

Knowledge Management System

By

Tim Walker

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

March 2004

Knowledge Management System

by

Tim Walker

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 2004 Tim Walker

The contents of this document are under copyright of the author. It may not be reproduced and distributed in whole or in part without the written permission of the author.

Tim Walker

Date

Annu Prabhakar, Faculty Advisor

Date

James Sullivan, Department Head

Date

Acknowledgements

I would like to give special thanks to all parties involved for their help in achieving such a successful project. This includes Professor George Suckarieh for creating the need for such a project and having an open mind in re-defining his vision, Professor Annu Prabhakar for her help and direction as my advisor, Chris Jaehnen for his help in designing the user interface of the project, Bill Krausen, current Vice President of Training at Messer Construction, for his support and focus, Jim Brady, Director of Messer Information Technologies, for his dedication and trust in my idea, and finally Scott Grothaus, Database Analyst at Messer Construction, for his help in deploying and supporting the Knowledge Management System.

Table of Contents

Section	Page
Acknowledgements	i
Table of Contents	ii
List of Illustrations	iii
Abstract	iv
1. Statement of the Problem	1
2. Description of the Solution	1
2.1 User Profile	4
2.2 Design Protocols	4
3. Deliverables	9
4. Design and Development	10
4.1 Timeline	10
4.2 Budget	10
5. Proof of Design	11
5.1 The End-User Application	11
5.1.1 Hierarchal Tree Control	12
5.1.2 Search Functionality	13
5.1.3 Print and Save Functionality	14
5.2 The Administrative Application	15
5.2.1 User Administration	15
5.2.2 Subtopic Administration	18
5.2.3 Document Administration	22
5.2.4 Exporting Documents	26
5.2.5 Importing Documents	26
5.2.6 Indexing Documents	27
5.2.7 Refreshing the End-User Web Tree Control	28
5.3 The COM+ Component	29
6. Testing Procedures	29
7. Conclusions and Recommendations	30
7.1 Conclusions	30
7.2 Recommendations	30
Appendix A Knowledge Management System Object Diagram	32
References	35

List of Figures

Figure 1. Initial Design	2
Figure 2. N-tier Architecture	5
Figure 3. Admin User Interface	7
Figure 4. End-User Interface	8
Figure 5. Entity Relationship Diagram	9
Figure 6. Hierarchal Tree Control	12
Figure 7. Full Screen Document View	13
Figure 8. Search Documents	14
Figure 9. Print and Save View	15
Figure 10. List of Users	16
Figure 11. Add New User	17
Figure 12. Editing an Existing User	18
Figure 13. List of Subtopics and Documents	19
Figure 14. Add New Subtopic	20
Figure 15. Editing an Existing Subtopic	21
Figure 16. List of Subtopics and Documents	22
Figure 17. Add New Document	23
Figure 18. Edit an Existing Document	24
Figure 19. Verify Document Change	25
Figure 20. Export Folder Selection Box	26
Figure 21. Import Folder Selection Box	27
Figure 22. Refresh Web Tree	28

Abstract

The Knowledge Management System is an online document management application that indexes and stores any file format in a hierarchal manner allowing each stored piece of information to be easily found by the end user. Specifically designed for use at Messer Construction to house documents, movies, and images pertaining to the construction industry, the Knowledge Management System can be integrated and used in any industry. The following document describes the process of designing, developing, testing, and implementing the Knowledge Management System.

Knowledge Management System

1. Statement of the Problem

This system was initially conceptualized after conversations with Dr. George Suckarieh, Professor Annu Prabhakar, and executives at Messer Construction. Messer Construction was interested in a system initially proposed by Dr. George Suckarieh to allow transfer of knowledge from experienced construction professionals and project managers to new graduates and employees. Messer Construction's initial problem focused on the fact that as experienced construction professionals left their institution, the knowledge that they had acquired from years of experience left the company as well. They needed an easily accessible central repository to house information and documentation on procedures and practices from experienced construction professionals as well as safety manuals, and reference materials.

2. Description of the Solution

The initial design of the Knowledge Management System was based on a template-based, static HTML system proposed by Dr. Suckarieh. He intended to create modules that had static elements that could be re-used for each category housed in the Knowledge Management System. Each category would have the same defined elements as tabs across the top of the page and would consist of static HTML pages. An example of his initial design can be seen below (See Figure 1.).

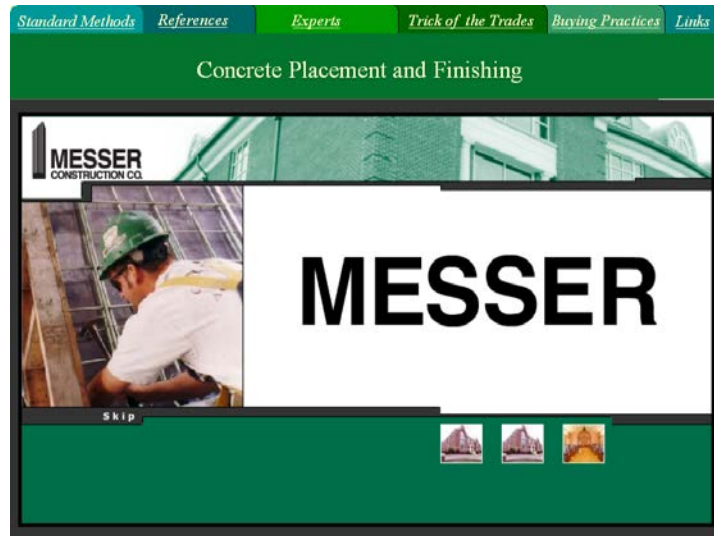


Figure 1. Initial Design

Upon further investigation into this design, it was decided that although this design would be a viable option, the amount of time it would take for creating each module along with the maintenance involved in maintaining each static HTML page would be an expensive and time consuming solution for Messer Construction. It was also determined that while the idea of a template for each category was excellent for standardization, some of the categories did not fit within the initial proposed template.

A new solution was proposed that would allow a more scalable application and be easily managed by Messer Construction. Instead of static HTML pages, the system would store the actual documents in a backend database system. These documents would be indexed and searchable and would have their hierarchal relation stored as well. This would allow users to browse the documents using a hierarchal tree view control, or they could use the search feature to find documents that were indexed by the application.

After reviewing the new solution with Messer Construction, Dr. Suckarieh, and Professor Prabhakar, it was determined that the application must have the following functionality:

- The administrative and end-user interface must be easy to use with little to no training;
- The application must be built using existing supported technologies that Messer Construction already has in place;
- The solution must be flexible so that new modules can be added and removed without additional programming;
- The application must have seamless authentication and log documents viewed by each employee;
- The solution must be reliable and scalable so that as more and more documents are added, the application will not be rendered useless;

The Knowledge Management System solved all of these requirements. The administrative user interface is easy to use and allows the user to use the native file editor to make changes to documents in the Knowledge Management System using software they are already familiar with. For example, if a Microsoft Word document is stored in the Knowledge Management System, Microsoft Word is the editor for that document. The end-user interface is also extremely easy to use and is familiar to users since it is based on Microsoft Windows Explorer. The application was built using Microsoft SQL Server 2000, Microsoft Visual Basic 6.0, and Microsoft Active Server Pages. These technologies were used since this was the newest programming and database applications supported by Messer Construction. For flexibility, any number of modules can be added or removed at anytime for the Knowledge Management System. To handle seamless authentication and logging, Integrated Windows Authentication was used inside of Microsoft Internet Information Server and every document viewed was logged with the

authenticated username. To ensure reliability and scalability, an n-tier design was used so that as the solution grew, more and more servers could be added to handle the workload.

2.1 User Profile

The Knowledge Management System was designed for two different categories of users: the end-user which would use the intranet to access documents and information and the administrative-user who would manage users, subtopics, and documents.

End-users consisted of the bulk of the users who would use the Knowledge Management System to search and find information pertaining to their interests. They could also peruse documents by using the hierarchal tree control as discussed in section 5.1.1.

Administrative users were made up of co-ops that would interview experts in the field and use a Microsoft Visual Basic 6.0 Windows application to add, edit, and remove users, modules, and documents. To maintain the Knowledge Management System administrative users should be familiar with the Microsoft Office Suite along with any other file editor of the file types stored in the Knowledge Management System.

2.2 Design Protocols

The Knowledge Base was designed using an N-Tier architecture for the application from its first inception. The decision to use an N-Tier architecture was decided upon due to making development and maintenance easier and helping with scalability while providing a centralization of the code (1).

The components of the N-Tier architecture used in designing the Knowledge Base can be broken into 3 categories. These categories are the user interface tier, the application or business logic tier, and the data tier. The user interface tier can be further divided into the

administration interface and the end-user interface that share the components of the business logic and the data tiers as shown below (See Figure 2.).

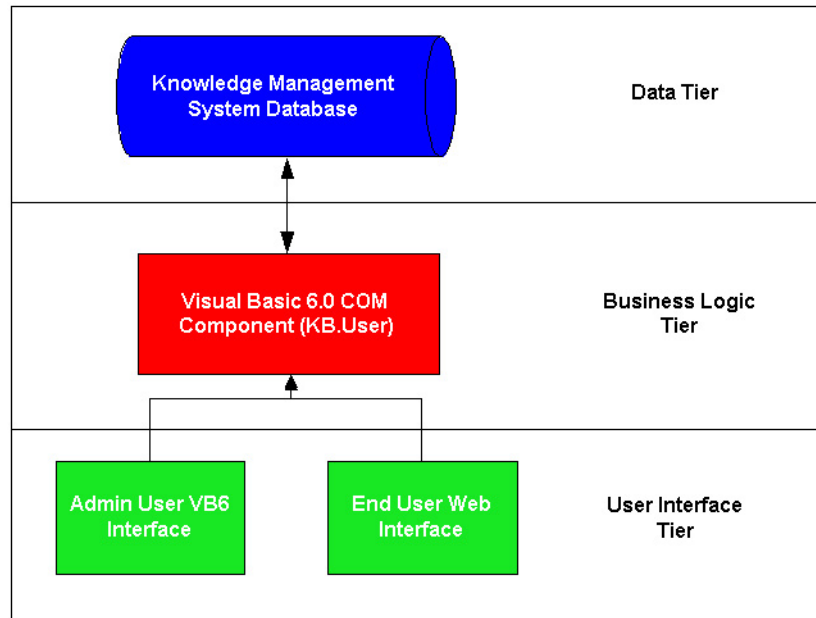


Figure 2. N-tier Architecture

Messer’s technology infrastructure and skill sets were heavily focused on Microsoft Visual Basic 6, Microsoft SQL Server 2000, and Microsoft Internet Information Server. Due to this focus, the decision was made to create the Knowledge Base using these same technologies. In addition to these technologies, it was decided to use the native file editor for editing documents (e.g. MS Word for MS Word documents). This gave the administrative user the most control over the documents without forcing the administrative user to learn HTML or any other technologies to post documents to the web. It was also decided to use Microsoft SQL Server’s Full Text Indexing (2) feature to allow easy indexing of all documents submitted to the Knowledge Base.

Before beginning the description of the user interface, a couple of terms should be defined for better understanding of the Knowledge Base system.

1. **Subtopic:** This is a container object that allows for storing of documents or subtopics. It can be compared to a MS Window's folder that can contain zero to many files (documents in the Knowledge Base) and zero to many folders (subtopics in the Knowledge Base).
2. **Document:** The document is any type of file that you wish to store in the Knowledge Base. It can be an image, an MS Word document, an MS Excel spreadsheet, or any other binary file. To be indexed and searchable the document must be either an MS Office file, an Adobe PDF file, or the proper IFilter (2) installed on the Knowledge Base database server.
3. **Knowledge Base Tree:** The structure that represents the Knowledge Base. This is similar to MS Window's file system in which there is a root container that houses all the folders (subtopics in the Knowledge Base) and files (documents in the Knowledge Base).

The administration user interface (See Figure 3.) allows an administrator of the Knowledge Base to add, edit, update, and remove users, subtopics, and documents. It also allows the administrator to easily import and export any portion of the Knowledge Base tree along with indexing the newly created documents and refreshing the cached web tree if new documents are added to the tree (more on this in the next paragraph). The interface allows easy manipulation of the documents using the program that is associated with the document type as the editor. For example, if a document that was created in Microsoft Word is opened for editing, Microsoft Word is then used as the editor to make changes to the document. The changes are then saved back to the Knowledge Base database and the document is re-indexed. The administration user interface is designed

using Microsoft Visual Basic 6.0 and interfaces to the business logic tier using COM+. This piece of the Knowledge Base was the last to be built due to the small number of users that would use the interface. Initially, loading of the Knowledge Base was done using an import tool that had very little functionality. After the completion of the administration user interface, the import tool was discarded and all importing is done using the administration user interface.

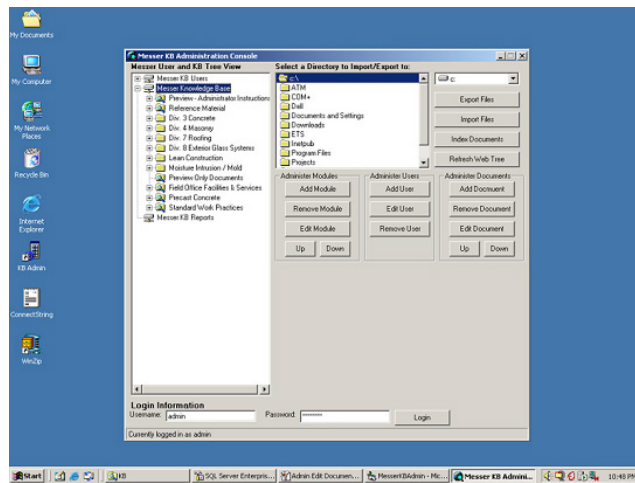


Figure 3. Admin User Interface

The end-user interface (See Figure 4.) is the main interface and is used by construction professionals to search and view documents, images, movies, and other file types. Construction professionals may either peruse the Knowledge Base tree looking for subjects of interest, or they may search the Knowledge Base by entering search terms in the Search box. The search may be limited to a certain category by selecting the category from the drop down box of top-level categories or the search may include the entire Knowledge Base. The interface is a web interface that is accessed by using Microsoft's Internet Explorer. The web server is Microsoft's IIS (Internet Information Server) and uses Microsoft's Active Server Pages to connect to the COM+ business layer interface. The Knowledge Base tree structure uses cascading style sheets for creating the Window's

Explorer like interface and is cached by the web server so that the tree is only created the first time the Knowledge Base is accessed allowing for a much greater response time. An Administrator of the Knowledge Base using the administration user interface may update the Knowledge Base tree structure that is cached. The user is automatically logged in to the end-user interface using Integrated Windows Authentication so that document access can be logged. The end-user interface logs each time a user views a document, capturing the time and date viewed along with the user that viewed the document.

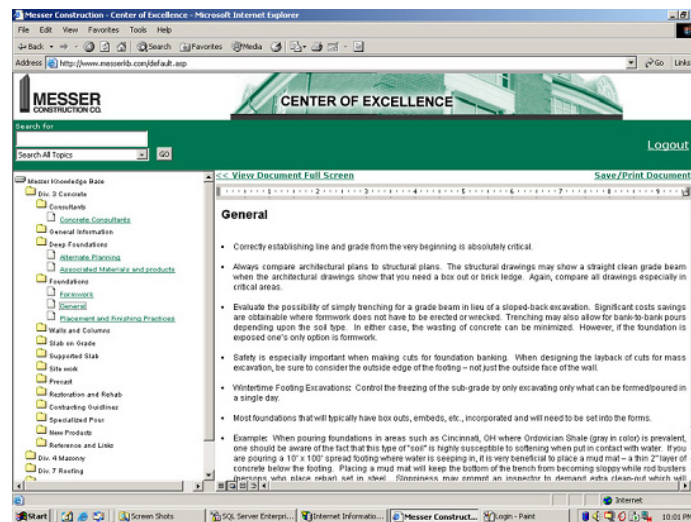


Figure 4. End-User Interface

The business logic tier is the workhorse of the Knowledge Base. All business logic is stored in this layer of the Knowledge Base, which includes creation, editing, and deletion of users, subtopics, and documents. The business logic tier is a Microsoft Visual Basic 6 COM+ application that resides on the application server. In the version that Messer uses for production and development, the business tier was loaded on the same server as the web server. If the application required a better response time or needed to handle more requests, the business tier could be moved to a separate server or even load balanced on multiple application servers.

The data tier (See Figure 5.) is the storage area for the Knowledge Base system. All documents, subtopics, and users are stored in this layer along with access logs and settings of the Knowledge Base. The data tier uses Microsoft SQL Server 2000 Standard Edition as the database server and makes extensive use of Microsoft's Full-Text Indexing. Full-Text Indexing allows all documents stored in the data tier to be indexed and searchable.

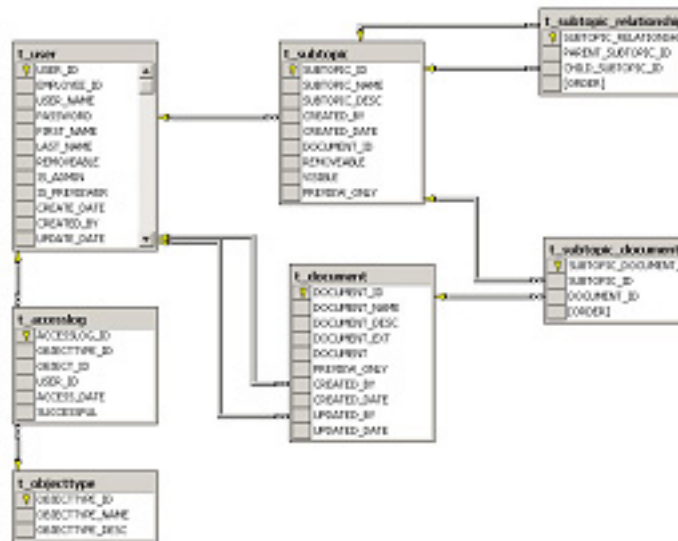


Figure 5. Entity Relationship Diagram

3. Deliverables

During the requirements gathering stage of this project, the following deliverables were identified in order for this to be a successful project:

- An intranet application that is easy to use and understand and groups the documents in a logical way of finding them
- Documents are easily uploaded to the Knowledge Management System using familiar tools already in use

- The Knowledge Management System is scalable and can be easily configured to fit any situation
- Seamless authentication using Integrated Windows Authentication inside of Internet Information Server
- Indexing and searching of MS Office and Adobe PDF files
- Logging of document access in order to do detailed reporting at a later time
- The ability for end-users to save and print documents stored inside the system

4. Design and Development

The following section is an overview of the project's timeline and an analysis of the budget required to develop the application.

4.1 Timeline

The following is a guideline for project milestones.

Project Stage	Date
Inception of project	November 15, 2002
Initial proposal for Knowledge Management System	November 20, 2002
Redesign of Knowledge Management System	December 1, 2002
Requirements gathering	December 15, 2002
Initial design document finished	January 1, 2003
Data tier development finished	January 10, 2003
Business tier development finished	February 1, 2003
End-user interface development finished	February 15, 2003
Import tool development finished	February 22, 2003
Testing of Knowledge Management System	March 1, 2003
Deployment to production servers	March 7, 2003
Administrative user interface development finished	March 15, 2003

4.2 Budget

The budget for developing and deploying the Knowledge Management System is outlined in the table below. Also note that some of the software, such as MS SQL Server 2000, can be used on one machine in multiple development environments.

Production Software		
Item	Cost	Actual Cost
MS Windows 2000 Server	\$999	\$0
MS SQL Server 2000 Standard Edition 1 Processor	\$4,999	\$0
Production Hardware		
Item	Cost	Actual Cost
Estimated Production Server Cost	\$5,000	\$0
Totals:	\$10,998	\$0

Staging Software		
Item	Cost	Actual Cost
MS Windows 2000 Server	\$999	\$0
MS SQL Server 2000 Standard Edition 1 Processor	\$4,999	\$0
Staging Hardware		
Item	Cost	Actual Cost
Estimated Staging Server Cost	\$2,000	\$0
Totals:	\$7,998	\$0

Development Software		
Item	Cost	Actual Cost
MS Windows 2000 Server	\$999	\$0
MS SQL Server 2000 Standard Edition 1 Processor	\$4,999	\$0
MS Visual Studio 6.0 Enterprise Edition	\$1500	\$0
Adobe Photoshop 6.0	\$649.00	\$0
Development Hardware		
Item	Cost	Actual Cost
Estimated Development Computer Cost	\$1,000	\$0
Totals:	\$9,147	\$0

5. Proof of Design

The following section is an overview of the finished product and how the finished product met all the deliverables mentioned in previous sections.

5.1 The End-User Application

The end-user application is the application used by the majority of the users. Developed using Microsoft Visual Interdev 6.0, Microsoft Active Server Pages, HTML, and cascading style sheets, the application runs on an internal web server as part of

The hierarchal tree control can also be hidden from view so that the entire document can be viewed using the “<< View Document Full Screen” hyperlink as shown below (See Figure 7.). By clicking the “>>” hyperlink, the tree view comes back into sight and can be used once again to peruse the Knowledge Management System’s documents.



Figure 7. Full Screen Document View

5.1.2 Search Functionality

The Knowledge Management System allows for easy retrieval of documents using the search capability as well. The search feature allows you to search all documents or just one of the top level categories (See Figure 8.), which is dynamically updated as subtopics are added, removed, and edited in the Knowledge Management System.

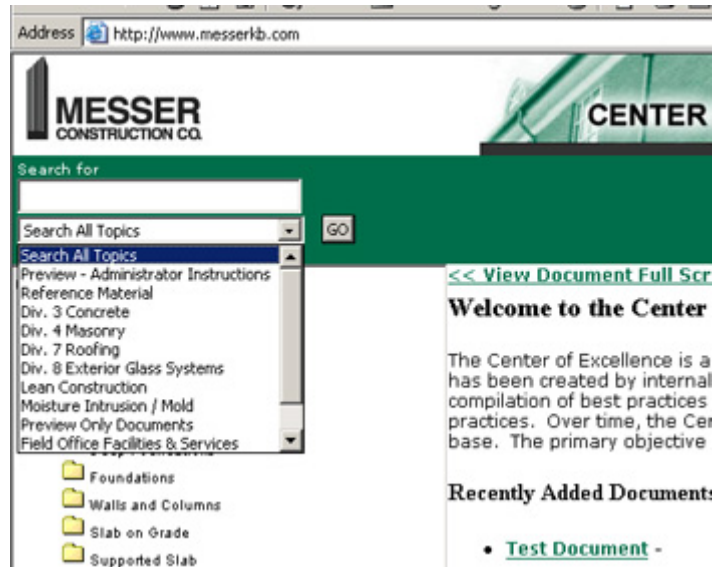


Figure 8. Search Documents

5.1.3 Print and Save Functionality

One of the deliverables of the project was to allow end-users of the Knowledge Management System to print and save documents stored inside the system. In order to provide this functionality due to limitations of MS Office documents displayed inside of an HTML frame, it was necessary to create a popup window that displayed the full document and had the MS Office toolbars so that the user can save or print the document as shown below (See Figure 9).

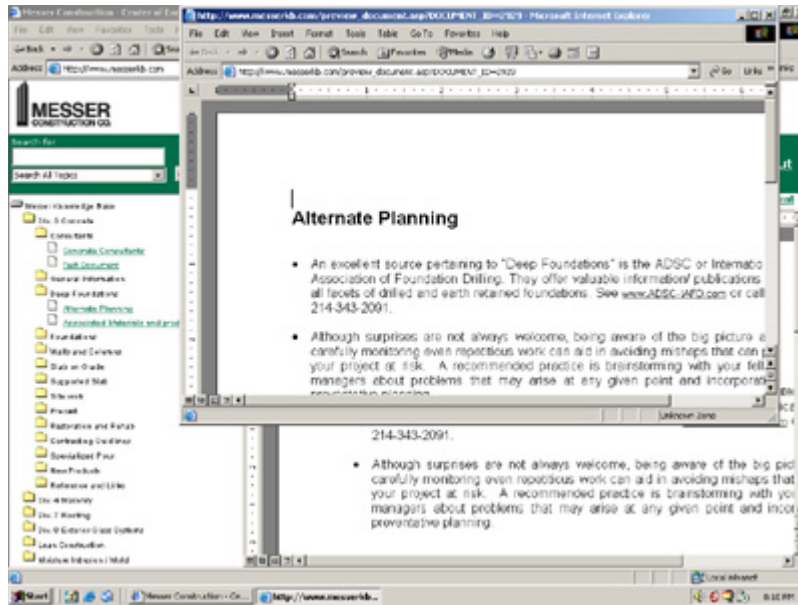


Figure 9. Print and Save View

5.2 The Administrative Application

The administrative application is the main interface for manipulating the Knowledge Management System. Designed using Microsoft Visual Basic 6.0, this interface allows administrative users to add, edit, and remove users, subtopics, and documents. It is a little more difficult to understand and use due to all the functionality built into the tool, but has many features that make it easy to use such as using the native file editor of the document to edit existing content.

5.2.1 User Administration

The administrative tool allows users to add, remove, and edit users that use the Knowledge Management System. Initially populated with users by exporting a comma-delimited list from Microsoft's Active Directory, the list of users is displayed on the left hand side of the administrative tool in the tree control as shown below (See Figure 10.).

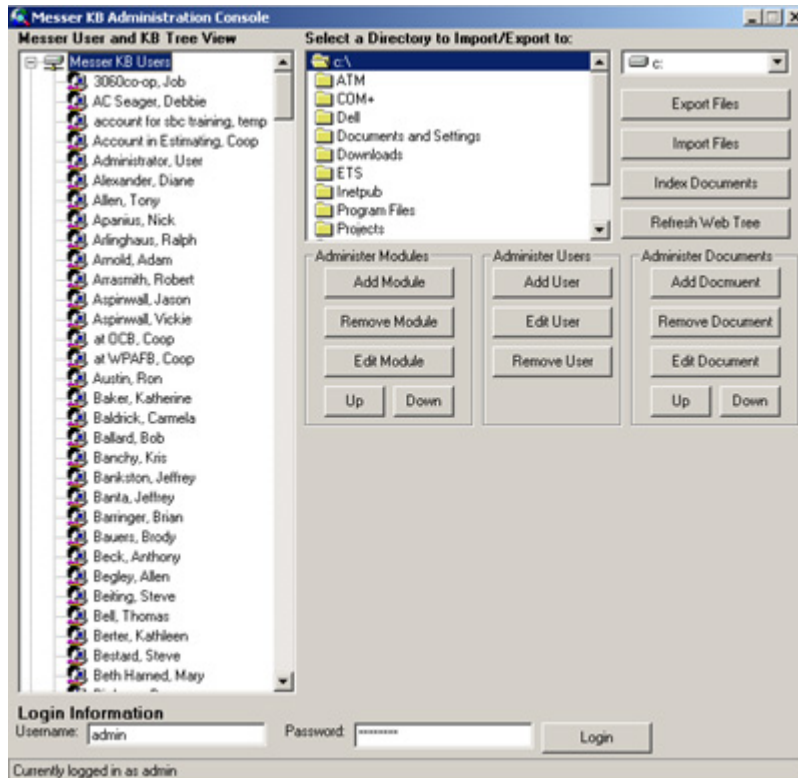


Figure 10. List of Users

To add a new user, the administrative user clicks the “Add User” button under the “Administrate User” section. The add new user frame will display as shown below (See Figure 11.) where the first name, last name, username, and password can be entered as well as administrative and preview privileges can be granted. The administrative privilege allows users to use the administrative application. The preview privilege allows users to view all documents and subtopics that have been marked as preview only. The preview functionality allows documents to be uploaded to the system and viewed only by designated “preview” users that can then verify the content and make the document or subtopic viewable for everyone. Once all the information has been entered, the user clicks the “Save” button to create the new user.

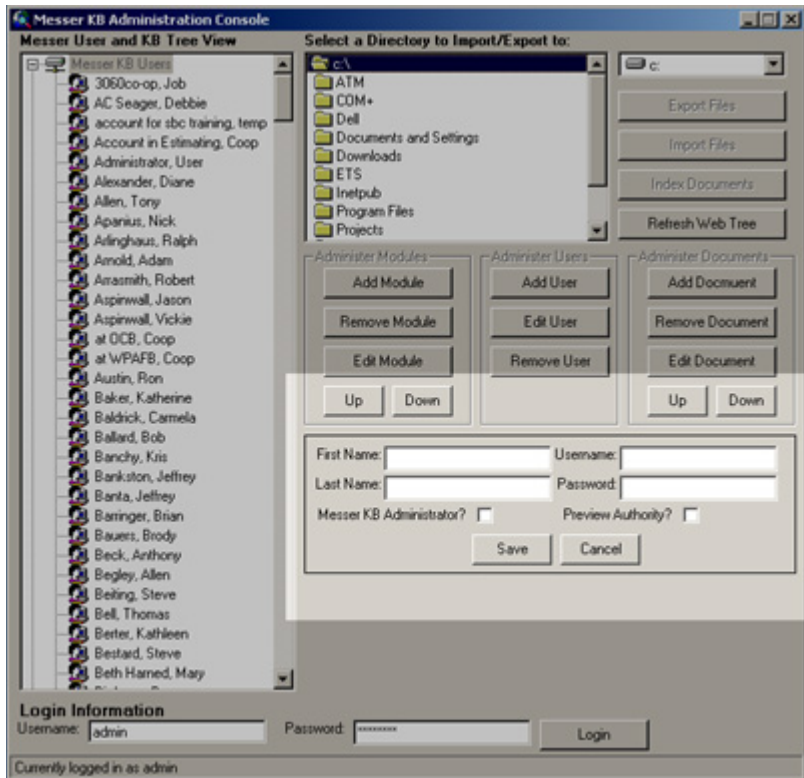


Figure 11. Add New User

To edit an existing user, the administrative user selects the user he wishes to edit from the list of users on the left and clicks the “Edit User” button under the “Administrate User” section (See Figure 12.). The user may also double-click the user in order to bring up the edit screen for the user as well. The administrative user can then make the necessary changes to the user account and save them by clicking the “Save” button.

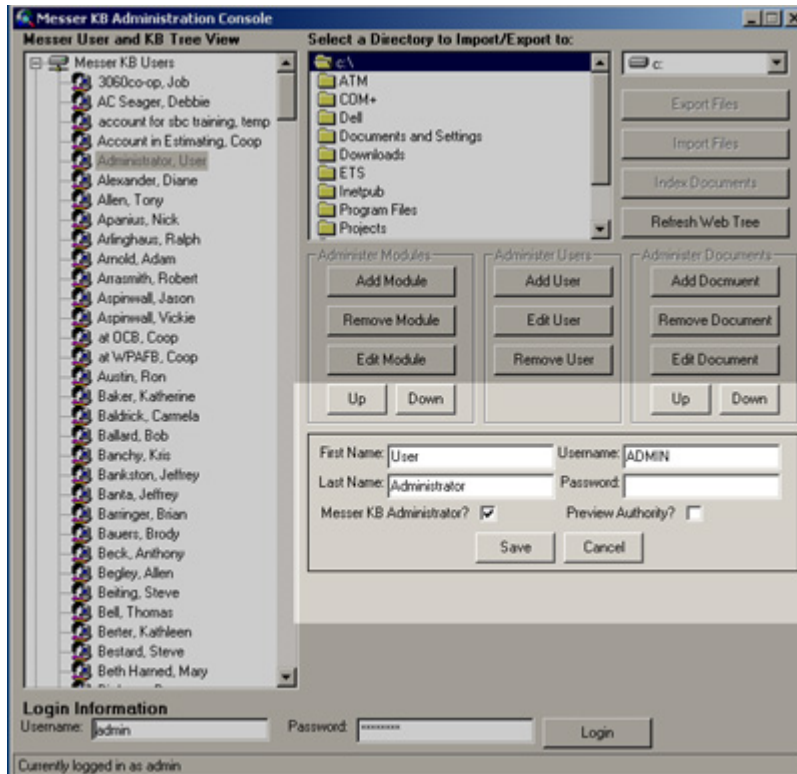


Figure 12. Editing an Existing User

To remove a user, the administrative user selects the user he wishes to remove from the list of users on the left and clicks the “Remove User” button under the “Administrate User” section. A message box will verify that the administrative user wants to delete the user, and the user will be deleted from the Knowledge Management System.

5.2.2 Subtopic Administration

The administrative tool allows users to add, remove, and edit subtopics that create the hierarchal relationship of the Knowledge Management System. The list of subtopics and documents is displayed on the left hand side of the administrative tool in the tree control as shown below (See Figure 13.).

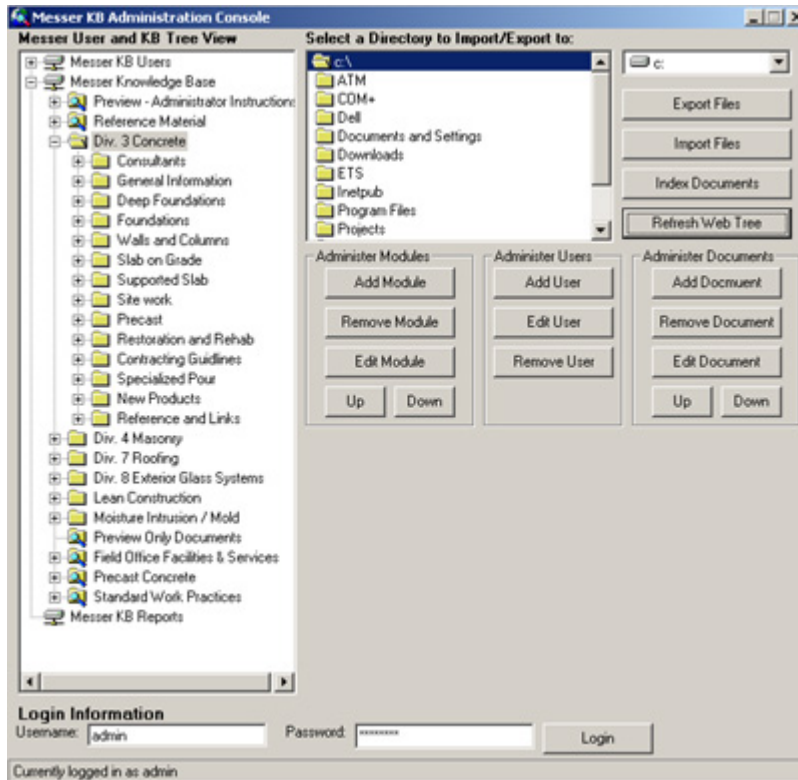


Figure 13. List of Subtopics and Documents

To add a subtopic to the Knowledge Management System, the administrative user selects the subtopic he wants to create the new subtopic in and selects “Add Module” under the “Administrate Modules” section. If the administrative user wishes to create a subtopic at the top level, he selects the “Messer Knowledge Base” root subtopic and then selects “Add Module” under the “Administrate Modules” section. The add new subtopic frame will display as shown below (See Figure 14.) where the subtopic name, description, whether the subtopic is for users with the “preview” permission only, and whether or not the subtopic is visible in the Knowledge Management System will be displayed. The visibility feature allows the administrative user to create an invisible folder with reference material or it allows them to do maintenance to the subtopic their working on without effecting the rest of the Knowledge Management System.

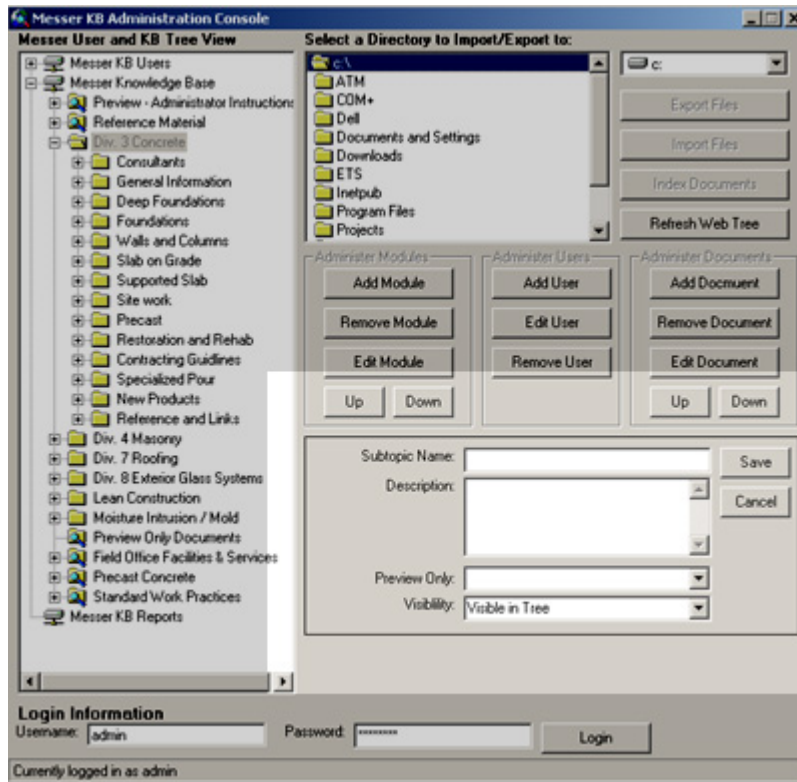


Figure 14. Add New Subtopic

To edit a subtopic, the administrative user selects the subtopic they wish to edit from the list on the left and clicks the “Edit Module” button under the “Administrate Module” section as shown below (See Figure 15.). The user then makes the necessary changes to the subtopic and then clicks the “Save” button.

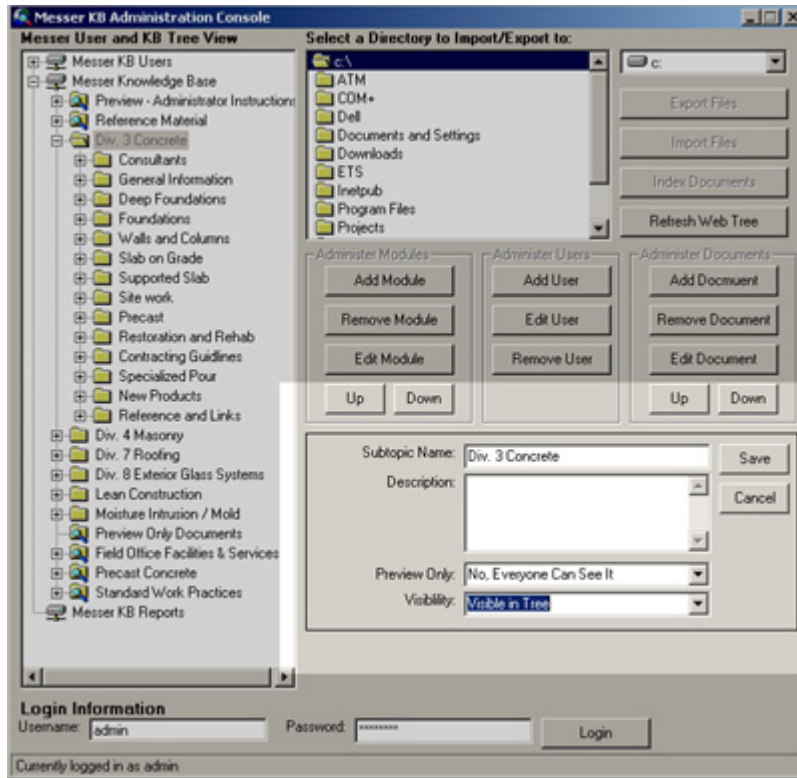


Figure 15. Editing an Existing Subtopic

To change the order of how the subtopics are arranged under a subtopic, the administrative user can select the subtopic he wishes to move and then click the “Up” and “Down” buttons under the “Administrate Subtopics” section and move the order of the selected subtopic. This allows the subtopics to be organized in any order that the administrative user selects.

To remove a subtopic, the administrative user selects the subtopic they wish to remove from the list on the left and clicks the “Remove Module” button under the “Administrate Module” section. A message box will verify that the administrative user wishes to delete the subtopic, and the subtopic will be deleted from the system.

5.2.3 Document Administration

The administrative tool allows users to add, remove, and edit documents that make up the content of the Knowledge Management System. The list of subtopics and documents is displayed on the left hand side of the administrative tool in the tree control as shown below (See Figure 16.).

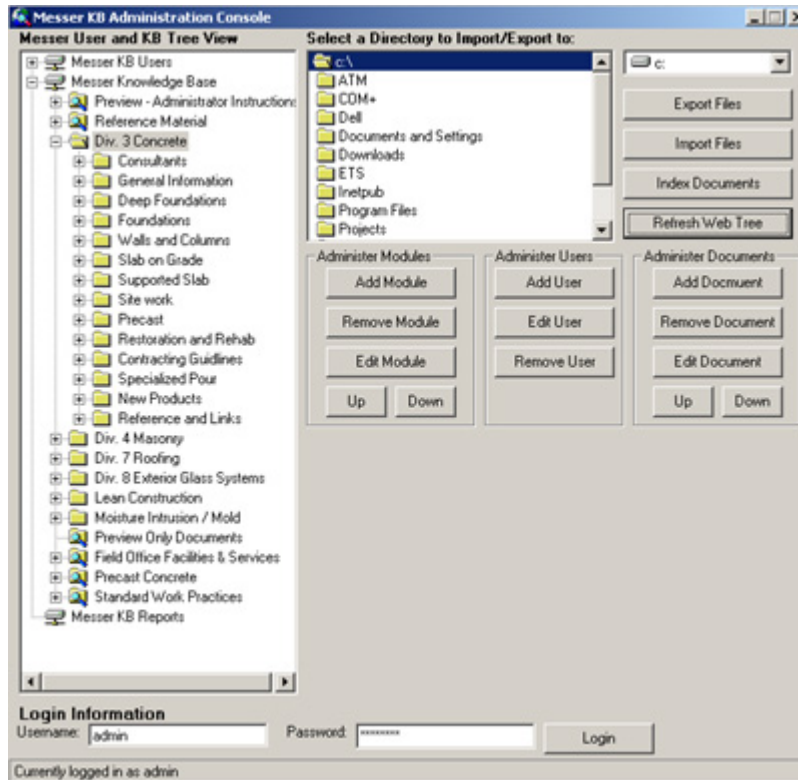


Figure 16. List of Subtopics and Documents

To add a document to the Knowledge Management System, the administrative user selects the subtopic in which they want to add the document and then clicks the “Add Document” button under the “Administrate Documents” section as shown below (See Figure 17.). The user can then enter the document name, which is the name that is displayed in the Knowledge Management System, the file that they wish to add as the document, whether this document is for users with the “Preview” privilege, and the

description of the document. After adding this information, the administrative user can then click the “Save” button to add the document to the Knowledge Management System.

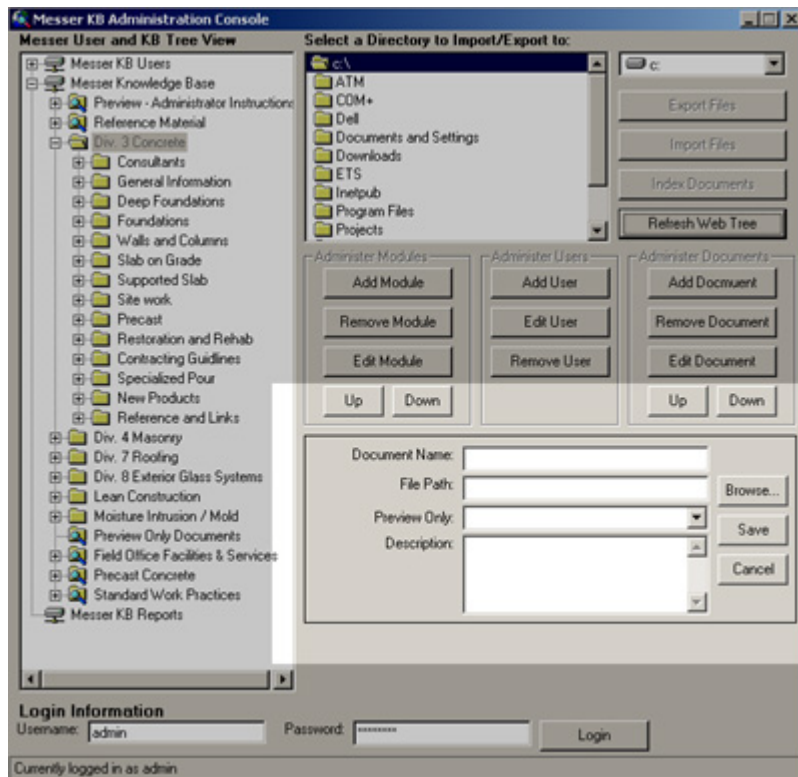


Figure 17. Add New Document

To edit a document, the administrative user selects the document they wish to edit from the list on the left and clicks the “Edit Document” button under the “Administrate Document” section as shown below (See Figure 18.). The administrative user may also select a document for editing by double-clicking the document. The user can then update the document name, description, and whether the document is “Preview” only. To modify the actual document, the administrative user clicks the “Edit Document Contents” button and the native file editor for that document, such as MS Word for an MS Word document, will be launched to edit the contents of the document. Once the document is edited, the user saves his changes in the native file editor and then closes the document.

The user then will be shown a message box asking if they wish to save the changes to the Knowledge Management System as shown below (See Figure 19.). If they select to save the changes, the documents updated content will be loaded to the Knowledge Management System.

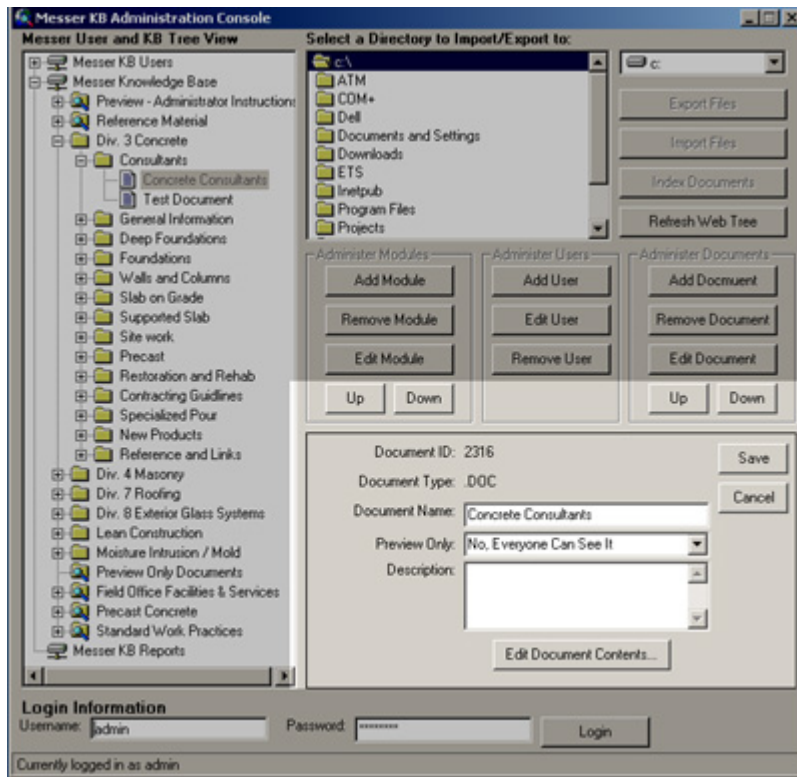


Figure 18. Edit an Existing Document

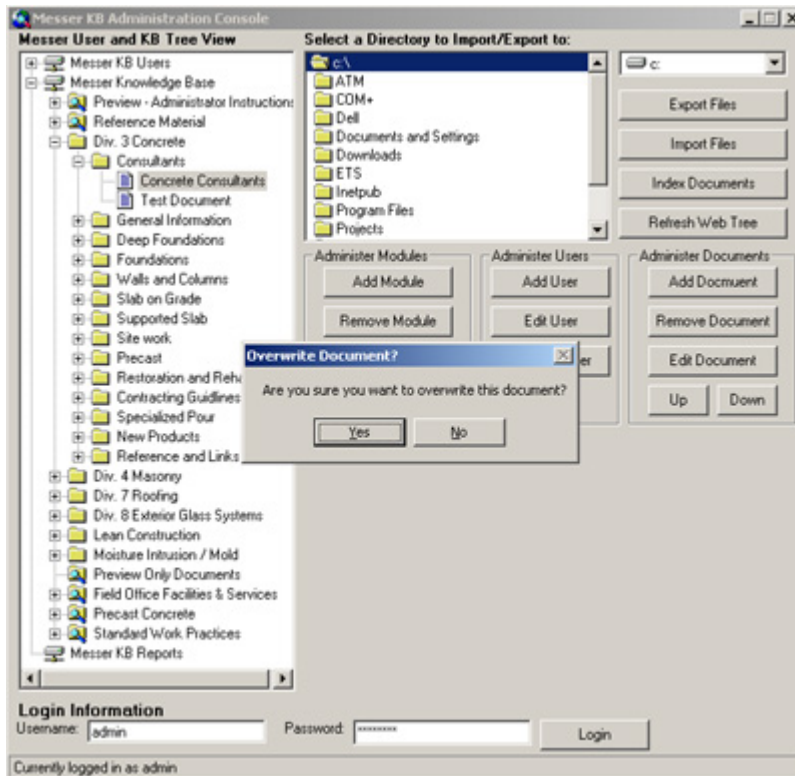


Figure 19. Verify Document Change

To change the order of how the documents are arranged under a subtopic, the administrative user can select the document he wishes to move and then click the “Up” and “Down” buttons under the “Administrate Documents” section and change the order of the selected document. This allows the documents in that particular subtopic to be organized in any order that the administrative user selects.

To remove a document, the administrative user selects the document they wish to remove from the list on the left and clicks the “Remove Document” button under the “Administrate Documents” section. A message box will verify that the administrative user wishes to delete the document, and the document will be deleted from the system.

5.2.4 Exporting Documents

The administrative user can export any portion of the Knowledge Management System by selecting the subtopic the administrative user wishes to export and selecting an empty folder in the export folder selection box located in the upper right hand corner of the Administrative Application and clicking the “Export Files” button as shown below (See Figure 20.). The export function exports the Knowledge Management tree structure creating file system folders for each subtopic and exports the contents of the documents with their associated file extensions.

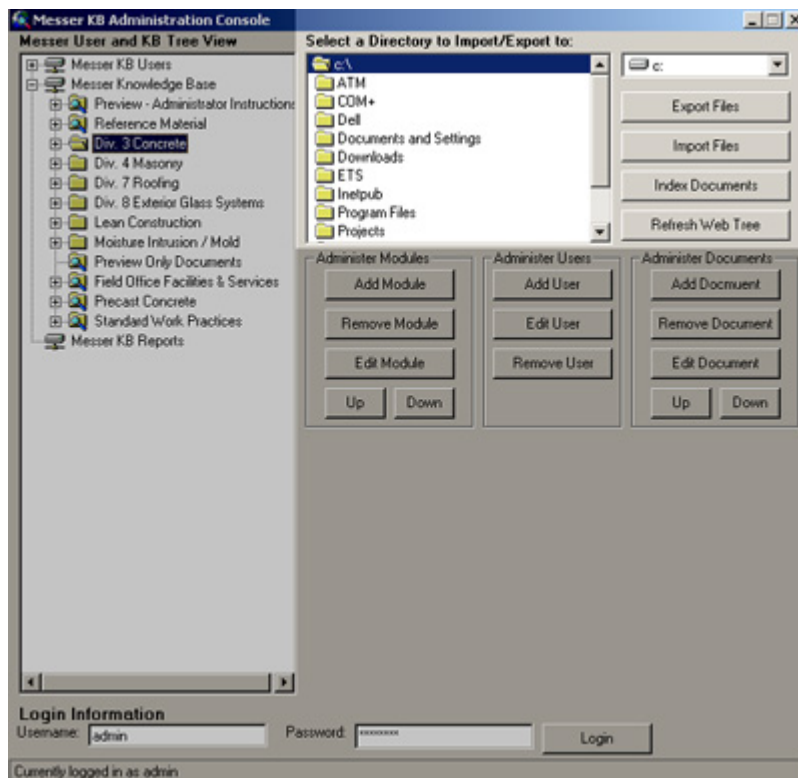


Figure 20. Export Folder Selection Box

5.2.5 Importing Documents

The administrative user can import complete folders to the Knowledge Management System by selecting the subtopic the administrative user wishes to import to and selecting

the folder in the import folder selection box located in the upper right hand corner of the Administrative Application and clicking the “Import Files” button as shown below (See Figure 21.). The import function imports the directory structure of the selected import folder by converting file system folders to subtopics and storing the files located in the directories as documents inside of the Knowledge Management System.

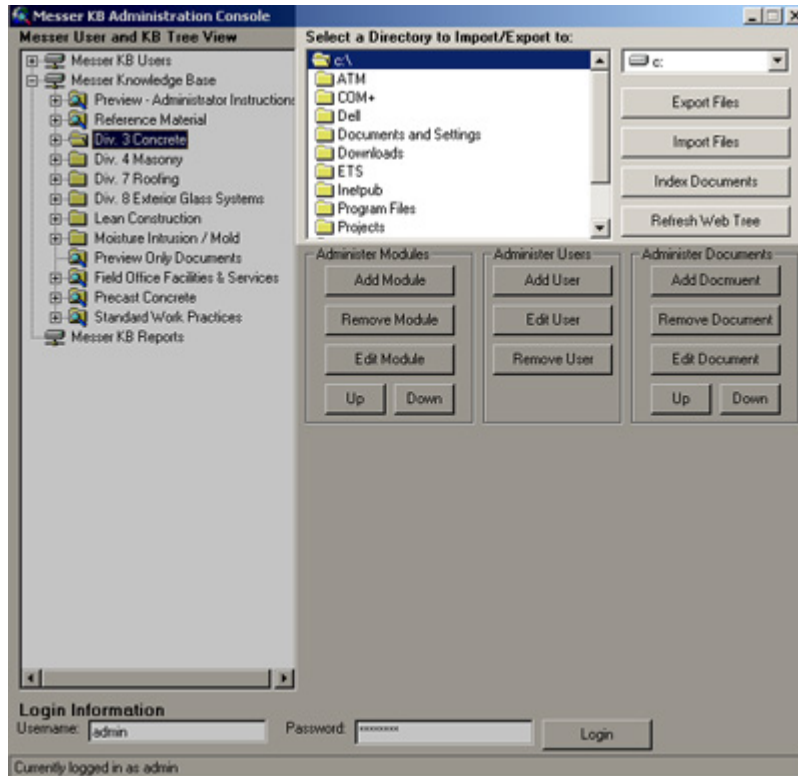


Figure 21. Import Folder Selection Box

5.2.6 Indexing Documents

The administrative user can index the documents that have been imported, added, or edited by clicking on the “Index Documents” button located in the top right portion of the Administrative Application. This feature was added so that indexing could be done after all the changes were made by the administrative user instead of constantly updating the index while each document was added, edited, or imported.

5.2.7 Refreshing the End-User Web Tree Control

The administrative user can refresh the cached web tree control by clicking on the “Refresh Web Tree” button located in the top right of the Administrative Application as shown below (See Figure 22.). This calls an MS Active Server Page on the Knowledge Management System’s web server to update an Application Object’s cached HTML representation of the Knowledge Management System’s hierarchal tree view control. If the administrative user does not use this feature, any subtopic or document added or removed will not be displayed to the end-user unless the web server is restarted or after a 24 hour period of time when the application resets this value automatically.

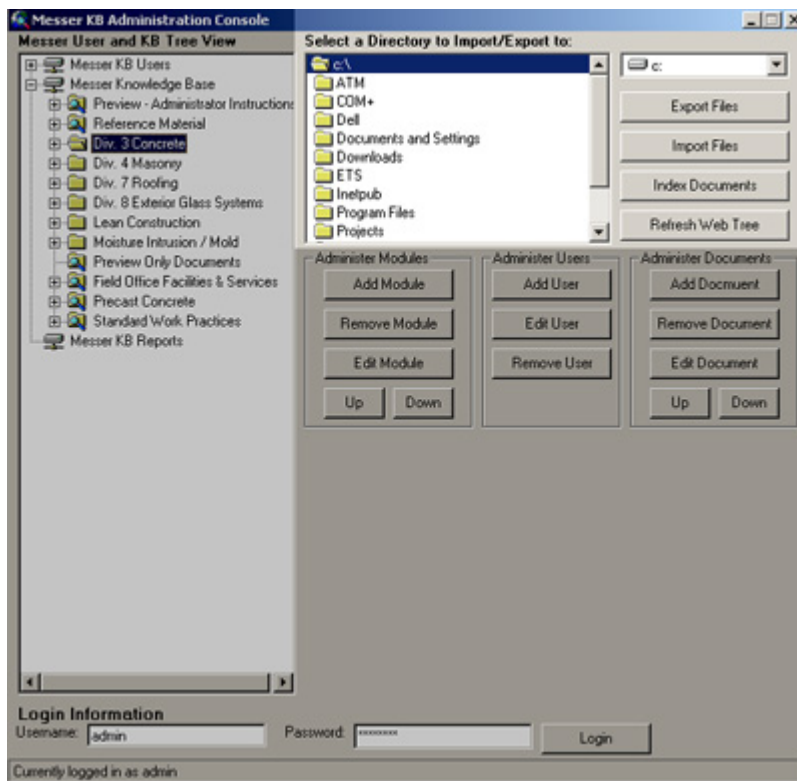


Figure 22. Refresh Web Tree

5.3 The COM+ Component

The COM+ Component is the heart of the Knowledge Management System. It makes up the business logic tier that defines the hierarchical structure and is the main interface for adding, updating, editing, and retrieving documents and subtopics from the Knowledge Management System. It also is the layer that connects the data tier and the user interface together and makes it easy to add additional user interfaces to the Knowledge Management System. It runs as a COM+ component inside of MS Windows 2000 Server Component Services and can be configured using the config.xml file located in the same directory as the COM+ component's DLL. The COM+ component also writes all errors to a log file located in the same directory as the COM+ component's DLL which has proven itself useful in troubleshooting issues and testing within the Knowledge Management System. For more information on the structure of the COM+ component, please see Appendix A. at the end of this paper for an overview of the components objects.

6. Testing Procedures

The Knowledge Management system was extensively tested using a variety of testing methods. During the development phase of the Knowledge Management System, unit testing was used to test each method and property of the core component to ensure that the code performed as anticipated. Conditional compilation statements were used inside the component during development to check for speed bottlenecks inside of the system. Unit testing was performed extensively on the data tier development as well to ensure stored procedures performed as expected. Once the end-user interface was completed and unit tested, the system was initially released to a small user group of ten users for testing

purposes. After testing for a week with the small user group and fixing a couple minor bugs, the Knowledge Management System was then released and used by the entire Messer Construction user base.

7. Conclusions and Recommendations

7.1 Conclusions

This project was designed to solve Messer Construction's need for an organized repository of knowledge that they could easily manage and update. The Knowledge Management System allows Messer Construction to easily create and manage the knowledge of their employees and also empowers their employees with a tool that they can use to find and locate the knowledge to be successful at their job. The project was completed using MS Windows 2000 Server, MS SQL Server 2000 Standard Edition, MS Visual Basic 6.0, MS Visual Interdev 6.0, MS Active Server Pages, MS IIS, HTML, and Transact SQL. The system was completed in a six month time frame, although there are continual enhancements that have been added to the system since its first inception. Since Messer Construction had the software and hardware already in place for other applications, the only costs to them for this system was for the labor involved, which was minimal. The Knowledge Management System has fulfilled all deliverables outlined in the previous section and the system has been tested extensively to ensure that it performs as expected.

7.2 Recommendations

As with all information technology endeavors, there were some problems encountered during the development of the Knowledge Management System. Initially we were using a JavaScript tree control for the end-user application which was successful for a small

amount of documents. When the number of documents increased the end-user application would throw an error about running out of memory on the client's computer. This was resolved by using cascading style sheets to create the tree structure, which has been tested with thousands of documents without any problems.

Another major issue found while developing the application is how MS Office documents is formatted when put in a web browser. Many times the document looks correctly on the administrative users computer, but as soon as its put out on the web the document becomes mangled with overlapping pictures and tables. To fix this issue it has been proposed to update the Knowledge Management component to create a PDF version of the document as well as store the original document for updating at a later point. This will get rid of any distortion created by displaying the document online and give the Knowledge Management System a consistent look and feel.

Appendix A.

Knowledge Management System Object Diagram

User Object

User Methods

- AllUsers() ADODB.Recordset
- AppendDocument(SubTopicID Long, Document Document) Long
- AppendSubTopic(ParentSubTopicID Long, SubTopicName String, Description String, Optional IntroDocument Document) Long
- Authenticate(Username String, Password String) Boolean
- CreateDocumentFromBinary(DocumentName String, Extension String, Description String, BinaryData() Byte) Document
- CreateDocumentFromFile(DocumentName String, Description String, DocumentFilePath String) Document
- CreateUser(Username String, Password String, FirstName String, LastName String, Optional IsAdmin Boolean, Optional EmployeeID String, Optional IsPreviewer Boolean) Long
- GetRecentDocuments(Optional NumberOfDocs Long) ADODB.Recordset
- IndexDocuments(RebuildIndex Boolean, FullPopulation Boolean)
- LoadDocument(DocumentID Long) Document
- LoadSubTopic(SubTopicID Long) SubTopic
- MoveDocumentDown(DocumentID Long)
- MoveDocumentUp(DocumentID Long)
- MoveSubtopicDown(SubtopicID Long)
- MoveSubtopicUp(SubtopicID Long)
- RemoveDocument(DocumentID Long)
- RemoveSubTopic(SubTopicID Long)
- RemoveUser(UserID Long)
- Search(SearchText String, Optional (FirstDocument Long = 1), Optional (LastDocument Long = -1)) Documents
- UpdateUser(UserID Long, Username String, Password String, FirstName String, LastName String, IsAdmin Boolean, EmployeeID String, IsPreviewer Boolean)

User Properties

- EmployeeID String
- FirstName String
- IsAdmin Boolean
- IsPreviewer Boolean
- LastName String
- Password String
- Root Subtopic
- Username String

SubTopics Object

SubTopics Methods

- NewEnum() IUnknown
- Item(Index Long) SubTopic
- Item(SubTopicName String) SubTopic

SubTopics Properties

- Count Long

SubTopic Object

SubTopic Methods

None

SubTopic Properties

- DefaultDocument Document
- DefaultDocumentID Long
- Description String
- Documents Documents
- PreviewOnly Boolean
- SubTopicID Long
- SubTopicName String
- SubTopics SubTopics
- Visible Boolean

Documents Object

Documents Methods

- NewEnum() IUnknown
- Item(Index Long) Document

Documents Properties

- Count Long

Document Object

Document Methods

- SaveToFile(FileName String)

Document Properties

- BinaryData Byte()
- CreatedBy Long
- Description String
- DocumentID Long

- DocumentName String
- Extension String
- FullPath String
- PreviewOnly Boolean
- SubtopicID Long
- UpdatedBy Long

References

1. Sheriff, Paul D. “Designing A .NET Application” April 2002
<http://msdn.microsoft.com/visualc/using/understanding/arch/default.aspx?pull=/library/en-us/dndotnet/html/designnetapp.asp#designnetapp_topic4>
2. MSDN (Microsoft’s Developer Network). “Platform SDK: Indexing Service”
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/indexsrv/html/ixrefint_9sfm.asp>
3. MSDN (Microsoft’s Developer Network). “Creating and Maintaining Databases: Full Text Indexing” <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/createdb/cm_fullad_3bs2.asp>