International Journal of Business and Technology

May 2013

An innovative web application for managing academic documents

Krenare Pireva
University for Business and Technology, krenare.pireva@ubt-uni.net

Petros Kefalas
University of Sheffield in Thessaloniki

Follow this and additional works at: https://knowledgecenter.ubt-uni.net/ijbte

Part of the Computer Sciences Commons

An innovative web application for managing academic documents

Krenare Pireva[1], Petros Kefalas[2]

[1]Faculty of Computer Science and Information System University of Business and Technology
Pristina, Kosovo
[2]The University of Sheffield International Faculty Thessaloniki

Abstract. Archiving versions of academic documents in consideration of green thinking was the motivation to develop an innovative tool that could organize academic documents in a centralized database. This paper presents a new web application, which aims to move towards a paperless University model for managing academic documents that are used within an educational institution, such as proposal courses, syllabuses etc. AcaDocMan,the developed application can be used by academic staff and quality assurance officers of institutions who are able not only to manage their course syllabuses but also to generate different consistent document formats for various purposes.

Keywords— Document Management System; Educational documents, semantic models.

Introduction

In the last decade, there is a growing demand to store and maintain vast amount of documents, large institutions have an emerging demand for managing versions of documents in a right approach and in different types and formats for different purposes. With the rapid growth of technology, document management systems are becoming indispensable tools for solving such problems [1]. A Document Management System (DMS) is usually understood as a computer system that contains software for storing, managing, tracking electronic and digital documents [2].

Nowadays, Universities are facing with difficulties on managing their own academic documents. Such documents includes different version of proposal courses, syllabuses, course descriptions etc. Only saving and archiving documents in an organized hierarchical manner, taking care of naming conventions is not a solution. Even when standards are followed for easier identification and retrieval of documents, the problem remains. People may encounter difficulties to access documents stored in different locations, especially when they want to find something quickly or in worst scenarios, when they want to access specific older versions quickly [3]. The situation becomes even worse when the content of a document should appear in various formats, thus in different files. A change in one document would need to fire a chain reaction of changes in others, which however cannot be achieved manually without errors.

Recently, there are a lot of commercial and non- commercial DMS tools that are offering opportunities to handle and share documents within the institution, by allowing users to access, store and modify data quickly and frequently. On the other hand, none of non-commercial DMS tools offer all functionalities that an institution would need. DMS development is based on general requirements and principles.

As a consequence, the choice of one generic DMS by an institution, without mapping it to the institution's specific requirements, will probably end up in an unused system.

 On the other hand, there exist two alternatives, one is to choose an existing open source DMS and start adapting for one particular institution needs. The other is to start developing a new DMS from scratch, based on the institution's requirements. Currently, web based applications supported by centralized databases, which are designed to store and manage data in centralized comportment, seem appropriate for most of the cases.

The main purpose of this paper is to describe an open source document management web application, which was developed for academic staff and quality assurance officers of Universities, who will be able to manage their course proposal and generate them into different formats, through a web based application.

The paper initially starts with an introduction to document management systems and the University difficulties that are faced while trying to organize their academic documents for quality assurance purposes. Section 2 continues with electronic document management systems and the current tools that are being used from different

39

vendors. Section 3, gives the proposal while Section 4 describes the new AcaDocMan application and it's functions. Finally, we conclude with improvements and future work.

1. Electronic document management systems

An all in one solution system for managing documents is hard to find. There are different existing tools which offer one of the many different DMS components, for instance auditing, capturing, storage, indexing, locating, retrieval, versioning, and metadata, etc [4]. This section will review some of them.

2. Electronic Document Management

EDM [5] system is based on three-tier architecture: (a) electronic documents generation, (b) electronic documents management and (c) electronic documents sharing. It is focused on general paperless systems but with emphasis on their main system. They presented their system as fundamental and realistic during the transition period from paper to paperless documents. EDM claims to be a fundamental and realistic approach during the transition period from paper to paperless documents. In EDM the documents are generate in digital format and are divided in two categories:

☐ academic and

☐ administration

Beyond these the system is categorized further. Administration documents are categorized in:

☐ archival,

☐ communication and

☐ advertisement

Academic documents are categorized furthermore in:

☐ department management,

☐ teaching support,

☐ courseware documents.

In EDM, the digitalized documents are shared in different departments though internet. Also, there are other tools that can be useful for implementing paperless systems. Example, Permarked
[5] is a system that is mainly focused in assignments and their marking annotation. Preoperative Risk Assessment (PRA) [5] is used in hospital environments, in order to avoid incompleteness and storage problems, with support for avoiding redundancy functions and validation checks. iJITin Office [5] manages digital and handwritten documents. The relation between both document types uses metadata annotations for fast searchable results. All of them are great to use but not in our case, since our concern is how we can manage the academic documents in a different versions excluding the assignments and the problems of storages

3. A Model for Document Management in e-Government

Systems Based on Hierarchical Process Folders

RAfEG [6] is an application that includes a DMS system. It is purposed for e-government environments for paperless solution. DMS in e-Government System uses a WebDAV [5], as interface between the system and document management system. The main system architecture is designed to be separated in two categories:

- hierarchical process folders and
- security level.

The process folders can be used in single authority software, as well as in distributed e-government software system. The hierarchical process folder consists of files that belong to a government process as well as all documents processed during process execution.

Security level is subcategorized in two different security levels:

- network security level and
- document security level.

The main advantage of this DMS proposal solution is that, all documents are stored in an organized manner where all changes made to any old document are stored as a new version. All changes of documents are highlighted and also the user information that has participated on those changes is saved, for traceable history process.

Finally, the network security level is crucial during the communication between external authorities, it offers different security levels either for networking or for documentation. Communication between main systems and document security level is very professional and it provides an encryption of documents that are traveling inter different authority systems. However, in our case the first phase of our application doesn't provide any encryption level since the documents that are going to be stored are not so risky, but if we are planning to include also other types of documents with sensitive content, than it is a good approach for the next phase or in the second version of our application.

- Semantic Models for Knowledge Management

Currently, it is a great need to include ontologies, which is semantic models, for making the systems more intelligent [7]. Using search engines can provide desired results for retrieving relevant data within expected time response. However, there are still only few companies that could implement it, since it is a rather new and evolving technology.
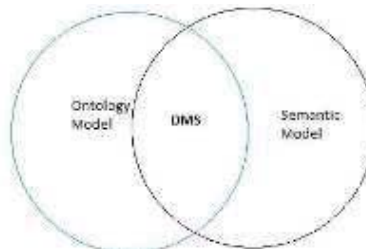

Figure 1. DMS Model for Knowledge Management

EXIP (Executive Information Portal) is one of the tools, that supports a team of strategic business analysts for their daily work (use this model for building semantic web) and also some other systems that are used for classifying relevant documents, within the domain. CHAR [7] and PlanetONTO [7] are two systems which are mentioned as common systems with EXIP and JUSTIC [7] as document classifier systems according to an ontology model.

The advantage of the above listed tools is that the system may evolve the model on basis of classified documents, while the system offers the ability to remove the unused nodes and expand those nodes that are overused.

Documents are classified based on goals and events. They use different machine learning techniques for classifying them. kNN (Nearest Neighbour) and NB (Naïve Bayesian) are two top semantic model classifiers that are mentioned for describing whether the document is relevant to a semantic model node or not. Both of classifiers, kNN and NB use the same logic for classifying the documents in a proper way. They present a document as vector in n-dimensional space, where "n" represents the number of keywords that are part of the document. Through this paper they have shown an experiment that they conducted with both classifiers and during their comparison with each other, where kNN resulted much better then NB. The test set was the Reuters-21578

collection [25] of documents which nowadays is used as a standard test set for research on information retrieval and machine learning.

4. A Three Layer System Architecture for Web-Based Unstructured Data Management

The internet users are suffering from contemporary unstructured and heterogeneous data problems. A web based data management system prototype, called TMUSIC system [8], gives a good example how a system can be developed based on its architecture, where users have the ability to create personalized services with advanced functions, such as cross media queries and automatic recommendations [8].

The aforementioned issues are integrated into three different layers, such as:

- ☐ data gathering,

- ☐ data managing and

- ☐ data servicing.

This system can offer this capability by integrating:

- ☐ data spaces,

- ☐ databases and

- ☐ meta search engines.

TMUCIS offered the retrieved data from the database and also from different search engines, in case that data could not be found within a database. Nevertheless, the ability of creating and storing the users profile may lead to data redundancy, since different users may have same requirements. However, creation of union de put y c la s s ma y de c r e a s e t he s t a t e d problem.

5. The problem in managing documents and the proposed dms

In this section, we will explore a case of an academic institution, namely the University of Sheffield International Faculty/ We will focus on a quality assurance procedure that deal with the design and approval of courses within a curriculum, The process is based on an academic document (course proposal) which is created by a member of staff and approved by a committee. The Faculty used to organize and monitor these proposal courses documents manually.

What normally happens is that a lecturer fills in an application form and sends this to the approval committee. The committee members, after reviewing it, will justify if the particular proposal course is worthy to be accepted as a new course. If not, then the committee will turn back the proposal course to the lecturer with comments for further improvements. The process could have few iterations. After the proposal course has been accepted, the contents of the submitted document (e.g. aims, description, learning outcomes, teaching and assessment methods, weekly content, textbook, references, etc.) will be needed for extracting other documents of various types. For example, syllabuses that are needed for students, course descriptions published on the web, catalogues for promotion, etc. would need to be generated though the main course approval. Through a manual system, the whole process is error prone since document versioning is not easy and consistency between various document types is hard to maintain, once is slight change is needed in the course proposal.

The new innovative web application will be able to save the main academic course proposal documents separately in a centralized database, present them via web interface and generate this documents into different formats as discussed above. The inspiration of this application is to support lecturers and committees on

42

managing their proposal course documents in a more efficient and consistent way. Furthermore, another problem that the web application will solve is consistency across various types of documents.   The system will always preserve all versions of proposal course documents, since the lecturers may change their proposal even when small changes are required (e.g. textbook, weekly content order etc.) that do not need formal approval by the committee. So the quality assurance officers might not always be able to track the changes that are made to that particular course, after accepting it as a course.

6. Acadocman: the proposed system

User Application Interaction

The sequence diagram in Figure 3 describes the communication between the user and the application, starting from the first step which is the "login step". After the users have logged in successfully, they can add a proposal course and save it into database. After the proposal course has been saved, the application offers the ability to show some details of that particular document and generate it into different formats. Finally the user will log out from the application.
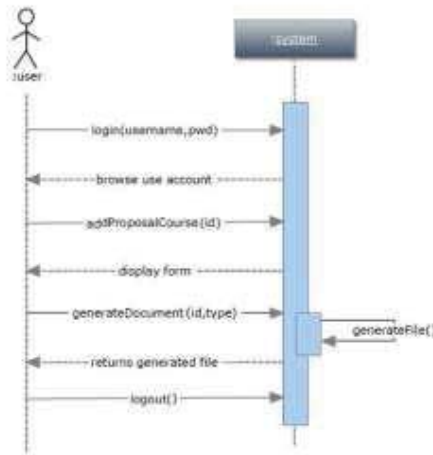

Figure 3. Sequence Diagram: user activity
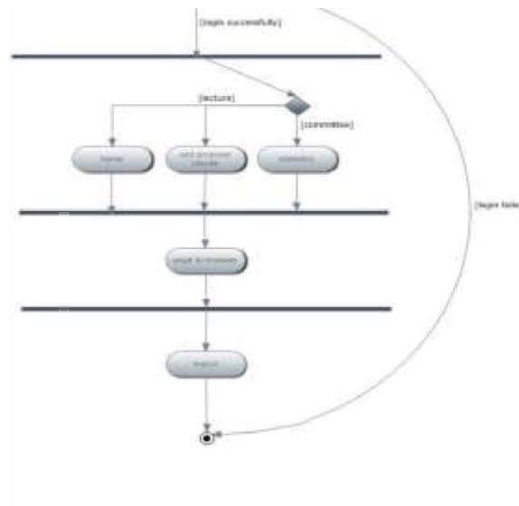
Web Site flow

Figure 4. Web site hierarchy

The UML diagram in figure 4 describes the activity of the web application, step by step. As it is shown in the diagram the activity of the users starts by login in the system. The system will check the privileges of the user and

43

redirect to its personal page. The committees users will have the ability to check the statistics of the papers that are submitted review them and accept the proposal or return them back for any additional changes. All changes will be recognized easily since they will be highlighted with red color.

Whereas, lecturer can add their proposals, modify them and after it has been accepted from the committee they can extract different types of documents in to different format types.
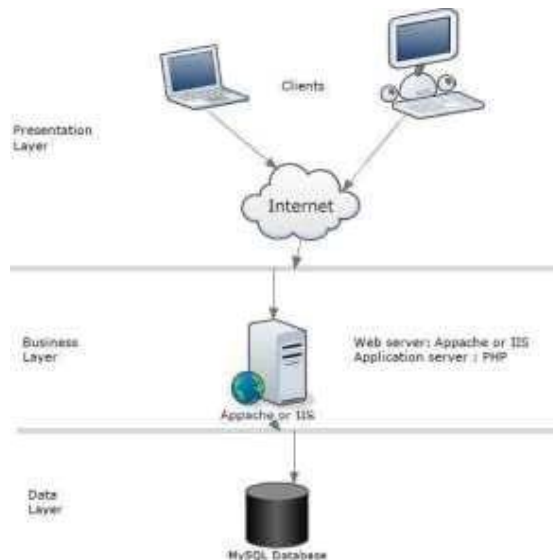
Application architecture

Figure 5. Web application architecture

Our application is based on multilayered architecture [9]. The presentation layer presents the front-end development part of the web application. This layer has direct communication with users, from where users may access the web server through internet cloud, which is illustrated in the business layer. The business logical layer organizes the application since the logic and dynamic of our application is executed in this particular layer.

Finally, arriving to the last layer called data layered; user may retrieve all the data that are stored in the database. The server is responsible for processing all database activities.

Database Model

In figure 6, only a part of the ERD Model is depicted, which describes the interrelationship between entities in a database.

The database contains a number of tables: department, users, proposal course, course, syllabus, taxonomy, methods, aims and learning outcomes, staff members, semester, research doctoral, etc.
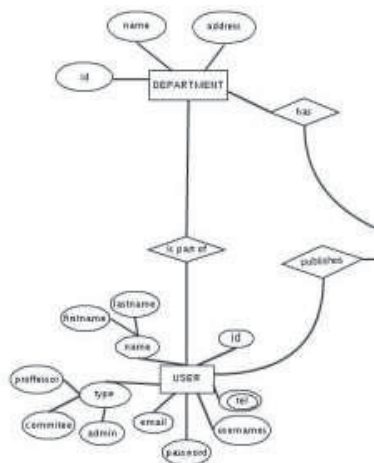
respectively containing their attributes.

Figure 6. Part of the ERD

44

Web Application

Figure 6. Add new proposal

Figure 6 will be displayed when the lecturer will click the "add" link, which is placed in the right panel of web page. It shows how the proposal course could be filled with data.

Figure 7. List of Proposals

Figure 7 shows the home page where all lecture's proposal courses that are retrieved from database are listed. In the left side of each proposal course is the action toolbox which is used in case of deleting, editing or reading the document.

The proposal course could be generated into different formats. From proposal course we can gather:

☐ Syllabus,

☐ Website course promotion description and

☐ Other quality assurance statistics

Syllabus could be generated in .doc and .pdf file, and it gathers from database only fields that are required to be presented in the syllabus template.

Discussion

The process of developing an application for academic staff is quite difficult and complex. Mainly, the application requirements were separated in two categories: the critical requirements and the optional requirements. Almost, all critical requirements like: adding new proposal course, generating the proposal course into different formats, generating a html file that contains only the content that is required for the web-site, generating a syllabus that is needed for students, is completed successfully. Nevertheless, evaluating AcaDocMan overall, it is a safe application and also soon it will be evaluated from users. After the evaluation phase, the tool will be published as an open source application. Publishing the application as an open source application will be an extra chance for further improvements. Everybody can participate, and for sure all volunteer programmers can download the code for free and adapt the code for their own requirements.

45

Conclusion and future work

AcaDocMan is a web-based application that is designed and developed to overcome the problems with organizing academic documents for quality assurance purposes within Universities. Instead of filling in a course application form, sending it back and forth to committee seeking approval, while printing it out several times and saving into archives in the meantime, we have alternatively developed an application which provides a much better solution.

For future work, we plan to include new features for organizing the documents. Such features include:

☐ Integrate a scanning device which will help for digitalizing the documents that already exist as paper documents.

☐ Introduce ontology or a semantic model for classifying relevant documents based on department's terminology and of course for improving the retrieving data performance.

☐ Develop new features for highlighting all new changes thus facilitating versioning of documents.

References

1. Laserfische Institute, "A guide to the benefits, technology and implementation essentials of digital document management solutions", 2007.
2. Matthew Jervis, Masood Masoodian," Design of a Modular Architecture for Integrating Paper and Digital Document Management Systems", 2008.
3. Cotelo-Lema, Jose Antonio, "A document management system and workflow to help at the diagnosis of hypertrophic cardiomyopathy", 2008.
4. Kyrene G. Alberto,Camille M. Abella,Ma. Gracia Corazon E.Sicat,Jasmin D. Niguidula,Jonathan M. Caballero, "Compiling
5. Remote Files: Redefining Electronic Document Management System Infrastructure (CReED)",2009
6. Syed Mushhad, Mustuzhar Gilani, Jamil Ahmed, Muhammad Azeem Abbas "Electronic document management: A paperless university model", 2011
7. Raphael Kunis, Gudula Rünger, and Michael Schwind , "A Model for Document Management in e-Government Systems Based on Hierarchical Process Folders" , 2009
8. Raoul Jarvis, Gregory McArthur, Patricia Rodriguez-Gianolli,Shun Zhou, "Semantic Models for Knowledge Management", 2002
9. Zhiyong Peng, Hui Wang, Yuwei Peng, Bo Xu, Zeqian Huang, "A Three Layer System
10. Architecture for Web-Based Unstructured Data Management",2010
11. Simple Doc, available: http://www.gosu.pl/simpledoc/ , accessed: 22 June 2
12. OpenDocMan, available: http://www.opendocman.com/, accessed: 20 October 2010
13. MarisLabs, available: http://marislabs.org/node/17, accessed: 20 October 2010
14. Philer, available: http://philer.sourceforge.net/, accessed: 20 October 2010
15. Ian Sommerville, "Software Engineering" (5th Edition), Addison- Wesley, 1996
16. C. Larman, "Applying UML and Patterns: An Introduction to Object- Oriented analysis and
17. Design aIterative Development" (3rd Edition), Prentice Hall, 2005
18. Kruchten, Philippe," The Rational Unified Process: An Introduction" (3rd Ed.), 2004
19. Doug Rosenberg, "Applying Use Case Driven Object Modeling with UML: An Annotated eCommerce Example", 2001
20. Mohapatra, Pratap K.J., "Software Engineering: a Lifecycle Approach", 2001
21. Sven Ziemer, "An Architecture for Web Applications", 2002
22. Ryan Stephens, Ronald Plew, "Database Design", 2001
23. Jeff Croft, Ian Lloyd, Dan Rubin, "Pro CSS Techniques (Pro)", 2007 [21] Bear Bibeault,
24. Yehuda Katz, "jQuery in Action",2nd Edition, 2010 [22] John Philips, "Learning PHP and MySQL", 2008
25. Gregory M. Kapfhammer, "Software Testing", Department of Computer Science
26. R.Patton, "Software testing", Sams Publishing 2006
27. Zakaria E., A.Rahmoun, and M.A. Bentaalah, "Using WordNet for Text Categorization", The International Arab Journal of Information Technology, Vol. 5, No. 1, January 2008