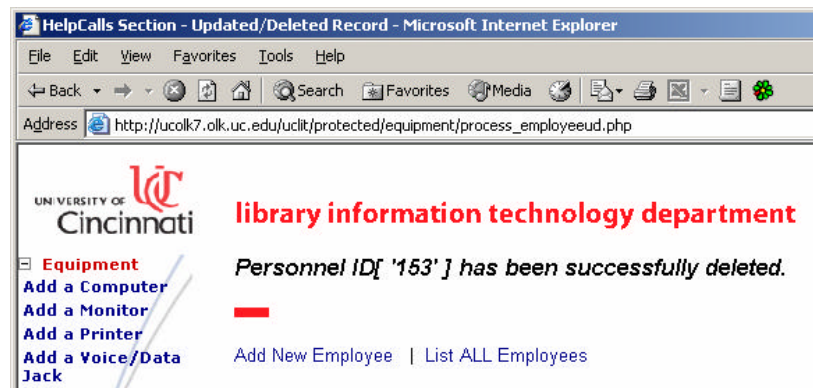


Figure 51. Employee Deleted Successful Page.

6. Testing Procedures

Testing of the functional requirements:

6.1. The User authentication is required to access the main function of the web application.

- 6.1.a. User in the .htpasswd file are able to login.
- 6.1.b. Unauthorized users are denied access.
- 6.1.c. Required text fields are not empty.
- 6.1.d. Duplication of primary key is not allowed.
- 6.1.e. When error occurs, error message is returned / displayed on document.

6.2 GUI Testing.

6.2.a. Navigation menu appears on the left side of every page

6.2.b. UC logo appears on the upper left corner of each page

6.2.c. UC logo links to www.uc.edu

6.2.d. Department title links to <http://ucolk7.olk.uc.edu/uclit/default.php>

6.2.e. A survey will also be passed out to the Library Information Technology department staff to gather user input as well as document of error occurred.

Suggestions gathered from the survey:

- ❖ Menus default to closed.
- ❖ Assigned equipment to employee should be by “workstation_id” rather than “computer_serial”.
- ❖ List all assigned equipment needs to be employee name rather than employee id. Workstation id instead of pc serial.
- ❖ “sel_by_tech.php”: post by column needs to be smaller and description column wider.
- ❖ List of all employees needs to be organized by department name not department id.
- ❖ Query for things by department. Equipment, personnel, help calls, etc.
- ❖ List of assigned equipment needs to be under equipment section not under help calls section.

7. Conclusions and recommendations

7.1 Conclusions

This project was created in response to the LIT department's need for a system that is web accessible so, that valuable time can be spent on resolving trouble calls rather than running back and forth to the office for needed information on equipment or on reported help call. I've developed, a system that is based on functionalities rather than greatly on graphical appearance, a database driven web application that allows LIT support personnel and system administrators the ability to access the equipment and help calls database via a web browser. The functionality provides the user with add, update, delete, and search capabilities in addition to viewing logs for each database access. This project was developed using Mandrake 8.2 Linux Operating System with Apache 1.3.23, MySQL 3.23.47, and PHP 4.1.2, JavaScript, Cascading Style Sheet, Macromedia Dreamweaver MX, and PremiumSoft MySQL Studio Navicat 5.0.

7.2 Recommendations

It is highly recommended that would be developers have some C/C++ or Java experience before starting, PHP is very similar. During development I ran into some problems with PHP pages not being able to be displaying on a server Apache and PHP installed. This problem was resolved by echoing from the start to the ending HTML tag. Furthermore, I discovered that PHP pages with `<? ?>` tags will not work properly and PHP coding is compromised by view source, users will be able to see server side variables.

Although, this project used some includes files for database connectivity, it was not used throughout the whole site due to lack of time and poor management and poor

project planning by the project manager. I recommend using include function to include a file with connection strings and databases name as well as tables to a database where the connection strings may be changed from a single page rather than multiple pages.

The version of MySQL does not have cascading delete and foreign key constraint so developer will have to manually enforce it. A newer version is in the work which will have those features built-in.

Appendix A.

Table Structure for Menu Database

```
#-----
# Table structure for nodes
#-----
create table nodes (
  nodeId int(11) not null auto_increment,
  title varchar(100) not null,
  url varchar(250),
  parentId int(11) not null default '0',
  primary key (nodeId),
  unique id (nodeId))
type=MyISAM;

#-----
# Records for table nodes
#-----

insert into nodes values
(1, 'Equipment', '', 0),
(2, 'Help Calls', '', 0),
(3, 'Floor Map', '', 0),
(4, 'Add a Computer',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_addcomputer.ph
p', 1),
(5, 'Add a Monitor',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_addmonitor.php
', 1),
(6, 'Add a Printer',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_addprinter.php
', 1),
(25, 'List of Assigned Equipt.',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_allequiptdetail.php', 1),
(8, 'Open Calls',
'http://ucolk7.olk.uc.edu/uclit/protected/helpcalls/show_tech.php', 2),
(9, 'User access log',
'http://ucolk7.olk.uc.edu/uclit/protected/helpcalls/show_huseraccesslog
.php', 2),
(10, 'Info Commons',
'http://ucolk7.olk.uc.edu/uclit/floor_map/InfoCommon_files/InfoCommon_g
if_1.htm', 3),
(11, '4FL Spine & Infocommons',
'http://ucolk7.olk.uc.edu/uclit/floor_map/4Spine_files/4Spine_gif_1.htm
', 3),
(13, 'Add a Voice/Data Jack',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_addpdjack.php'
, 1),
(12, '5FL Spine',
'http://ucolk7.olk.uc.edu/uclit/floor_map/5Spine_files/5Spine_gif_1.htm
', 3),
(14, 'Home', 'http://ucolk7.olk.uc.edu/uclit/default.php', 15),
(15, 'Main', '', 0),
```

```

(16, 'User access log',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_euseraccesslog
.php', 1),
(17, 'Report by Type',
'http://ucolk7.olk.uc.edu/uclit/protected/helpcalls/show_hreport.php',
2),
(18, 'Personnel', '', 0),
(19, 'Employee Form',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_addemployee.ph
p', 18),
(20, 'Workstation Search',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_pcsearch.php',
1),
(21, 'List of Employees',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_allemmployee.ph
p', 18),
(22, 'Assign Equipment',
'http://ucolk7.olk.uc.edu/uclit/protected/equipment/show_assignequipt.p
hp', 1),
(23, 'Library Staff', '', 0),
(24, 'Add A Help Call',
'http://ucolk7.olk.uc.edu/uclit/show_addhelp.php', 23),
(26, 'Add/Edit/Delete Admin.',
'http://ucolk7.olk.uc.edu/uclit/protected/helpcalls/show_addadmin.php',
2),
(28, 'Add/Edit/Delete Help Type',
'http://ucolk7.olk.uc.edu/uclit/protected/helpcalls/show_addhelptype.ph
p', 2);

```

Appendix B

Equipment Database

```
use equipment ;
#-----
# Table structure for computers

#-----
create table computers (
  workstation_id varchar(15) not null,
  computer_serial varchar(25) not null,
  computer_uc_property_num varchar(7) not null,
  computer_type varchar(30) not null,
  cpu_type_speed varchar(30) not null,
  ram varchar(7) not null,
  hd_capacity varchar(10) not null,
  video_card varchar(40) not null,
  sound_card varchar(40) default '\N/A',
  removeable_storage varchar(10) default '\N/A',
  cd_rw smallint(1) not null default '0',
  dvd smallint(1) not null default '0',
  scsi_adapter smallint(1) not null default '0',
  wand_gun smallint(1) not null default '0',
  card_reader smallint(1) not null default '0',
  additional_peripherals varchar(100) default '\N/A',
  installed_date date default '2000-00-00',
  cascade_date date default '2000-00-00',
  cascade_from_location varchar(25) default '\N/A',
  surplus_date date default '0000-00-00',
  nic_make varchar(20) default '\N/A',
  nic_model varchar(20) default '\N/A',
  network_access_speed varchar(15) default '\N/A',
  ip_address varchar(15) default '\DHCP',
  ip_gateway varchar(15) default '\N/A',
  mac_address varchar(17) not null,
  primary_server varchar(15) default '\N/A',
  operating_system varchar(30) not null,
  office_suite varchar(40) default '\N/A',
  browser_app varchar(25) default '\N/A',
  email_app varchar(25) default '\N/A',
  graphic_app varchar(40) default '\N/A',
  specialized_appz varchar(100) default '\N/A',
  catagory varchar(20) not null,
  primary key (computer_serial),
  unique computer_serial (computer_serial),
```

```
index computer_uc_property_num (computer_uc_property_num),
index workstation_id (workstation_id))
type=MyISAM comment="stores computer related data";
```

```
#-----
```

```
# Table structure for departments
```

```
#-----
```

```
create table departments (
  dept_id smallint(5) unsigned not null auto_increment,
  dept_name varchar(25) not null,
  building varchar(25) not null default '\NULL',
  room varchar(10) not null default '\NULL',
  mail_location varchar(10) not null default '\NULL',
  primary key (dept_id))
type=MyISAM;
```

```
#-----
```

```
# Table structure for equipment_detail
```

```
#-----
```

```
create table equipment_detail (
  detail_id int(11) not null auto_increment,
  personnel_id int(11) not null default '0',
  computer_serial varchar(25) not null,
  monitor_serial varchar(25),
  printer_serial varchar(25),
  jacks_id int(10),
  primary key (detail_id))
type=MyISAM;
```

```
#-----
```

```
# Table structure for EquipmentUserAccessLog
```

```
#-----
```

```
create table EquipmentUserAccessLog (
  id int(11) not null auto_increment,
  username varchar(12) not null,
  useraction text not null,
  timestamp datetime not null default '0000-00-00 00:00:00',
  primary key (id))
type=MyISAM;
```

```

#-----
# Table structure for monitor_brand
#-----
create table monitor_brand (
  `mid` int(10) unsigned not null auto_increment,
  mbrand varchar(20) not null,
  primary key (`mid`))
type=MyISAM;

#-----
# Table structure for monitors
#-----
create table monitors (
  monitor_make varchar(25) not null,
  monitor_model varchar(30) not null,
  monitor_serial varchar(30) not null,
  monitor_uc_property_num varchar(20),
  monitor_lock_key varchar(6) binary default '000000',
  monitor_size enum('14inch','15inch','17inch','19inch','20inch','21inch','22inch','24inch')
not null default '14inch',
  primary key (monitor_serial),
  index monitor_uc_property_num (monitor_uc_property_num))
type=MyISAM comment="stores monitor data";

#-----
# Table structure for personnel
#-----
create table personnel (
  pid smallint(5) unsigned not null auto_increment,
  lname varchar(25) not null,
  fname varchar(25) not null,
  mname varchar(15) default '*',
  dept_id smallint(5) unsigned not null default '0',
  mail_location varchar(10) not null default '*',
  email varchar(30) not null default 'No E-mail',
  phone varchar(21) default '*',
  fax varchar(15) default '*',
  web_role varchar(50) default '*',
  primary key (pid))
type=MyISAM comment="Stores Supported Employee Information";

```

```

#-----
# Table structure for printer_manufacturer
#-----
create table printer_manufacturer (
  id smallint(6) not null auto_increment,
  brand_printer varchar(20) not null,
  primary key (id))
type=MyISAM;

#-----
# Table structure for printers
#-----
create table printers (
  printer_serial varchar(25) binary not null,
  printer_name varchar(20) not null default 'LOCAL',
  printer_server varchar(20) default 'NONE',
  printer_make varchar(18) not null,
  printer_model varchar(30) not null,
  printer_uc_property_num varchar(15) not null default 'NONE',
  primary key (printer_serial),
  index printer_uc_property_num (printer_uc_property_num))
type=MyISAM comment="stores printer data";

#-----
# Table structure for voice_data_jacks
#-----
create table voice_data_jacks (
  jack_id int(10) unsigned not null auto_increment,
  department varchar(25) not null,
  phone_jack varchar(10) default '0',
  data_jack varchar(10) not null,
  room_jack varchar(20) default '\N/A\ ',
  primary key (jack_id),
  unique data_jack (data_jack),
  index phone_jack (phone_jack))
type=MyISAM comment="stores location, phone, data jack data";

```

Appendix C

Help Calls Database

```
use helpcalls ;
#-----
# Table structure for help_catagories
#-----
create table help_catagories (
  help_type_id smallint(5) unsigned not null auto_increment,
  help_type varchar(25) not null,
  primary key (help_type_id))
type=MyISAM;

#-----
# Table structure for HelpcallsUserAccessLog
#-----
create table HelpcallsUserAccessLog (
  id int(11) not null auto_increment,
  username varchar(50) not null,
  useraction mediumtext not null,
  datetimestamp datetime not null default '0000-00-00 00:00:00',
  primary key (id))
type=MyISAM;

#-----
# Table structure for techs
#-----
create table techs (
  tech_id smallint(6) not null auto_increment,
  tech_name varchar(25) not null,
  `status` smallint(1) unsigned not null default '1',
  primary key (tech_id),
  index tech_name (tech_name))
type=MyISAM;
```

Appendix D

1. Create a folder where you wish to password protect the files, any folder level within the /var/www/http/htdocs directory.

2. Create a .htaccess file within the protected directory and type the following

Place below in .htaccess file

```
AuthUserFile /home/joe/public_html/log/passwords/.htpasswd
AuthGroupFile /home/joe/public_html/log/passwords/.htgroup
AuthName "Secure Area, Login Required"
AuthType Basic
<Limit GET>
require valid-user
</Limit>
```

or

```
AuthName "restricted stuff"
AuthType Basic
AuthUserFile /etc/httpd/users/.users

<Limit GET POST PUT>
require user
</Limit>
```

3. Open httpd.conf for edit and add the <Directory>....</Directory> tag

Example on a Linux box:

```
<Directory "/var/www/http/uclit/protected">
AllowOverride AuthConfig
</Directory>
```

4. Create .htpasswd file to store usernames and passwords, in one of the Apache directory.

First go to console and navigate to the bin directory of Apache where the htpasswd command is stored.

Type the following to create the .htpasswd file for a user

Note the '-c' is used the first time to create the .htpasswd file and if the file exists it will be over written.

```
htpasswd -c /etc/httpd/users/.htpasswd username
```

5. Add another user
htpasswd /etc/httpd/users/.htpasswd username

References

1. Chris Lea, Wankyu Choi, Allen Kent, Ganesh Prasad, and Chris Ullman. Beginning PHP4. Birmingham [England]: Wrox Press, 2000.
2. "CSS." Meridian Training & Development.
<http://www.meridiantd.co.uk/development/pages/web/technology/css.asp>
(2, Dec. 2002)
3. David Sklar and Adam Trachtenberg. PHP Cookbook. Sebastopol: O'Reilly & Associates, Inc. 2003.
4. Fritsche, Erma. Personal Interview. 12 Aug. 2002.
5. "HTML." Jupitermedia Corporation. 2002.
<http://www.webopedia.com/TERM/H/HTML.html> (10, Nov. 2002)
6. Hugh E. Williams and David Lane. Web Database Applications with PHP & MySQL. Sebastopol: O'Reilly & Associates, Inc. 2002.
7. "JavaScript." Jupitermedia Corporation. 2002.
<http://www.webopedia.com/TERM/J/JavaScript.html> (10, Nov. 2002)
8. Julie C. Meloni. PHP Essentials. Rocklin: Prima Publishing. 2000.
9. "Macromedia MX." Macromedia, Inc. 2002.
<http://www.macromedia.com/software/dreamweaver/> (2, Dec. 2002)
10. Paul DuBois. MySQL. Indianapolis: New Riders, 1999.
11. Paul DuBois, MySQL Cookbook. Sebastopol: O'Reilly & Associates, Inc. 2003.
12. Rasmus Lerdorf and Kevin Totroe. Programming PHP. Sebastopol: O'Reilly & Associates. Inc. 2002.
13. Scherz, Thomas. Personal Interview. 12 Aug. 2002.
14. Shoryer-Matchan, Cherie. Personal Interview. 12 Aug. 2002.