

**Asset Management Application for
Hamilton County Probate Court**

By

Jonathan Ring

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

University of Cincinnati
College of Applied Science

May 2008

**Asset Management Application for
Hamilton County Probate Court**

By

Jonathan Ring

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

© Copyright 2008 Jonathan Ring

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

Jonathan Ring

Date

Hazem Said, Ph.D., Faculty Advisor

Date

Hazem Said, Ph.D. Department Head

Date

Acknowledgements

I would like to give special thanks to the people at the Hamilton County Probate Court who made this project possible, especially, Judge James Cissell, Christian Moeller (IT Director), Mark Perry (Assistant Court Administrator), and Scott Weikel (Purchasing Director), for their help and encouragement. I would also like to thank Dr. Hazem Said, IT Department Head and my Academic Advisor, for all of his suggestions and advice. Finally, I would like to thank my wife Jessica and my two children, Zoe and Jacob, for their constant emotional support, patience, and understanding.

Table of Contents

Acknowledgements.....	i
Table of Contents.....	ii
List of Figures.....	iv
Abstract.....	v
1 Statement of Problem.....	1
2 Description of the Solution.....	2
2.1 User Profile.....	2
2.2 Department Supervisors.....	3
2.3 Purchasing Director.....	3
2.4 System Administrator(s).....	3
2.5 Design Protocols.....	4
2.5.1 Organizational Structure and Navigation.....	4
2.5.2 Database Design.....	6
2.5.3 The Login Page.....	8
2.5.4 The Dashboard.....	9
2.5.5 Department Supervisor Functions.....	9
2.5.6 Purchasing Director Functions.....	10
2.5.7 System Administrator Functions.....	10
2.5.8 Interface Design.....	10
2.5.9 Style Elements and Graphics.....	12
3 Objectives of the Project (“Deliverables”).....	12
4 Development.....	15
4.1 Timeline (tasks and schedule).....	15
4.1.1 Senior Design I.....	16
4.1.2 Senior Design II.....	16
4.1.3 Senior Design III.....	17
4.2 Budget.....	17
4.3 Software.....	18
4.4 Hardware.....	19
5 Proof of Design.....	20
5.1 User Login and Authentication.....	20
5.2 The Dashboard.....	21
5.3 Adding a New Inventory Asset.....	22
5.4 The Inventory Browser.....	23
5.5 Pending Transactions.....	25
5.6 Reporting.....	27
5.7 System Administration.....	28
5.7.1 User Management.....	28
5.7.2 Role Management.....	30
5.7.3 Resource Management.....	32
6 Testing Plan and Procedures.....	32
7 Conclusions and Recommendations.....	33
Appendix A.....	34

Appendix B	35
Bibliography	36

List of Figures

Figure 1 – Use case diagram.....	5
Figure 2 – Hierarchical site structure.....	6
Figure 3 – Database diagram	7
Figure 4 – ASP .NET login controls.....	8
Figure 5 – Web site style theme.....	11
Figure 6 – Web site color scheme.....	12
Figure 7 – Deliverables.....	14
Figure 8 – Gantt chart	15
Figure 9 – Budget.....	18
Figure 10 – Login page.....	21
Figure 11 – Dashboard page	22
Figure 12 – Add asset page.....	23
Figure 13 – Inventory browser page	25
Figure 14 – Pending transactions page	26
Figure 15 - Print inventory report page.....	27
Figure 16 – User management page.....	29
Figure 17 – Role management page.....	31
Figure 18 – Test Plan (1)	33
Figure 19 – Code sample from the Inventory Services Class.....	34
Figure 20 – Testing procedure for the addAsset.aspx Web page	35

Abstract

The *Asset Management Application for Hamilton County Probate Court* provides the ability to add, modify, transfer, and dispose of court assets by authenticated users. The Hamilton County Probate Court desperately needed an application to manage its fixed assets. Previously, the purchasing director used a Microsoft Excel spreadsheet. The *Asset Management Application for Hamilton County Probate Court* will be more efficient and easier to operate. It will improve data integrity and security. This application was developed using Visual Basic .NET, ASP .NET 2.0, Ajax, and a MySQL DBMS backend. As a result, it will be more scalable for future court intranet applications. Because this application is Web-based and developed to run in the Probate Court's secure intranet environment, it does not require local installation on individual user's workstations. This application was developed using the court's current Web site design for greater conformity and user familiarity. Likewise, because of the .NET technologies used and the similarity to the Web site design, this application will enable easier maintainability for the court's current IT staff. With this application, Department Supervisors will create transactions to add, modify, transfer, and dispose of court assets. The Purchasing Director will then approve or deny these transactions upon receiving E-mail notification. Finally, the Purchasing Director will be able to generate inventory reports for the Hamilton County Auditor's annual accounting and for asset submissions to the Hamilton County auction Web site.

Asset Management Application for Hamilton County Probate Court

1 Statement of Problem

The Hamilton County Probate Court is a separate division of the Hamilton County Court of Common Pleas. The court is responsible for the administration of particular civil cases regarding Ohio Probate Law (e.g., estates, guardianships, trusts, adoptions, civil commitment of the mentally ill, and issuing marriage licenses). The court currently maintains two office locations: downtown Cincinnati and Forest Park, Ohio.

The court was in need of a program to manage the court's fixed assets. For the court's purposes, fixed assets are tangible, operational assets (i.e., equipment, furniture, computers, software, and vehicles) that are not used up within a year of purchase (4, p. 11). The court's fixed assets are distributed to both office locations and at the Hamilton County Clerk of Courts Records Center on Winton Road. The Purchasing Director is responsible for maintaining an accurate record of the court's many different fixed assets. He had been using a Microsoft Excel spreadsheet to record and account for inventory assets (5). Once a year, an accounting of these assets must be submitted to the Hamilton County Commissioners and the Hamilton County Auditor for submission to the Ohio Department of Administrative Services (3, O.R.C. 125.16).

The Hamilton County Probate Court was without a proper system to manage its fixed assets. The purchasing director was using a Microsoft Excel spreadsheet. This spreadsheet system had both real and potential problems. Updating and maintaining the Excel spreadsheet was time consuming. The spreadsheet had grown over time and took an inordinate amount of time to navigate. Moreover, the spreadsheet method was limited

in scope. Because of the fact that the document was just a flat file, there was no real way to integrate it into the court's IT infrastructure. Finally, the spreadsheet method, without data validation, was prone to errors and data corruption. There was no way to verify that the data being entered was valid for that specific field (i.e., alpha-only, alpha-numeric, etc.).

2 Description of the Solution

The solution focused on three areas of information technology: software development, Web technologies, and database management. The proposal included the development of an in-house application using software and hardware technology with which the staff was already familiar for better usability, maintainability, integration and scalability. The court had other in-house applications developed in Visual Basic using a MySQL database management system. The development of this particular project using ASP .NET, ADO .NET, Visual Basic .NET, and Ajax allowed the court's IT staff to maintain software familiarity while making the solution more scalable and easier to integrate into a future intranet.

2.1 User Profile

Three distinct user profiles are intended for this application. These profiles are referred to as "roles" and are as follows: Department Supervisors, Purchasing Director, and System Administrator(s). All inventory transactions created by the Department Supervisors will be subject to the approval of the Purchasing Director. The Purchasing Director will have access to additional features of the system (e.g., generating inventory reports, submitting assets to county administration for auction). Finally, a system

administrator (a member of the IT Department) will have access to all of the features available to the Department Supervisors and the Purchasing Director. Moreover, the System Administrator will have access to other features used for managing the inventory system, such as managing roles, users, and vendors.

2.2 Department Supervisors

In the proposed system, Departmental Supervisors will have the following responsibilities:

- Add New Asset
- Transfer Asset
- Dispose of Asset

2.3 Purchasing Director

In addition to the responsibilities of the Department Supervisors, the Purchasing Director will be responsible for:

- (Approving/Denying) Pending Transactions
- Assigning/Uploading Pictures
- Assigning/Generating/Printing Barcodes
- Sending Assets to Auction
- Generating Inventory Reports
- Searching for Inventory Assets

2.4 System Administrator(s)

The System Administrator will be responsible for the following:

- Managing Roles
- Managing Users
- Managing Department Codes
- Managing Category Codes

2.5 Design Protocols

The use of software development, Web development, and database management technology is immediately evident from the design of the application. It has a traditional Web site organizational structure that utilizes the separation of the three system roles. The Asset Management Application is a Web browser-based application with a MySQL database backend. The User Interface incorporates the latest technology available in the .NET 2.0 framework and the new Ajax code library for .NET. The site style and graphics are consistent with the Probate Court's Internet Web site.

2.5.1 Organizational Structure and Navigation

The Asset Management Application takes advantage of the Microsoft .Net 2.0 framework. Using ASP .Net, the application is browser based and only available through the court's internal network. The Asset Management Application has a hierarchical structure that separates pages and functions based on the three system roles: Department Supervisors, Purchasing Director(s), and System Administrators. This is best demonstrated by viewing the Use Case Diagram (See Figure 1, below).



Figure 1 – Use case diagram

All of the various use cases are worked into the overall structure of the Web site. This hierarchical structure begins with the Login page. After authentication, all navigation for the Asset Management Application flows from a Dashboard page that serves as the main access point to the entire site. From the Dashboard, users will navigate to other pages based on the action they will need to perform and the role(s) they are assigned. Figure 2 illustrates the overall structure of the application.

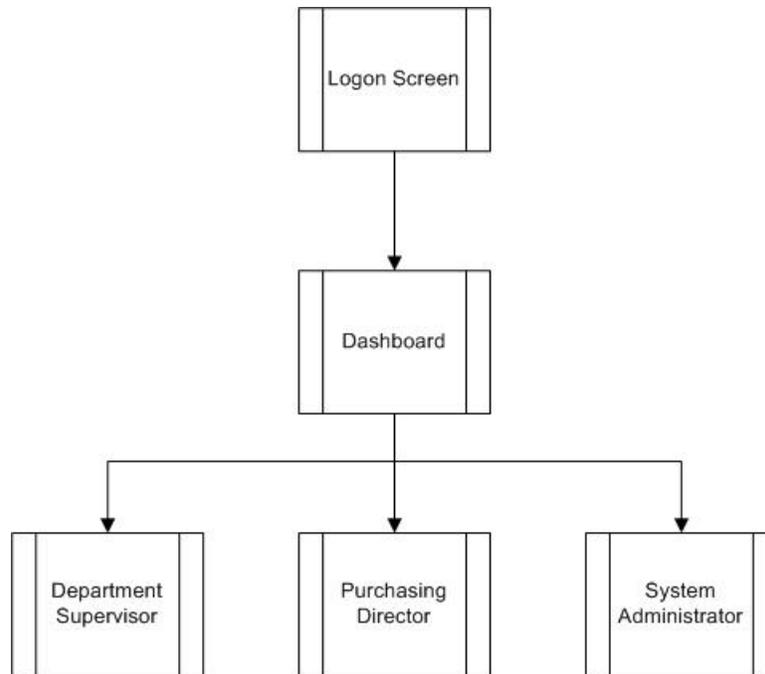
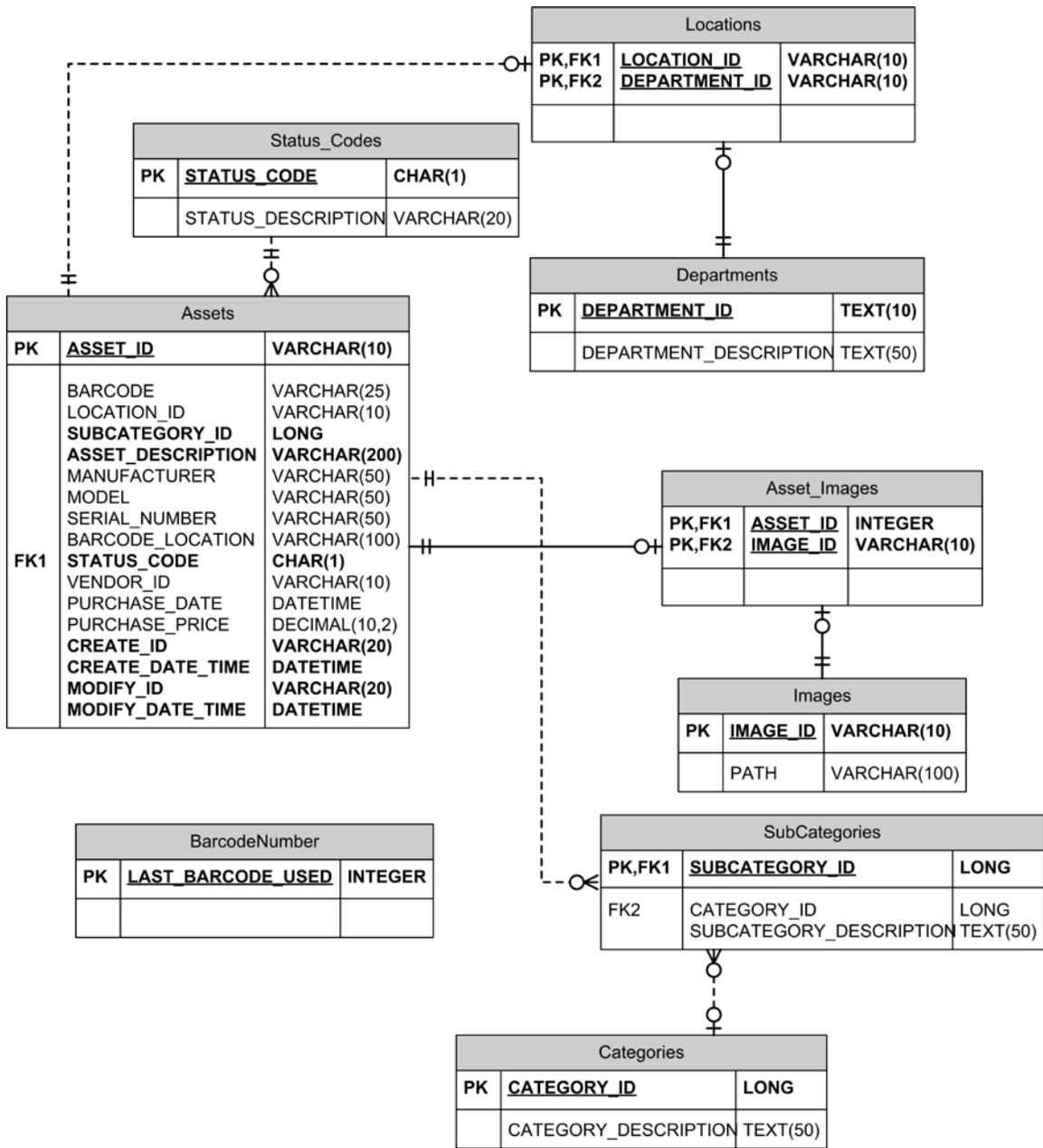


Figure 2 – Hierarchical site structure

2.5.2 Database Design

In order to ensure better data integrity and scalability, a proper database management system was necessary. MySQL Server 5.0 was selected for this project because it is currently being used as the backend for the court’s current case management system. This compliments the IT staff’s current skill set and contributes to consistency with the court’s overall IT infrastructure. Database design followed the development of the Use Case diagramming. Tables, attributes, and relationships were selected in based on the defined requirements of the system. At the heart of the inventory database is the assets table. In addition, there are other tables that support the various data attributes related to inventory assets (e.g., the categories table and the departments table). Figure 3 illustrates the various database tables, their attributes, and their relationships.



Rev. 2.0

Figure 3 – Database diagram

2.5.3 The Login Page

At first, all users are presented with a simple login page. This page validates users using the ASP .NET built-in security methods (See Figure 4).

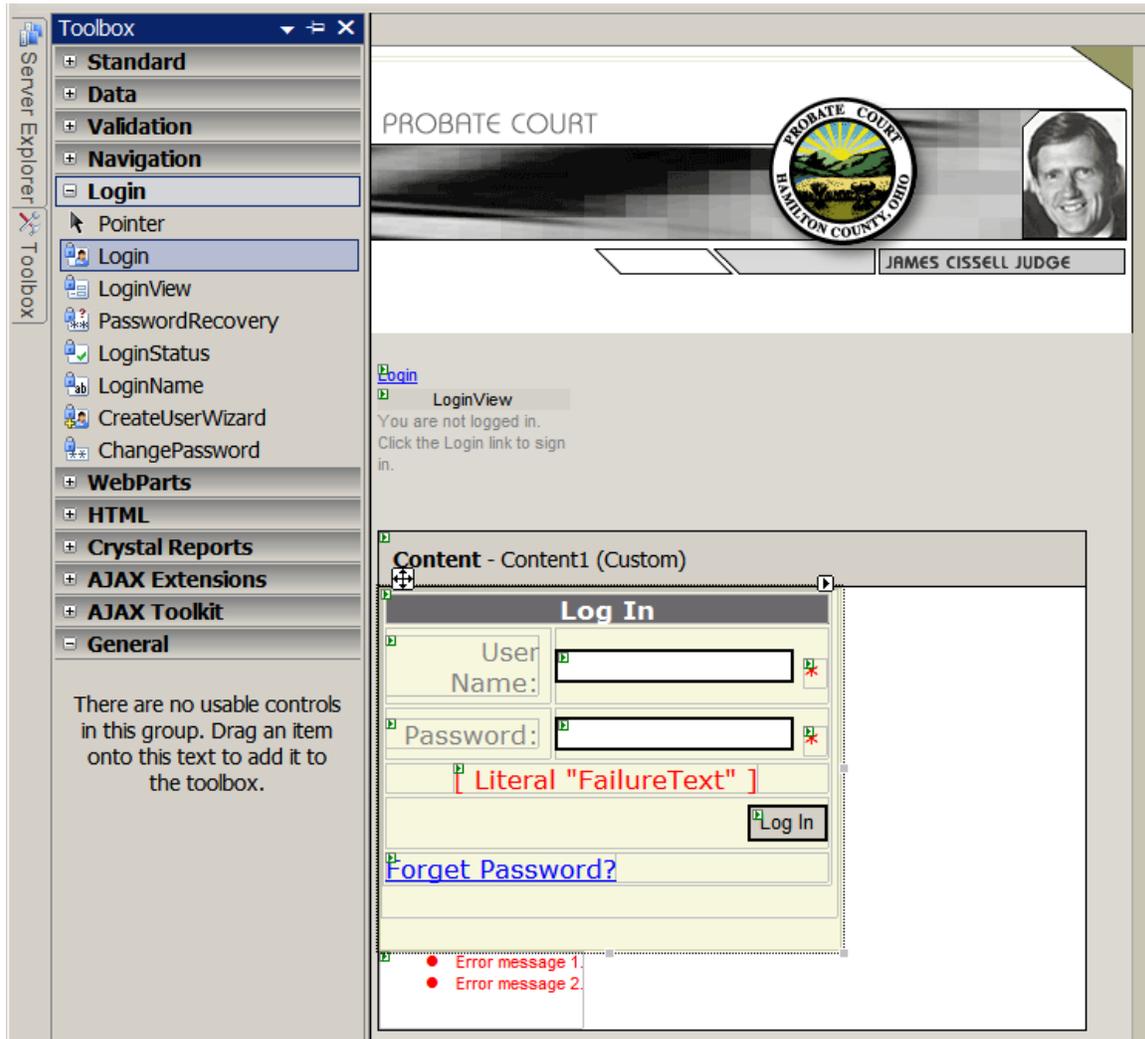


Figure 4 – ASP .NET login controls

If validated, the user then proceeds to the dashboard page. If the user name or password cannot be validated, an error message is displayed. If the user selects the “Forgot Password?” link, he or she is redirected to a forgotten password page. This “forgotten password” page allows the user to receive a new password via E-mail.

2.5.4 The Dashboard

After successfully logging into the application, all users are presented with a dashboard page. Only functions that are available to the user's role are displayed on this page. These functions are accessible to the user via hyperlink. This Dashboard page serves as the main page for the entire application. After completion of various functions, users are returned to this dashboard page. If additional features are added to the application, navigation to these new features will begin with this dashboard page as well.

2.5.5 Department Supervisor Functions

Because department supervisors have the role with the least permissions, functions available to them are accessible to all users of the system. Most functions assigned to the department supervisor are available via the Inventory Browser page. This browser displays inventory items based on filters such as departments, locations (or room numbers), and categories.

From the inventory browser, users are able to update the various attributes of individual assets. The department supervisors can mark assets for transfer and/or disposal. These actions then auto-generate E-mail to the Purchasing Director, informing him of the proposed change. It is then be up to the Purchasing Director to approve or deny the proposed action of these items.

2.5.6 Purchasing Director Functions

In addition to having access to all of the Department Supervisor functions, the Purchasing Director has additional functions specific to his/her role. Most importantly, the Purchasing Director is responsible for approving the pending transactions (newly added assets, transfers, disposals) generated by the Department Supervisors. However, any transaction generated by the Purchasing Director does not require additional approval. For example, if the Purchasing Director enters in new inventory assets when a shipment arrives, these items will post directly without the need of additional intervention.

2.5.7 System Administrator Functions

The System Administrator role has access to all the functions of the Department Supervisors and the Purchasing Director. In addition, System Administrators are able to add users, roles, and resources. Several pages are dedicated to the functions of the System Administrator, such as User Management, Role Management, and Resource Management page.

2.5.8 Interface Design

The design of the User Interface takes advantage in the rich features and code libraries available in the .NET 2.0 framework. It also uses the new Ajax code library for .NET. By using collapsible panels and update panels, both Ajax controls, greatly reduced the amount of Web pages needed. Now one page can include multiple Web forms. These Web forms are used to capture user input for all pages within the site design. User input is captured using a combination of input boxes, drop down lists, check boxes, and

list boxes. Standard buttons are used for submission of data contained in Web forms. However, for optional functions and for the submission of individual items listed in data grids, link buttons are used. These link buttons appear like hyperlinks when rendered. The appearances, or styles, of all user input controls are managed either by the psmain.cs cascading style sheet or with the HCPCSkin.skin file contained within the HCPCTheme folder in the project directory (See Figure 5).

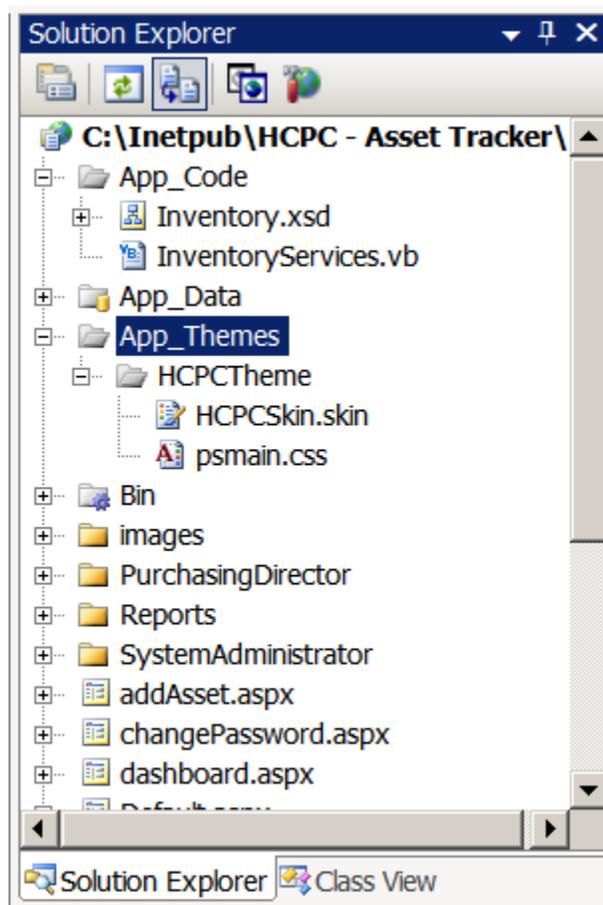


Figure 5 – Web site style theme

2.5.9 Style Elements and Graphics

In order to fit the court's current design scheme, all icons, graphics, fonts, and style elements are in conformity with the court's public Web site (<http://www.probatect.org>). Aside from pictures of the actual inventory items, all other images have been kept to a minimum. The court's standard banner and left-hand navigation images are the obvious exceptions. The overall style of the Web site is maintained through the use of an ASP .NET Master page, through which all other pages are derived, and an ASP .NET Theme. This theme contains both a single cascading style sheet and a Skin file that is used for ASP specific Web form controls. Again, the color scheme used mirrors that of the court's public Web site (See Figure 6). The result is a rich and user-friendly experience.



Figure 6 – Web site color scheme

3 Objectives of the Project (“Deliverables”)

All functions and deliverables ultimately belong to one of six major functions. These functions are: add an asset, modify an asset, transfer an asset, dispose of an asset, print reports, authenticate users, and manage users, roles, and resources. Because the project is a Web browser application, all navigation is conducted via hyperlinks and the browser's navigation buttons.

The following table (See Figure 7) lists all of the agreed-upon deliverables for the asset management application. The Purchasing Director, the IT Director, the project advisor, and the author agreed upon these deliverables. In the future, the court hopes to look at creating some value-added features to the system. These features may include aesthetic improvements to the user interface or methods to increase program efficiency.

Task	Description	Users	Deliverable
Create Use Case Scenarios		Developer Project Manager	Use Case Diagram
Database Design		Developer Project Manager	Database Diagram
Create Classes		Developer Project Manager	Class Diagram
Draft User Interface Design		Developer Project Manager	User Interface Diagrams
Create Timeline		Developer Project Manager	Timeline
Login	Authenticate users based on user ID and password	All system users	ASP .NET Web form
Forgot Password	Offers to E-mail users with new password	All system users	ASP .NET Web form
Dashboard	Page with links to functions based on user's role	All system users	ASP .NET Web form
Add New Asset	Inserts new asset into inventory database; E-mails purchasing director for approval	All system users	Asset Management Class
Transfer Asset	Updates the location (department) of the asset; E-mails purchasing director for approval	All system users	Asset Management Class
Dispose of Asset	Updates the asset and marks it for disposal; E-mails the purchasing director for approval	All system users	Asset Management Class
E-mail notification	Sends E-mail to the designated user	All system users	Services class
Pending Transactions	Allows the purchasing director to approve or reject new assets, transfer of assets, and/or disposal of	Purchasing Director	ASP .NET Web form

Task	Description	Users	Deliverable
	assets		
Assign Picture	Update an asset to assign an image	Purchasing Director	ASP .NET Web form
Upload Picture	Insert a new image to the database	Purchasing Director	ASP .NET Web form
Generate Barcode	Generate a new unique barcode for later assignment	Purchasing Director	ASP .NET Web form
Print Barcode	Print the bar code label to affix to an asset	Purchasing Director	Services Class
Assign Barcode	Update an asset to assign a bar code	Purchasing Director	ASP .NET Web form
Send To Auction	E-mail data of assets to be auctioned to County Administration	Purchasing Director	Services class
Inventory Reports	Generate reports of inventory assets	Purchasing Director	ASP .NET Web form; .NET Report
Inventory Search	Browse and search for inventory assets	Purchasing Director	ASP .NET Web form
Manage Roles	Insert/Update/Delete system roles	System Administrator	ASP .NET Web form
Manage Users	Insert/Update/Delete system users	System Administrator	ASP .NET Web form
Manage Departments	Insert/Update/Delete Probate Court Departments and Locations	System Administrator	ASP .NET Web form
Manage Categories	Insert/Update/Delete Inventory categories and subcategories	System Administrator	ASP .NET Web form

Figure 7 – Deliverables

4 Development

4.1 Timeline (tasks and schedule)

Because of the Probate Court’s desire for an Asset Management application by the summer of 2008, this proposed project’s deadline fittingly corresponded to the spring 2008 deadline for the Senior Design Project (See Figure 8).

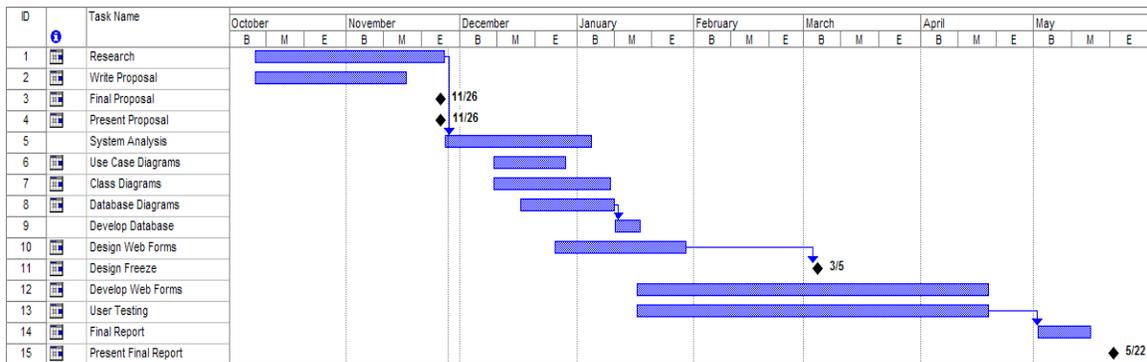


Figure 8 – Gantt chart

The design phase contained basic layouts, prototypes, and diagrams. An iterative process of design and development, based on the Unified Process, was employed for this project (2, sec. 2.2). As major development milestones were completed, some basic unit testing was conducted. Final acceptance testing began in May 2008 after development was finished. The application was implemented and presented at the end of Spring Quarter 2008.

4.1.1 Senior Design I

During the first phase of Senior Design, conducted from September through December of 2007, a great deal of research was conducted to define the exact needs and requirements of the Hamilton County Probate Court. Time spent interviewing staff and reviewing county and court documentation. Time was also devoted to determining the technology to be used and the feasibility of implementing a proposed solution. The following list shows the major milestones achieved during Senior Design I:

- Researched the specific inventory needs of the court.
- Defined the project requirements.
- Identified the system users.
- Studied the feasibility of developing a Windows-based vs. a Web-based solution.
- Developed a written proposal.
- Presented the proposal to students and faculty.

4.1.2 Senior Design II

From January until March of 2008, the second phase of the Senior Design project focused on design and development. With help and guidance from Dr. Hazem Said, a weekly agenda was created to break up larger goals into smaller milestones. This phase saw the refinement of the design of the overall system and the final development of a working prototype. These are the major milestones achieved during Senior Design II:

- Developed a weekly agenda/timeline.
- Developed System Diagrams:
 - Use Case Diagrams
 - Class Diagrams
 - Database Diagrams
 - User Interface Diagrams
- Designed the Site Hierarchy.
- Designed and developed the backend inventory database.
- Developed the ADO .NET Data Access Layer.
- Began developing the needed classes and Web forms.
- Developed the security scheme and authentication methods.
- Developed methods for printing barcodes and loading asset images.

- Began designing reports.
- Developed design freeze.
- Presented the design freeze to students and faculty.

4.1.3 Senior Design III

Finally, the third and final phase of Senior Design took place from March through June of 2008. Much of this phase, however, was completed by the middle of May 2008. This stage was a finalization of application development and the construction of detailed testing procedures. Both Unit and Acceptance testing was performed, along with a plan for implementing the completed application. The major milestones achieved during Senior Design III are listed here:

- Finished developing inventory reports.
- Finished developing Web forms.
- Developed detailed testing plans.
- Began Unit Testing.
- Began Acceptance Testing.
- Developed plan for implementation.
- Developed final report.
- Presented final report.

4.2 Budget

Fortunately, the budget has been able to remain as initially proposed. Other than the cost of normal salary, the court has only needed to expend the \$463.90 for a bar code reader (See Figure 9). The court had previously paid all other hardware and software costs in the completion of other projects or for normal business expenses.

Hardware / Software	Description	Cost
Visual Studio 2005	Integrated development environment	\$0.00
MySQL 5.0	Open source database management system	\$0.00
Windows Server 2003	Windows Server operating system	\$0.00
Internet Information Services 6.0 (IIS)	Web server implementation (included in Windows Server 2003)	\$0.00
Database Server	HP ProLiant	\$0.00
Application/Web Server	HP ProLiant	\$0.00
Digital Camera	Sony Cyber-Shot	\$0.00
Label Printer	Seiko Smart Label Printer	\$0.00
LaserJet Printer	HP Color LaserJet Printer	\$0.00
Bar Code Reader	USB compatible	\$463.90
	Total Cost:	\$463.90

Figure 9 – Budget

4.3 Software

For greater consistency, maintainability, and integration, this project was developed using the following software technologies:

- ASP .NET
- ADO .NET
- Visual Basic .NET
- Ajax
- MySQL

A critical piece of software required was Visual Studio 2005. Fortunately, the court already possessed an available license. Likewise, for the database backend, MySQL 5.0 was necessary. Again, the court already had MySQL 5.0 running on a database server.

Finally, in order for the finished project to run in a Web browser environment, IIS 6.0 (or later) was needed and was already available. However, it needed to be configured for this particular project in an intranet environment.

4.4 Hardware

The following is a list of the hardware that was necessary to develop and implement the Asset Management application:

- Web Server
- Database Server
- At minimum, 2 Client Workstations (1 for development, 1 for implementation)
- Digital Camera
- Bar Code Reader
- Label Printer
- LaserJet Printer (for printed reports)

In order to host the proposed project as an internal Web site using IIS 6.0, the court needed a server. There was a server available that met this need. Likewise, the current database server was utilized for the database backend. The court had the necessary workstations available for development, testing, and implementation.

The intended incorporation of photographs of the actual fixed assets also required the use of a digital camera. The digital cameras in use by the guardianship investigators fulfilled this requirement. The Purchasing Director already had a label printer locally attached to his workstation (via USB) that satisfied the proposed project requirement. Finally, the Purchasing Director also had an HP LaserJet printer in his office that was utilized for printing inventory reports.

5 Proof of Design

The following demonstrates the fulfillment of project deliverables and address the challenges that arose through design and development.

5.1 User Login and Authentication

All users of the system are initially met with a User Login page. User login controls illustrates just one of the many benefits of using ASP .NET 2.0. ASP .NET login controls were used in conjunction with the built-in security features for the authentication of users. A successful login redirects the users to the Dashboard page. An unsuccessful login prompts the user with an error message. The Login page also presents the user with a “Forgot Password?” link in the event the user cannot remember his/her password (See Figure 10). In this case, the user is redirected to the Forgot Password page, where he/she provides a user name, and a new password is assigned via E-mail.

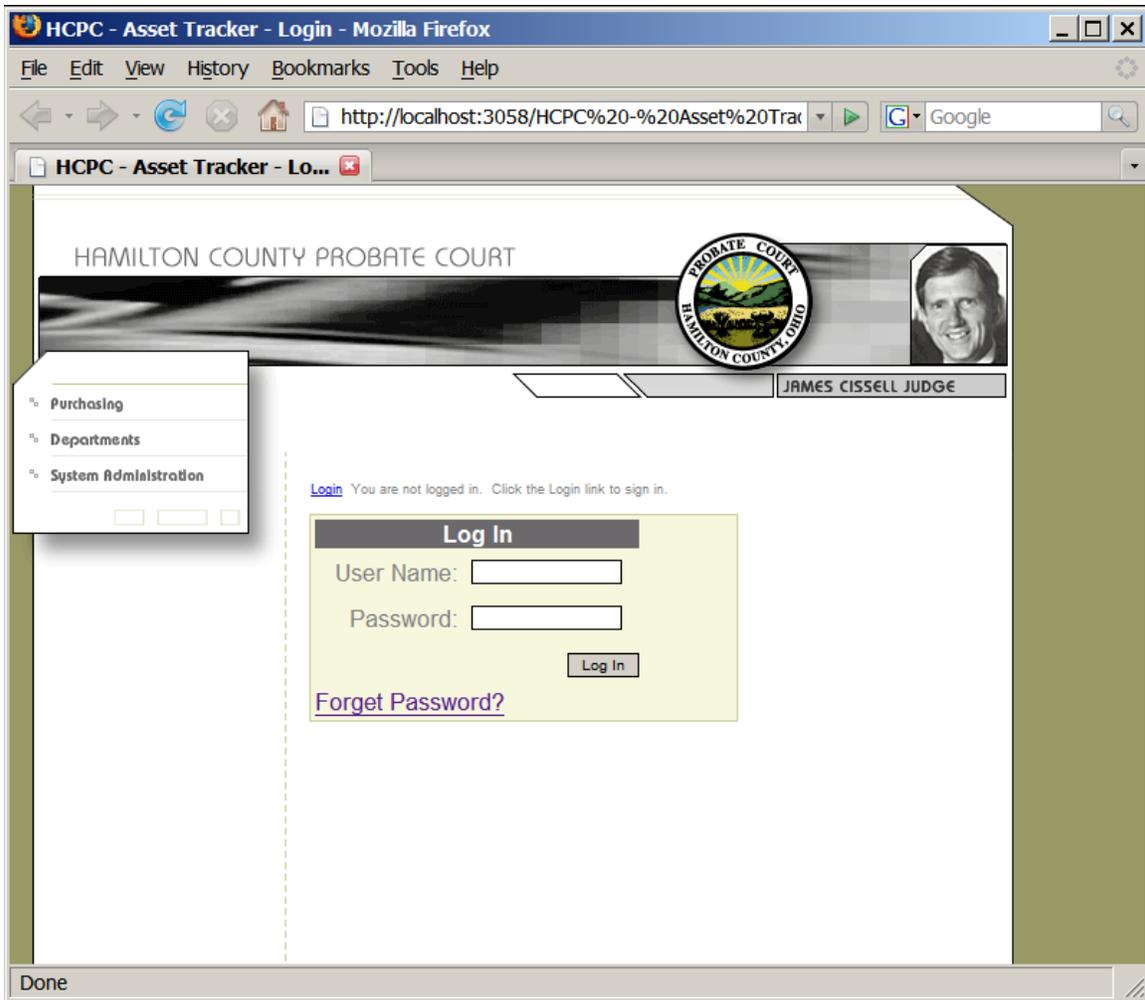


Figure 10 – Login page

5.2 The Dashboard

The Dashboard page provides quick navigational access to the various resources assigned to a particular user. The links are made visible only after reviewing the roles that the user has been assigned using the Roles class in the “System.Web.Security” namespace in ASP .NET. This page serves as the index, or home page, for the Asset Management Application. From here, users can be redirected to Add An Asset, the Inventory Browser, Pending Transactions, Reporting, and/or System Administration provided that the user has the proper security clearance (See Figure 11).

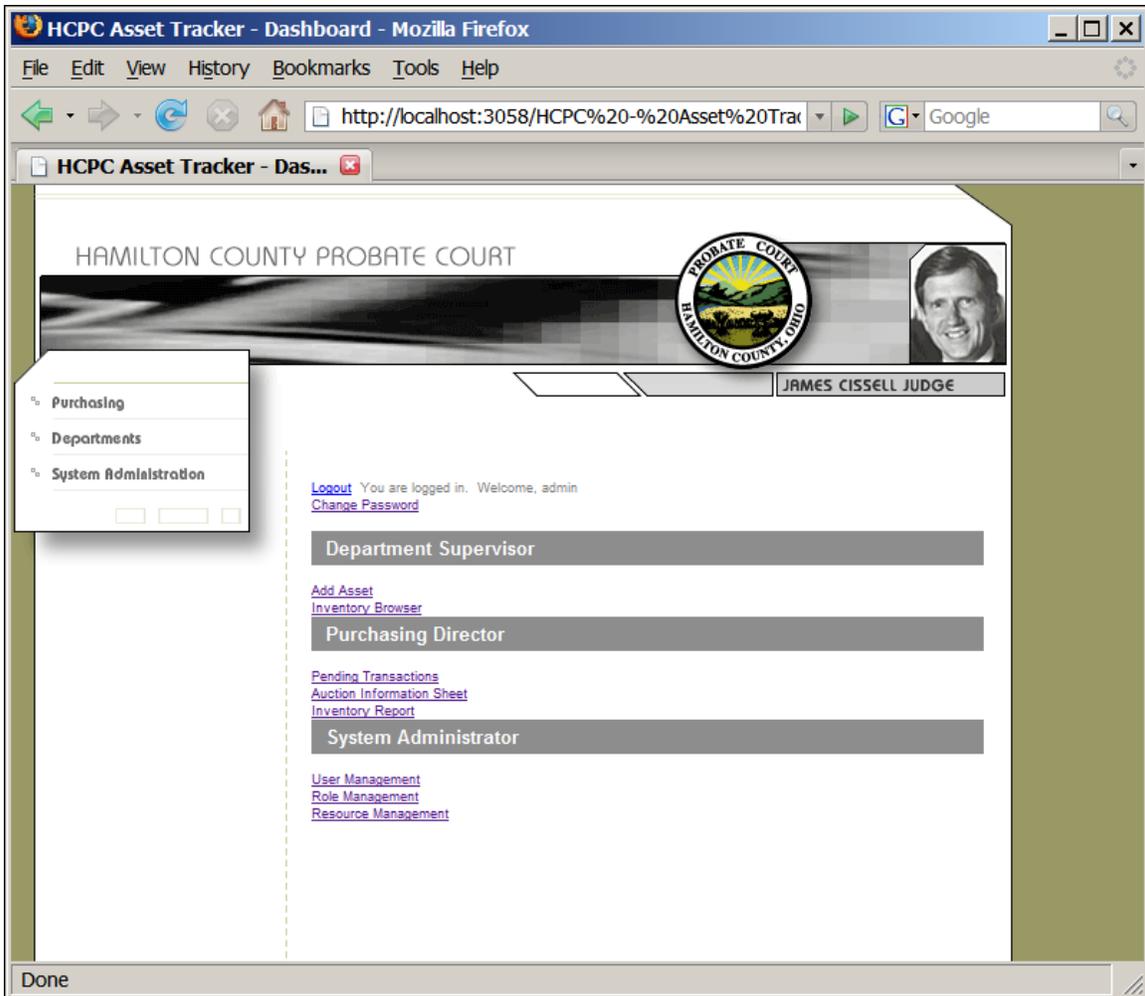


Figure 11 – Dashboard page

5.3 Adding a New Inventory Asset

Adding a new asset to the inventory is accomplished through the Add Asset page. Upon entering the necessary data for the asset item, a user can assign the asset a new barcode, or enter an existing barcode, and there is a link button for printing a new barcode to the label maker printer. In order to ensure proper data validation, the department, location, category, and subcategory fields are bound to existing valid data in the database. Furthermore, there are validation checks for the other fields as well. For example, before storing the data to the database Purchase Date is verified to be a valid

date. Upon clicking the Submit button, the Purchasing Director is notified via E-mail that an asset as been added (See Figure 12).

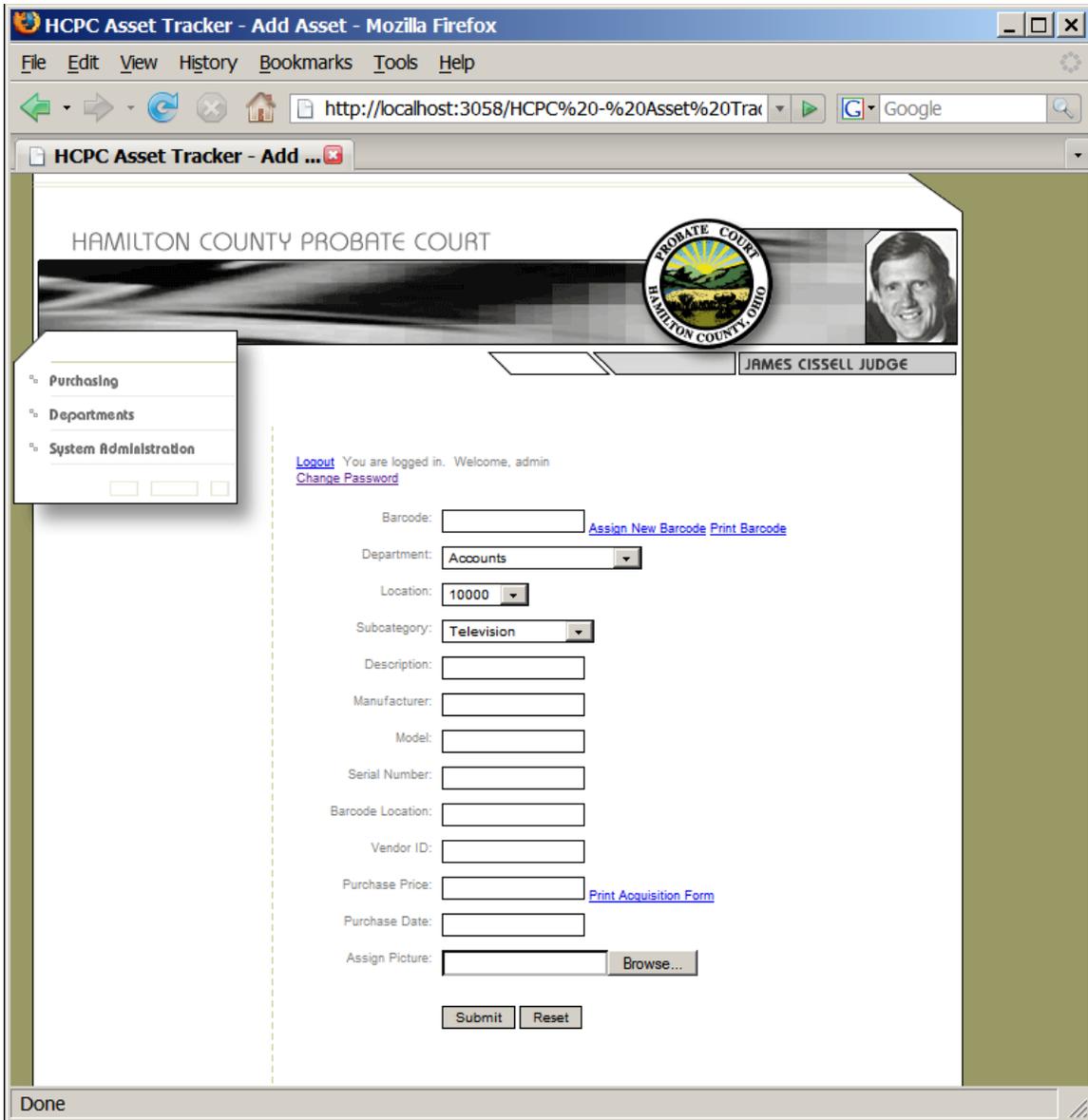


Figure 12 – Add asset page

5.4 The Inventory Browser

It is through the inventory browser that modifications and requests for transfers and disposals are conducted. The browser is designed so that, by default, all inventory items are displayed. Using drop down lists at the top of the page, a user can filter the displayed

assets by department, location, and/or category. Each inventory asset displayed has an Edit link. Once in Edit mode, the various fields can be modified. Again, for data validation, users are restrained to only using valid values from a drop down list for location id, subcategory, and status code. Moreover, there are links to assign a new barcode and print barcode. Requests for transfers and disposals are handled by changing the status to either “T” or “D”. Once modifications are complete, the user can click the Update link to save changes. For assets that have a status code changed to transfer or disposal, an E-mail is sent to the Probate Court’s Purchasing Director for approval. Figure 13 shows the Inventory Browser page.

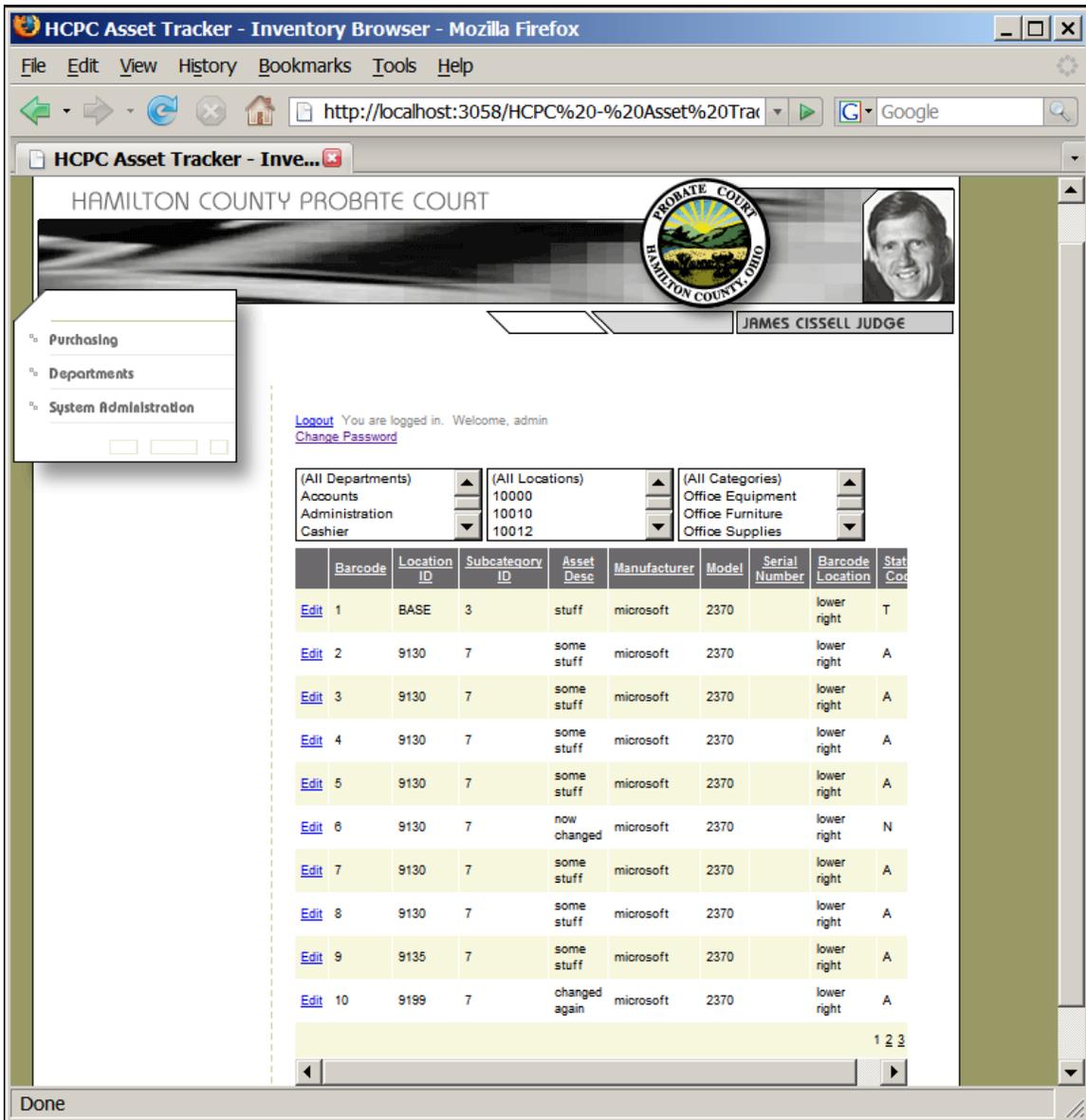


Figure 13 – Inventory browser page

5.5 Pending Transactions

The Pending Transactions page was developed for the Purchasing Director role, though the System Administrator has rights as well. This page allows the Purchasing Director to approve the addition, transfer, and/or disposal of an asset. To approve the transaction, the Purchasing Director clicks the Edit link for the asset and then changes the

status code. For newly added assets with a purchase price of \$5,000.00 or greater, the Purchasing Director has the option to click the Print Acquisition Form link. Likewise, for disposed assets, the Purchasing Director can choose to click the Print Auction Form for items being sent to the Hamilton County Auction (See Figure 14).

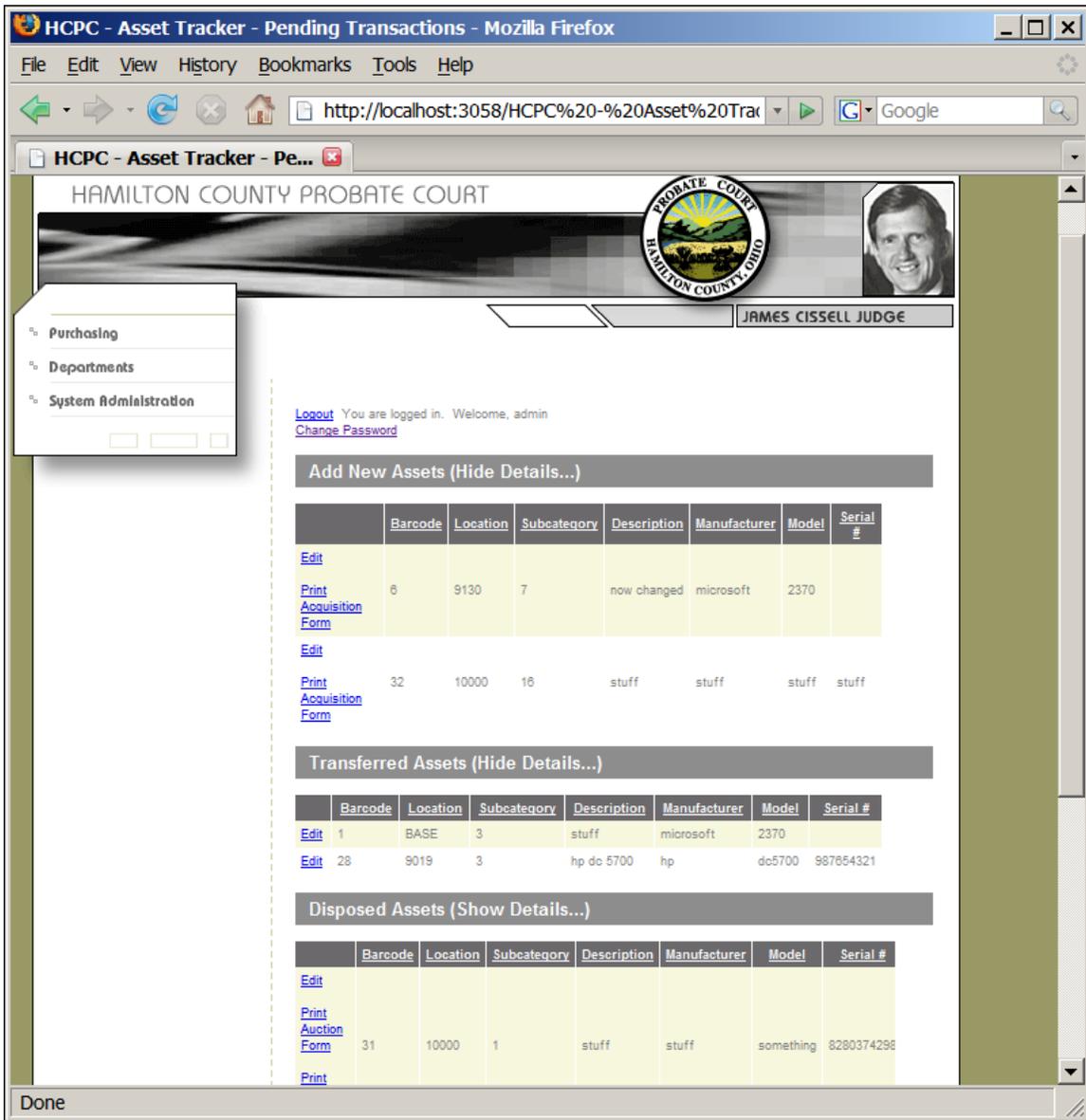
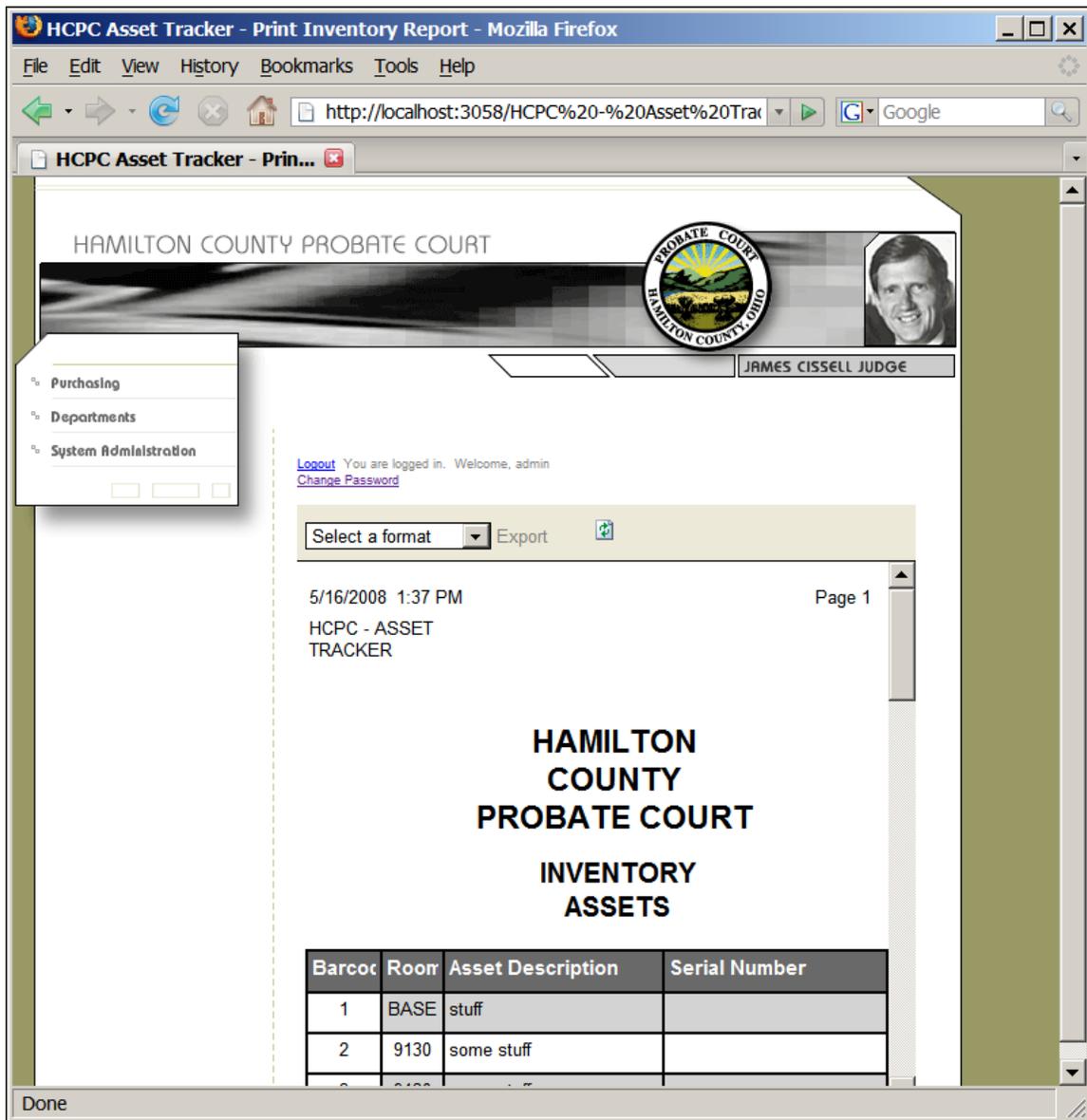


Figure 14 – Pending transactions page

5.6 Reporting

The Inventory Report that the Purchasing Director prepares for the Hamilton County Auditor is accessible through the Dashboard. This report was prepared using the built-in .NET reporting tools. It is a simply report that lists all of the Probate Court's inventory assets (See Figure 15).



The screenshot shows a web browser window titled "HCPC Asset Tracker - Print Inventory Report - Mozilla Firefox". The address bar shows the URL "http://localhost:3058/HCPC%20-%20Asset%20Trac". The page content includes the Hamilton County Probate Court logo and a portrait of James Cissell, Judge. A navigation menu on the left lists "Purchasing", "Departments", and "System Administration". The main content area displays the report title "HAMILTON COUNTY PROBATE COURT INVENTORY ASSETS" and a table of assets.

Barcode	Room	Asset Description	Serial Number
1	BASE	stuff	
2	9130	some stuff	

Figure 15 - Print inventory report page

The Capital Asset Acquisition and Disposal forms can be generated through the Pending Transaction page for new assets and assets being disposed that have a value of \$5,000.00 or greater. Likewise, the Auction Information Sheet can be printed from the Pending Transaction page or directly from the Dashboard page.

5.7 System Administration

Administration of the application, for example, adding new users, roles, and resources, is accomplished through the User Management, Role Management, and Resource Management pages respectively. All three of these pages are accessible via the Dashboard page. Furthermore, for greater control over system administration, the ASP .NET Web Site Administration Tool can be used to those who have access to the Asset Management Application's source code through Visual Studio 2005.

5.7.1 User Management

In order to manage users through the User Management page, an authenticated user assigned the System Administrator role can simply follow the link on the Dashboard page. The User Management page consists of three panels: Add User, Manage User, and Delete User (See Figure 16).

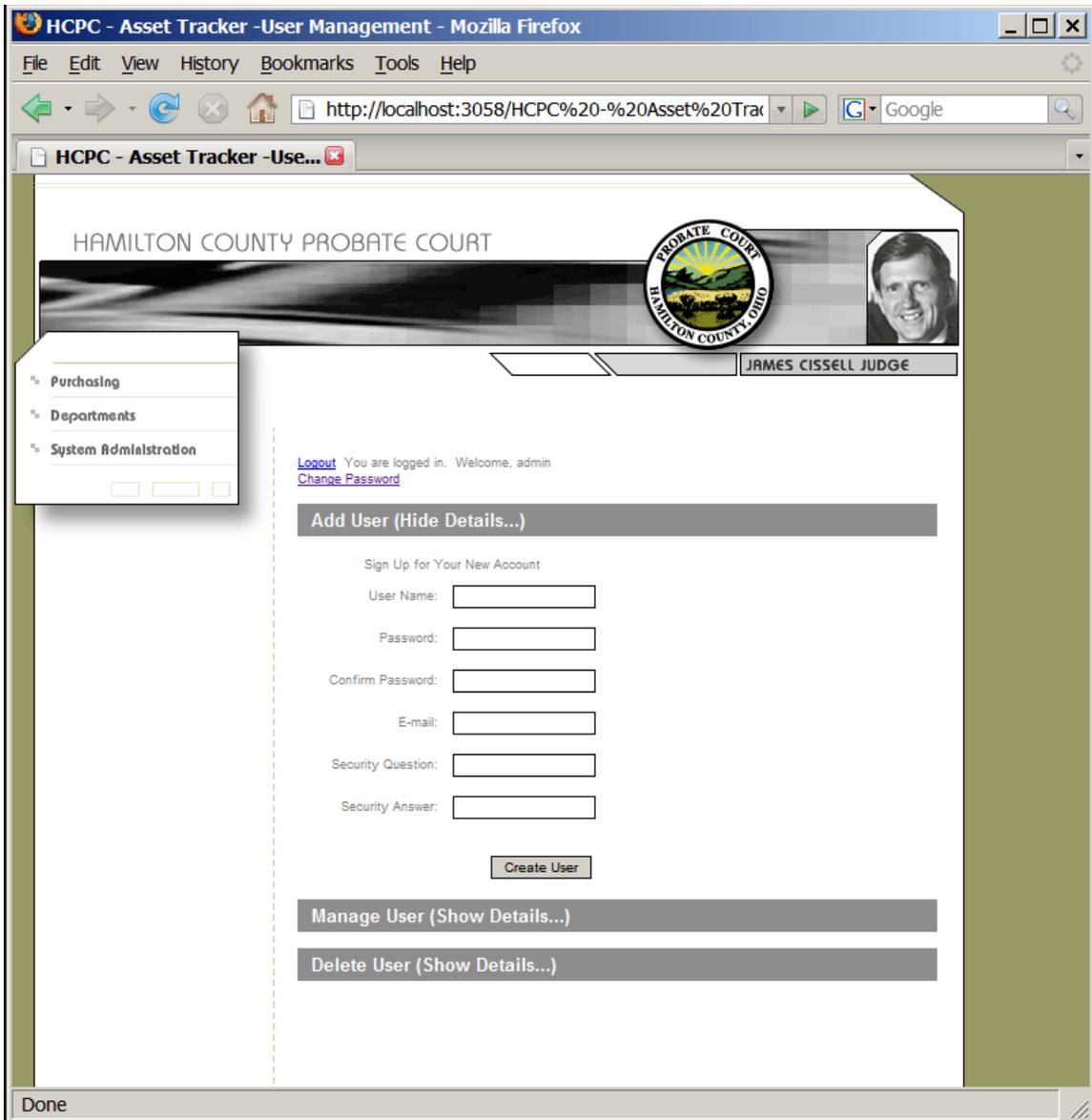


Figure 16 – User management page

To add a user to the system a user name, password, E-mail address, Security Question, and Security Answer are provided. All fields are validated for proper input and the password must be at least seven characters with at least one non-alphanumeric character. The “Manage User” section entails selecting a user and then checking which roles the user is to be assigned/unassigned. Finally, the “Delete User” only requires a user, or users, to be selected for deletion upon clicking the Submit button. These User Interfaces

all take advantage of both the Role and Membership classes in the “System.Web.Security” Namespace available with ASP .NET 2.0.

5.7.2 Role Management

Similar to the User Management page, the Role Management page is accessible from the Dashboard page to all System Administrator users. This page is also divided into three sections: Add Role, Manage Role, and Delete Role (See Figure 17).

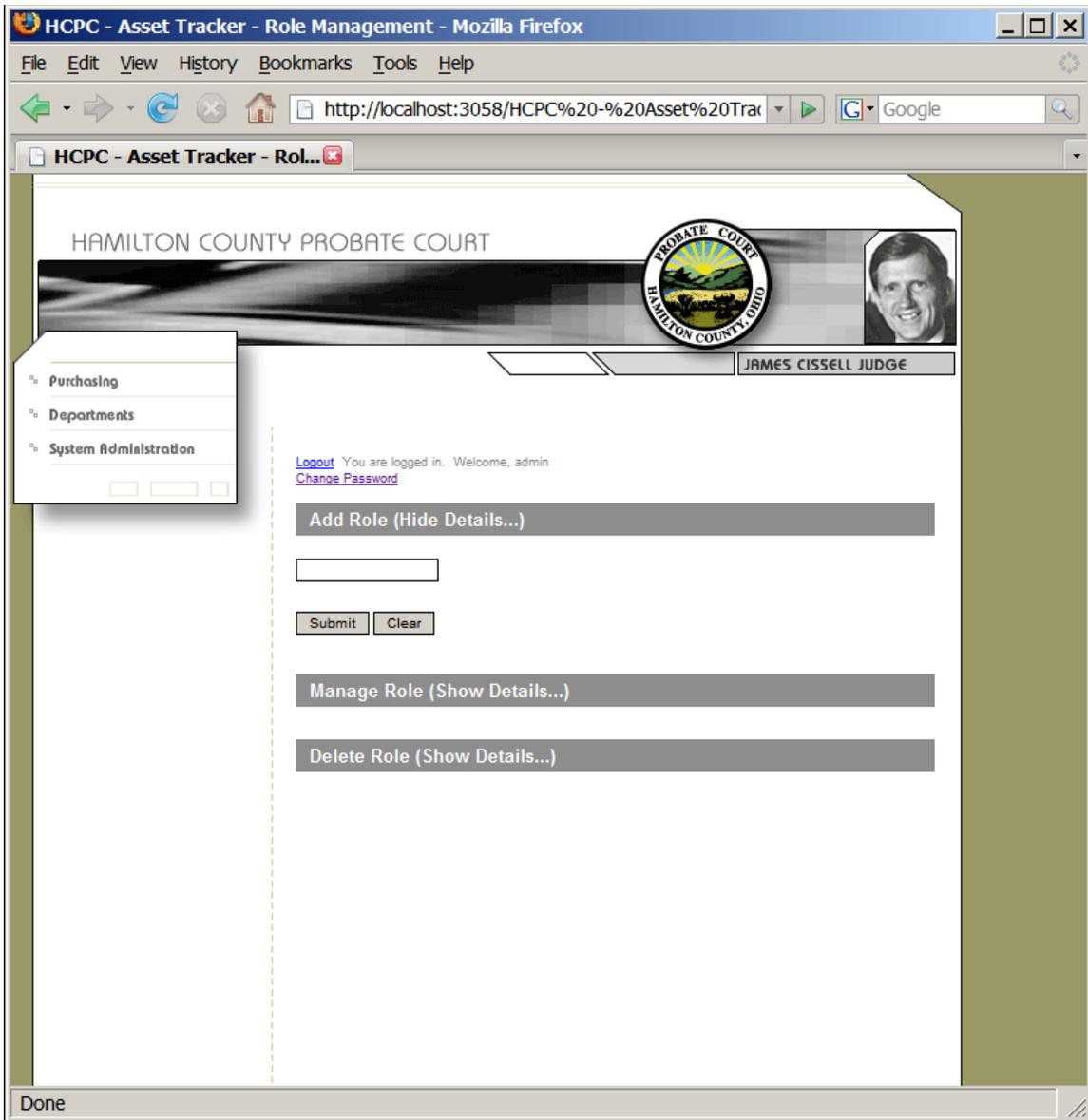


Figure 17 – Role management page

Adding a role simply requires a role name to be entered. Role management is a secondary way to assign users to roles. After selecting a role from a drop down list, the System Administrator selects which user, or users, to assign to the selected role. Under the “Delete Role” section, putting a check mark next to the role, or roles, and clicking the Submit button will delete a role. Like the User Management page, the “System.Web.Security” Role and Membership classes are used for Role Management.

5.7.3 Resource Management

For consistency, the Resource Management page was designed like the User Management and Role Management pages. Separate sections exist for the management of categories, subcategories, departments, asset images, locations, and status codes. However, these resources relate to data tables in the MySQL database management system. Therefore, any changes made to these items are written directly to the Inventory database. Furthermore, any image files are first uploaded to a directory on the server hosting the intranet site, and then a database record is created to reference that particular image file. It is this database record that is related to an asset record through the Add Asset page, the Inventory Browser page, or the Pending Transactions page.

6 Testing Plan and Procedures

Because of the small scale of the project and the relatively few users, testing for this application remained fairly simple. During the design phase, the Purchasing Director, IT Director, and author, were responsible for acceptance testing of the requirements definition. Likewise, we performed a simple system test of the logical design of the application. The author was responsible for unit testing the detailed designs and the individual deliverables as they were coded (See Figure 18).

After all of the deliverables were completed, a working model of the application was system tested by the Purchasing Director and a Department Supervisor (See Appendix B). They returned any changes and needed revisions to the author. This process continued iteratively until the application was ready for acceptance testing by the Purchasing Director and the IT Director. Upon final acceptance, a final data migration populated the new database and the system was made available for authorized users.

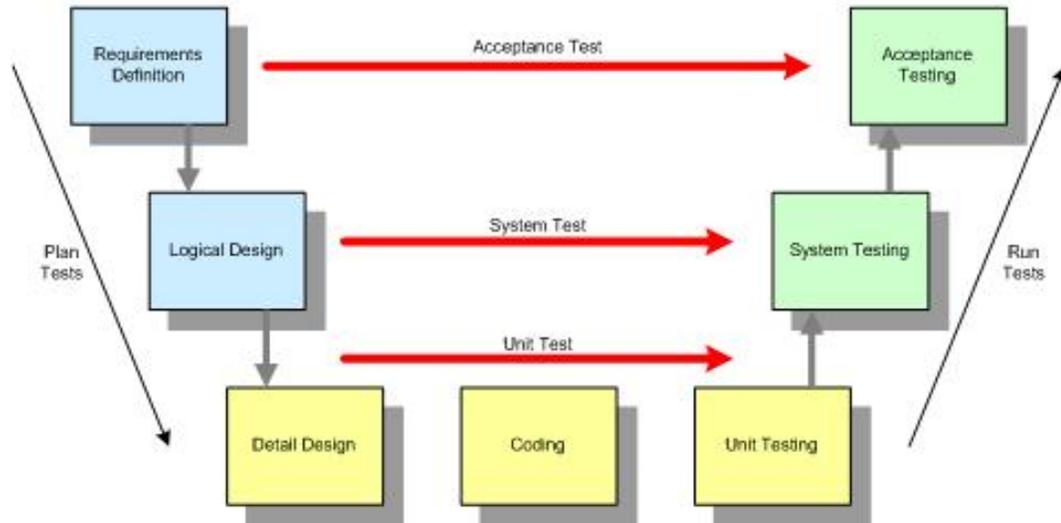


Figure 18 – Test Plan (1)

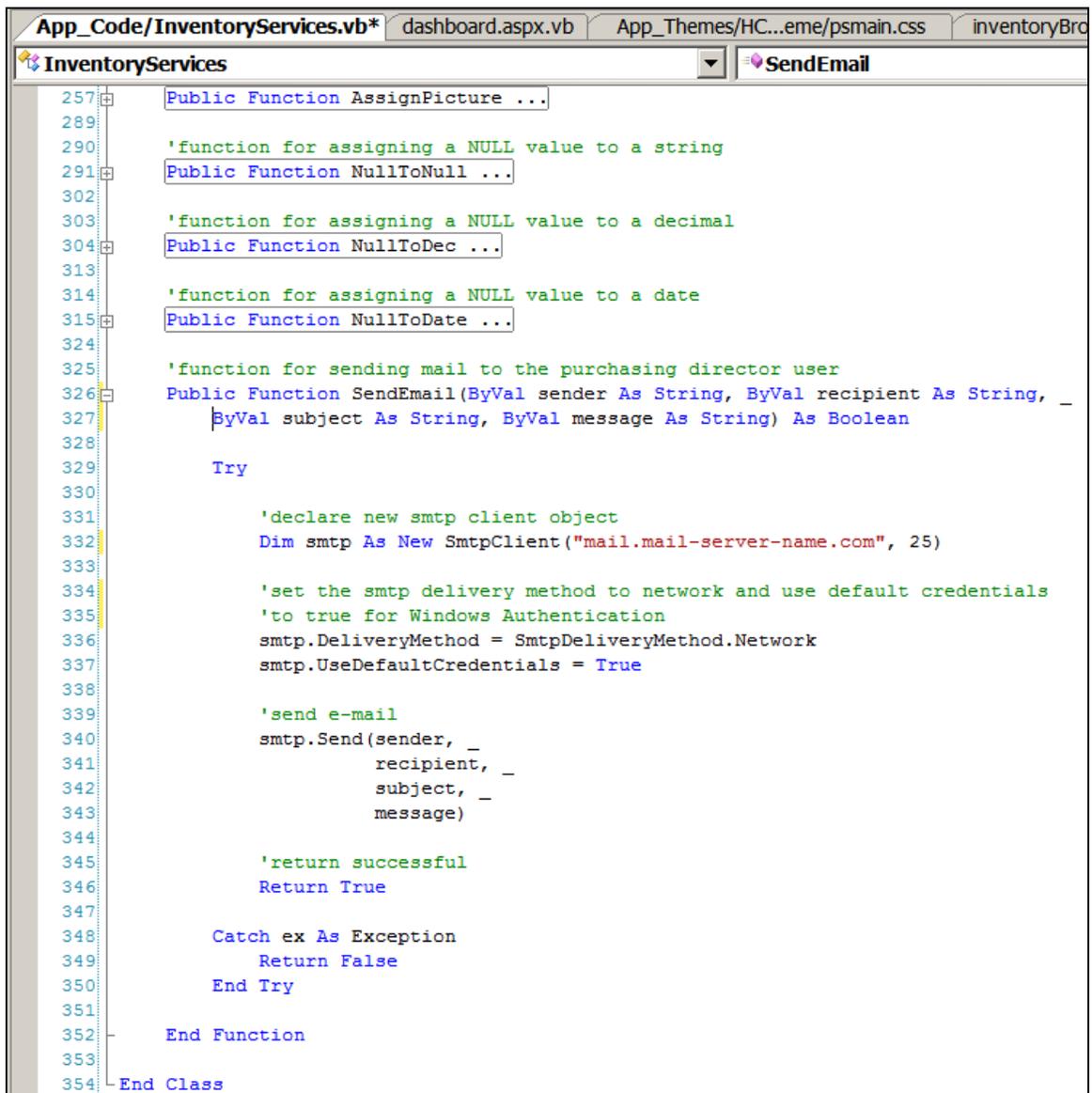
7 Conclusions and Recommendations

The Hamilton County Probate Court was desperately in need of an application to manage the court’s fixed assets. This application needed to provide the ability to add, modify, transfer, and dispose of court assets by authenticated users. The application developed consisted of three different types of user roles: Department Supervisors, Purchasing Director, and System Administrators. Deliverables consisted of documentation, a MySQL database, and various ASP .Net Web forms and classes. The Unit, System, and Acceptance Testing plans ensured the application would be successful and would fulfill the necessary requirements.

Appendix A.

VB .NET Code Sample

The Inventory Services class was developed to handle some of the Asset Management Application's utility procedures. For example, the code sample in Figure 19 shows the code used to send E-mail.



```
App_Code/InventoryServices.vb* dashboard.aspx.vb App_Themes/HC...eme/psmain.css inventoryBro
InventoryServices SendEmail
257 Public Function AssignPicture ...
289
290 'function for assigning a NULL value to a string
291 Public Function NullToNull ...
302
303 'function for assigning a NULL value to a decimal
304 Public Function NullToDec ...
313
314 'function for assigning a NULL value to a date
315 Public Function NullToDate ...
324
325 'function for sending mail to the purchasing director user
326 Public Function SendEmail(ByVal sender As String, ByVal recipient As String, _
327     ByVal subject As String, ByVal message As String) As Boolean
328
329     Try
330
331         'declare new smtp client object
332         Dim smtp As New SmtpClient("mail.mail-server-name.com", 25)
333
334         'set the smtp delivery method to network and use default credentials
335         'to true for Windows Authentication
336         smtp.DeliveryMethod = SmtpDeliveryMethod.Network
337         smtp.UseDefaultCredentials = True
338
339         'send e-mail
340         smtp.Send(sender, _
341             recipient, _
342             subject, _
343             message)
344
345         'return successful
346         Return True
347
348     Catch ex As Exception
349         Return False
350     End Try
351
352 End Function
353
354 End Class
```

Figure 19 – Code sample from the Inventory Services Class

Appendix B.

Testing Procedures

The following table (See Figure 20) shows an example of the testing procedures developed for the system-testing phase of the Asset Management Application. A similar procedure exists for each Web page of the application. The Purchasing Director and a Department Supervisor stepped through each item in the procedure and noted the actual results they received.

Module: addAsset.aspx

Item	Instruction	Test Date	Expected Results	Actual Results	Accepted Y/N
1	Manually enter barcode. Fill in the remaining fields. Click "Add Asset".		A "Store Successful" message should appear. Asset is added to the database.		
2	Click the "Assign New Barcode" link. Fill in the remaining fields. Click "Add Asset".		The barcode field should automatically be populated with the next available barcode number. A "Store Successful" message should appear. Asset is added to the database.		
3	Click the "Print Barcode" link		A printed barcode label should be generated and printed based on the text in the Barcode field.		
4	Leave the "Description" field blank.		An required field error should display. Data should not be stored.		
5	Fill in the all fields. Click the "Clear" button.		All fields should be reset to default values. Data should not be stored.		
6	Fill in the all fields. Enter a "Purchase Price" greater than \$4,999.99. Click the "Print Capital Acquisition Form" Button.		The user should be redirected to capitalAssetAcquisition.aspx		

Figure 20 – Testing procedure for the addAsset.aspx Web page

Bibliography

1. Kenzie, Anthony. "OCN Test Plan – ETL Process". February, 2008.
2. Larman, Craig. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Third Edition*. New Jersey: Pearson Education, Inc., 2004.
3. Lawriter, LLC. "Ohio Laws and Rules". October 30, 2007. <http://codes.ohio.gov/>
4. Peterson, Raymond H. *Accounting for Fixed Assets 2nd Edition*. New York: John Wiley and Sons, Inc., 2002.
5. Weikel, Scott. Purchasing Director, Hamilton County Probate Court. Personal interview. October 2, 2007.