Implementation of a Web-based Data Archival Management System

[1]Emmanuel O.C. Mkpojiogu, [2]Gerard Efe Akusu, [3]Azham Hussain,
[4]Wahidah Hashim

[1,3]School of Computing, Universiti Utara Malaysia, 06010 UUM, Sintok, Malaysia
[2]Department of Computer and Information Technology, Veritas University, Abuja, Nigeria
[4]Institute Of Informatics And Computing In Energy. Universiti Tenaga Nasional, Malaysia.
[3]azham.h@uum.edu.my

Abstract

This study reports on the implementation of a web-based data archival management system. Informed and vital decisions are made on the basis of the information that is available to either individuals or organizations. Since information is important, it is needful to achieve and manage them properly for ease of their retrieval when needed. Nonetheless, the sorting, storing, searching, retrieving and moving of files has hitherto been done manually and with papers which is inefficient, tedious, and time-consuming. This research implements a data archival and management system in Veritas University, Abuja which solves the challenges and complexities in the manual system of archiving and managing of information in the institution.

Keywords: Data archival, implementation, management system

1. Introduction

Most of the previous works carried out and systems developed were either strictly on Data Archival or on Document Management. A good number of works relates to this paper. Zhang et al. [16] in their paper, 'web-based medical data archive system', proposed an online distributed archival system for maintaining, sharing, updating, searching, and processing of medical information conveniently and consistently. Medical files typically contain data in different formats, like text description, audio recordings, image files as well as possibly video clips. There is increase in the amount of medical data for each patient and new medical diagnostic technologies are developed and applied in the clinical practice, resulting in more new modalities. Even with the digital archival systems available today, the medical files are typically inconsistent and fragmented, varying from different medical institutions or medical centres to different hospitals. The proposed system not only allows the user to review all the files in different modalities, but also allows the user to do online processing and/or annotation directly on top of the files. The user could save the processed files and/or the original files along with the newly written annotation into his/her local folders, or contribute them to the whole community by uploading the files back to the system, so that this information will be archived in the system. The system is a full spectrum of online communications, processing, and annotation tools, but also provides powerful multimodal search functionalities to the users. In addition, the database is always kept in "live" mode such that information contributed by users is periodically indexed. The architecture of the system consists of the standard 3-tier, browser-server model. The systems may be run with any standard web browser, on any hardware and with any operating system provided there is internet connection. The system uses the Oracle 9i database. The functionalities of this system include online data processing, macro language, online annotation and powerful querying capabilities, and membership control.

In addition, Malik et al. [11] in his work 'web-based archive management and student guidance for final year projects' proposed a system to help final year students and their supervisors organize and manage final year projects and on completion, the project is moved to the repository for archiving. The system used XAMPP so the web server can be deployed on any operating system. Other tools used include Macromedia Dreamweaver, Macromedia Flash and Macromedia Photoshop. The systems was designed and developed according to the requirements of King Abdulaziz University. The final year students usually form groups of two or three and are

assigned a project to work on and a supervisor, hence the system has the functionality of creating groups and adding students to groups, creating projects and assigning groups to created projects. Students need to communicate with their supervisors on matters of the project so the system provides the functionality to send messages to and from the supervisors and students (though not in a social media chat manner) and also external links can be dropped for students which they can refer to for additional guidance on the their projects. Also, milestones can be added in the schedule, information like task name, deliverables and end date etc. and for which department or group of students. The authors stated that 91.9% of final year students read previous students' projects for guidance on how to go about their own projects, hence, this system was built as a management system with an archive feature so while students work on their projects, they can easily refer to past projects from time to time.

The Archivematica Project, carried out by Garderen et al. [4], was aimed at creating an open-source digital preservation system designed for standards-based, long-term access to digital materials. Due to the problems and gaps identified in digital preservation systems, the Archivematica project focused on maintaining accessibility, usability, and authenticity of digital information objects over space, time, and technology. The ISO-OAIS functional model and other digital preservation standards were used in the development of the system. UML Activity Diagrams were used to express functional requirements and the resulting Archivematica system uses a micro-services design pattern to implement these requirements as granular system tasks and provides an integrated suite of free and open-source software tools that allows users to process digital objects from ingest to access. The system's default preservation strategy is to normalize digital objects into preservation formats upon ingest, in order to make best use of the limited time that organizations will have to process and monitor large, diverse collections of digital objects.

In the project work of Timo-Pekka Kemppainen [7], 'data archive project', he attempted to produce a web-based data archival system for Tutkimus-Kehitysja Innovaatiotoiminta (TKI) department at Helsinki Metropolia University of Applied Sciences to aid faster access to different projects in the university and eliminate the need to browse manually through project archives, locate projects and do all estimation calculations by hand. The interface of this system was specially tailored in other to meet requirements and the system only worked in the Metropolia intranet for the purpose of easy accessibility. The tools used to create the system include HTML 5 for the GUI, PHP as the web server based programming language, Eclipse as the text editor, XAMPP to provide an easy to install virtual web server that has Apache and MySQL capabilities and MySQL was used to create the database. Though the project did not result in a working product, it however managed to produce database design to be suitable for Metropolia TKI group. It also laid ground work if such a project were

attempted in the future and a reference for future database development.

Furthermore, Fiona [2] in her research project, 'development of a data management system for students' final year project', stated that the Information Systems department at Universitas Pelita Harapan's School of Computing required a centralized and more practical way of storing and processing the data and grades of final year students. The project was aimed at developing a database management system to store and manage students' final year project data and grades. The system allowed its users to calculate grade averages from multiple examination points, change the weighting-scheme of the final year thesis or project and generate reports. It was developed as an online system with access limited to the university's network, which ensures that data is accessibility at all times. User-friendliness and usability were important aspects that were focused on during development of the system considering the background and skill of the target users and their learning curve(s), so to design the interface they used the eight golden rules by Shneiderman. Also some of Sklar's principles for designing web pages were applied. To design this new system, analysis of the current final year project procedure, data and grade management was conducted. The results of the analysis formed the foundation of the design and development of a database management system – the core support of the data management system. The functional modelling of the system was done using UML and the database supporting the system was created using MySQL. The interface of the system was designed and built on the principles of usability. Due to the minimalist design of the interface, low bandwidth connection did not affect the accessibility of the system and the system was able to run on Internet Explorer 7, Firefox and Google Chrome browsers.

Similarly, Aadamsoo's [1] 'web based project management system', is a management system developed for small companies to help programmers and other users who are not in the same geographical location to create well-functioning web applications for clients, where all needed documentation, coding, testing and bugs related to project are in one place for one specific project. The web based project management system was done as an improvement to an already developed system; the TRAC system. The functionalities of TRAC already included file upload, deleting updated files, user management, with adding new users with data related to it, adding user to specific group, change password, or delete user. However the project aimed to add and implement functionalities such as project management, with adding new project and specify administrator for the project. The tools used included PHP, Python, HTML, SVN (version control), MySQL, Apache, Netbeans, SSH and then UML for the requirements modelling. Since the system is web-based, no desktop installation or updates are required and it could be run on any web browser as well as operating system.

In the work of Meimaris et al. [8], 'an archiving system for managing evolution in the data web', they presented an archive system that is able to handle information as it changes dynamically and is gotten from remote and different sources, taking into consideration change management, provenance, temporality, and querying and implemented the archive based on a data model and query language designed specifically for addressing preservation and evolution management in the data web. The resulting approach is a scalable archiving system that enables multi-versioning, metadata annotations, evolution management from single records and entities to datasets as wholes and common representation layer that offers abstraction. The authors used the data model presented in their other works 'towards a framework for managing evolving information resources on the data web' and 'a framework for managing evolving information resources on the data web', as the basis for the developing of the archive system. Based on this model, a query language was designed and implemented for intuitive request of information from the evolving datasets. A query can be comprised of sub-queries that define particular datasets, versions, changes, temporal ranges, as well as variable bindings for particular changes, records and/or attributes. SPARQL was implemented as the query language, Spring 5 framework for developing the GUI and interactivity of system, and extended SPARQL 1.1 grammar parser and query planner based on Jena ARQ 6 [8].

In the paper 'web based document archiving using time stamp and barcode technologies – a case of the University of Zambia', Mutale and Phiri [10] developed an archival system using C# programming language. The significant features of this system is that it attaches barcodes to documents for quick identification and easy retrieval and also short message service (SMS) and email were integrated into the system to produce and send reminders to the system's users to remind them of schedules for meetings, expiry dates for memorandum of understanding (MoU) documents with other universities and organizations and so on.

Similarly, Joshi [6] presents a system that provides a simple user-friendly interface, for ease of use, where a university faculty can enter, maintain their personal information, manage information on their classes, upload images, send emails to students registered in a class and interact with students on a discussion board for each class, though the system was tailored specifically for the San Diego State University. PHP was used for developing the system and MySQL was used for the database. The features/functionalities of this system include creating accounts, User login and managing of personal page, adding/creation of classes, uploading of images, registration for each class, and discussion board for each class and sending of email to users. One unique thing about this system is that the developer designed and developed it considering users who might have one disability or another. For example flashing effects are avoided or made optional in other not to put users prone to seizures caused by these effects at risk. The system was tested and is compatible with Firefox 3.5 and onwards, Internet Explorer 7 and onwards, Safari 5.0.3 and Google chrome 8.0.552.237 web browsers.

2. Comparison between Proposed System and Related Works

This section used analytical tables to compare and contrast systems in prior works with the proposed system of this study, using a number of functional criteria. Based on the criteria, the tables highlight the gaps created or stated in other works and the gaps the proposed system will possibly fill. The tables were adapted from the work of Malik et al. [11] to fit and serve the purpose of this paper. Boxes with the (**A**) symbol means the system satisfies the criteria, the (**R**) symbol means the systems does not and the (-) symbols means the system's literature did not clearly state whether or not the system satisfies the criteria.

TABLE 1: Analytical Table for Archival Systems

SYSTEM / CRITERION	Web-based Medical Data Archive System [16]	Data archive project [7]	An Archiving System for Managing Evolution in the Data Web [8]	Web-Based Document Archiving Using Time Stamp & Barcode Technologies [10]	The Archivematica Project [4]	Proposed System: ARCHIVO 1.0
Compatible with all standard browsers	A	R	–	–	A	a
System access (within & beyond Institution's intranet)	A	R	–	R	A	A
Signup and Login (Authentication)	A	A	A	A	A	A
Upload and archival of data	A	A	A	A	A	A
Precise quick search	A	A	A	R	R	A
Data retrieval efficiency	A	A	R	A	A	A
Direct transfer of file between users	R	R	R	R	R	A

TABLE 2: Analytical Table for Management Systems

SYSTEM / CRITERION	Web-based Archive Management and Student Guidance System [11]	File Tracking System for Tertiary Institutions [13]	A Data Management System for Students' Final Year Project [2]	Web Based Project Management System [1]	Web Based Content Management System [6]	Proposed System: ARCHIVO 1.0
Compatible with all standard browsers	A	–	A	A	A	A
System access (within & beyond Institution's intranet)	R	–	R	A	R	A
Signup and Login (Authentication)	A	A	A	A	A	A
Upload and archival of data	A	A	R	A	R	A
Precise quick search	A	R	R	R	R	A
Data retrieval efficiency	A	–	R	A	–	A
Direct transfer of file between users	R	R	R	R	R	A

From the reviews and tables above, it can be seen that most systems, be it archival or management, have similar features and functionalities, though for archival systems the editing or manipulation of files is a functionality that is mostly not implemented and for management systems, quick search is an important functionality that is mostly not implemented or does not function properly. However one feature that hardly exists in both systems is the functionality of direct transfer from one user to another. This study aims to develop a single system, integrating features of both systems, thus filling the gaps the individual systems may have. An attempt to improve the quick search function by using references and keywords to produce more precise search results and also to implement the function of direct user to user file transfer through the system was made.

The proposed system will carry the name ARCHIVO and would be referred to as such hence forth. The system is also a prototype system and further improvement will be made to it with each released version and so this prototype version will be version 1.0. Thus this proposed system will be called ARCHIVO 1.0.

3. Materials and Methods

This section points out the tools/materials, steps and methods used to implement the proposed system.

Implementation: In implementing the data archival management system, the following tools and technologies were used:

i. **MySQL:** this is a database management tool that was used to implement the database of this system. It is the world's most popular open source database management tool. It enables the cost-effective delivery of reliable, high-performance and scalable web-based and embedded database applications. It is ideal for both small and large applications. MySQL is very fast, reliable, and easy to use. It supports standard SQL and is platform independent.

ii. **PHP:** Hypertext Pre-processor, is a widely-used, open source scripting language that runs on multiple platforms also making it platform independent and compatible with almost all web servers in use today. It also supports and can manipulate data on a wide range of databases and runs efficiently on the server side and can restrict users from accessing some pages on a website. The implementation of the backend of the system was done using PHP.

iii. **Bootstrap 4:** this is an open source CSS framework that contains CSS and JavaScript-based design templates, typography, forms, buttons, navigations and other interface components. It was used to develop the user interface or front end of the system.

iv. **XAMPP:** this is an open source cross-platform web server package that makes it easy to test and deploy a

web server on any operating system. Due to the nature of PHP being a server side scripting language and cannot be tested locally unless the Apache web server is running locally, a virtual server was required. Hence, XAMPP was used to provide a virtual web server that has Apache and MySQL capabilities.

Sequel to the deficiencies in prior archival systems, there are some specific functionalities that this system aimed to either improve or implement, below are the method(s) used in integrating these features into the system [3].

i. **Direct file transfer**: this function was implemented using the file upload, where the sender uploads the desired file into the system's database server and then the receiver downloads the file from the server. PHP was used to implement this.

ii. **Quick search**: this function was implemented using PHP and MySQL. Whenever a user enters a keyword into the search box, a query is sent and the database is searched to retrieve matching data. References and folders help group related files, and so, if a user cannot recall the name of a specific file, entering the name of the folder or the reference of the file will produce results of all related files. AJAX was used, along with the PHP to create a live search functionality, where the search results will be displayed as you start typing some character in search input box, thus making the search function synchronous.

iii. **Security**: for security, this system implemented user authentication - a security mechanism that is used to restrict unauthorized access to a system. For this system user authentication was implemented using the PHP and MySQL. When a user attempts to login and enters his/her username and password, and submits, these inputs will be verified against the credentials that were stored in the database by the Administrator. If the username and password match, the user is authorized and granted access to the site as either a super or normal user, otherwise the login attempt will be rejected.

Testing: Testing process is used to verify that the requirements of the system are met and function as expected. After the implementation of the system, the system was tested for the following:

i. User authentication functionality: this was tested by attempting to login without having an already created user account. Also an attempt to login using a wrong password and/or email address and lastly using a correct email address and password.

ii. Data archival and retrieval functionalities: these functionalities were tested by attempting to upload files from the computer into the system and then attempting to download them.

iii. Quick search functionality: different keywords were typed into the search boxes, in the different modules of the system to see if the system will produce results that were most related to the typed word.

iv. Data transfer from user to user: this was done by attempting to select a file from the computer and sending it through the system to another user on the system. And also attempting to send files that were already uploaded into the system.

v. The system's compatibility with standard web browsers: this test was done by attempting to launch the system on different web browsers.

vi. System access privileges and limitations to the different types of users: this was tested by creating different accounts with differing user account types using the administrators account and then logging into the different accounts to see if the privileges and limitations for each user are provided as intended.

4.   Implementation and Documentation

This section describes and discusses the implementation and the results of testing the features and functionalities [5][13][15] of the resulting system- ARCHIVO 1.0, at the end of the project period.

Implementation: the system implementation is documented and discussed as follows:

i. **Authentication**: A session on ARCHIVO 1.0 begins when a user logs in by inputting his or her e-mail address and password, provided the user already has an account on the system created by the administrator. After logging in, the user can then perform various operations available based on the user type of the user, i.e., super user or normal user. Users other than the administrator do not have the ability to register/signup on the system, this is to make sure that information on the system is not available to just anybody. In the case where a user forgets his/her password, there is the "Forgot Password" option on the log-in page which will have the user's password e-mailed to his/her e-mail address. In the database the passwords are hashed and appear as random string of characters, this ensures that even when an unauthorized user accesses the database, the passwords to user accounts would not still be available. It also ensures that even if two passwords are the same, they will have two different values in the database.

ii. **User management**: the system provides the user with the ability to manage users. The dashboard presents the user with the "Users" option, which allows the Administrator to input necessary credentials and add users and allows all added users to be viewed. Users can also be deleted from the system and user accounts can be disabled and activated when necessary.

iii. **Department management**: the system provides the ability to add departments, view added departments and delete departments when necessary. The "Department" option is present on the dashboard.

iv. **Upload and download of files**: files can be uploaded to the system when either the "Add to Archive" or "File transfer" options are clicked. A window opens for the user to select the desired file

to be uploaded. Archived and/or transferred files can also be viewed and downloaded at will, when the "View files" option is clicked, either in the "Archive" or "File transfer" tabs on the dashboard.

v. **File Management**: all files on the system can be archived, transferred, viewed and deleted. Files are deleted from the system by clicking on the trashcan icon at the end of each file's row and they are immediately removed for the system's database. These functionalities are available under the "Archive" or "File transfer" tabs on the dashboard.

vi. **Archival of files**: a database was/has been implemented for the system using MySQL, to retain any file that is uploaded to and archived/stored on the system. However, only users in the same department can view files that where archived under that department.

vii. **Quick search**: the system provides search functionality when looking for users, references, departments or files on the system. The system produces results that match the most to what the user types into the search box and at the moment the user is typing. For files, the use of references and folder names are added to aid in more accurate and precise search results. Search boxes are present on all the view options on the system.

viii. **File transfer**: the "File transfer" option on the dashboard allows the user to select a desired file by opening a window on the system and then selecting the user the file should be sent to. The file selected will be uploaded into the database server; the intended user will then see it and download it.

ix. **File editing**: the editing feature/function was not implemented as intended. Though files cannot be edited on the ARCHIVO system, whenever a user wants to edit a file he/she can download the file in their computer, carry out the necessary manipulations and upload the file back into the system.

Other implementations include:

i. ARCHIVO 1.0 was designed and implemented so that it is compatible with standard web browsers. It was tested and verified to be working on browsers like Firefox, Internet Explorer and Google chrome. The system was also tested on different machines like Dell, HP and Compaq laptops and is verified to be working fine.

ii. A Local Area Network was created with the server hosted locally, and the system was verified to be working on any computer that was connected to the network, thus, the system will work fine within the Veritas University Intranet.

iii. Since the server of the system was hosted locally, the system will always be available and accessible within the University's Intranet. The System has a database and so any information stored on the system will be retained for as long as necessary.

iv. ARCHIVO 1.0 was designed and implemented in such a way that access to the system is only available to a single administrator and only users that the administrator added to the system.

v. User type dictates the amount and type of operation/privileges that would be available to the user. A super user will not see the same dashboard menu or have access to the system functions as a normal user. In the same way, the administrator has his own menu which enables him to perform his functions. The Administrator has access to every feature and functionality on the system. A super user has access to the Reference, Archive, File transfer and User functions on the system. A normal user can only view files, view users, view references, transfer files and download files.

5. Test Results and Discussion

As mentioned earlier, during the system's testing, the user authentication worked as intended. Users could not login without having an already created account and the system would not let a user login without correct credentials. The data archival and retrieval functions were tested and verified to be working fine as well as the file transfer function. The system's search function work very well and produces accurate results, the system also runs on standard browsers and lastly user type privileges are well defined and function properly.
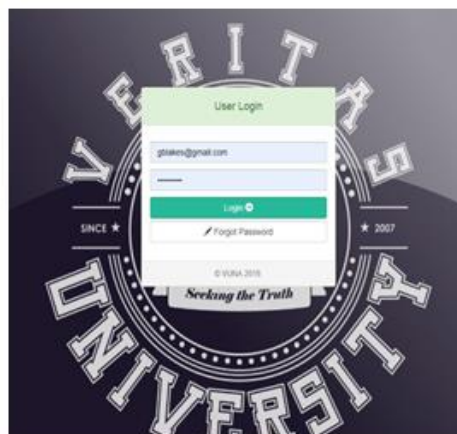
Screenshots of Implemented Requirements
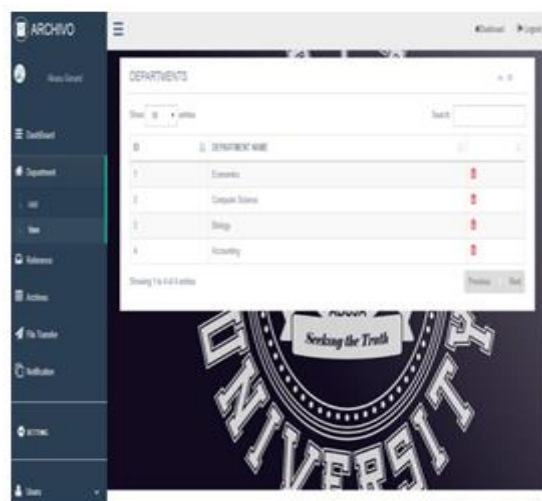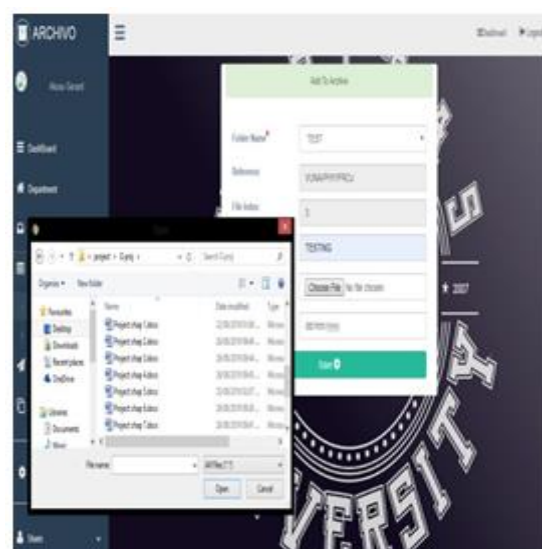

Figure 1: User login page

Figure 2: Add user page

Figure 5: View department page

Figure 3: View user page

Figure 6: File upload on the Add to Archive page
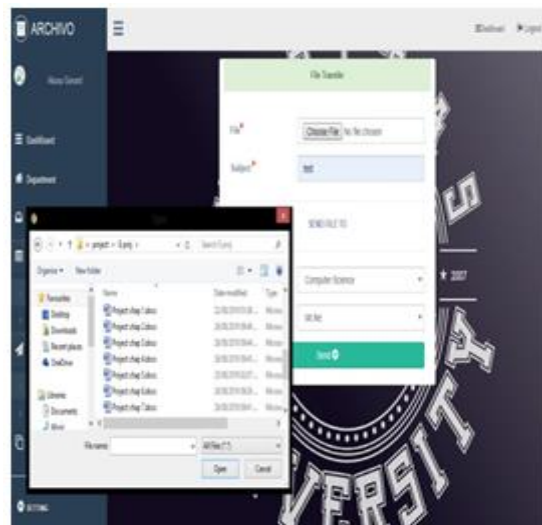

Figure 4: Add department page
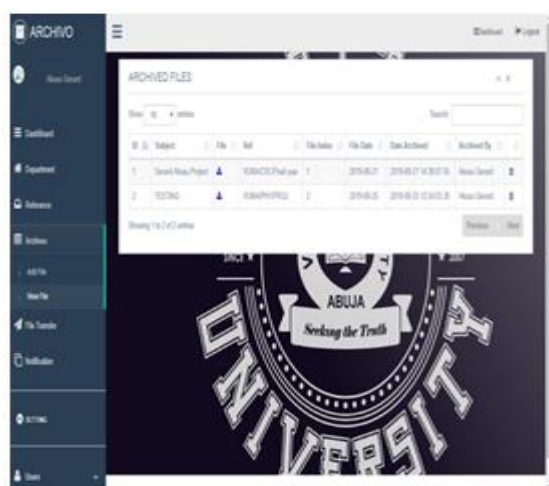

Figure 7: File transfer page

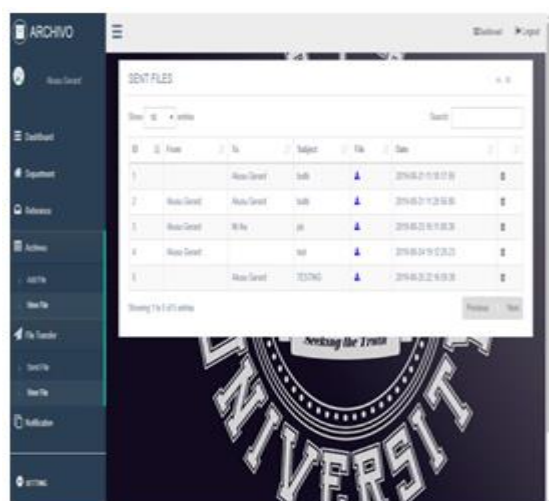Figure 8: Archived files' view page with download and delete buttons

Figure 9: Transferred files' view page with download and delete buttons
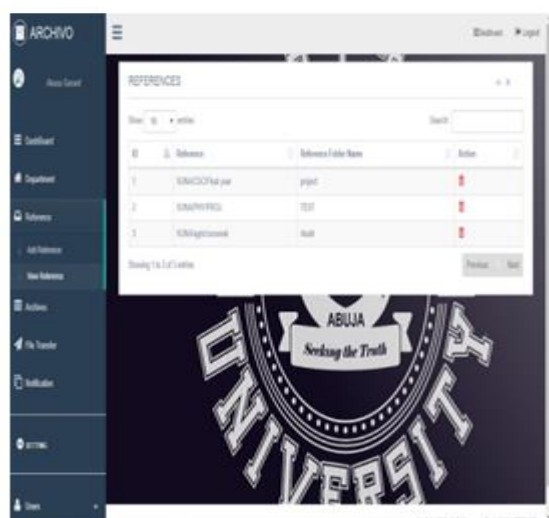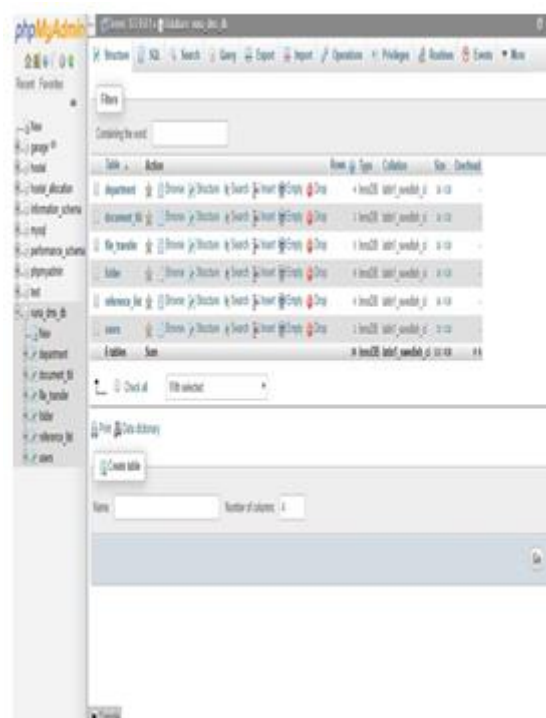

Figure 10: References' view page

Figure 11: System's Database

Documentation: The documentation of ARCHIVO 1.0 is handled in this sub-section. This documentation is intended for the staff of Veritas University, Abuja. It details the different features and functions of the system and instructions on how to use them. The system consists of an "Administrator end" and the user end. The user end consists of two parts; the "Super user end" and the "Normal user end". Every end of the system has a similar dashboard design with a side bar menu, however not all features and functions are available to all users at the end of the system.

User Authentication: To use the ARCHIVO system, you will be required to login; you can only login if you have an account on the system already created by the administrator. To login you will need to input your email address and password in the specified text fields and then click the "Login" button and proceed to use your account. In the case where you cannot recall the password you used, you will need to click the "Forgot Password" button and then input your e-mail address (the e-mail address must correspond to the one used to create your account), your password will then be sent to your email account.

Department: By clicking on the "Department" button on the sidebar menu, a submenu will drop down where you will have the "Add" and "View" options. Clicking on "Add" will take you to a page where you can type in the name of the department you would like to add and then click on the "Save" button to add the department. In the case where the department you are attempting to add has already been added before, the system will alert you with

the "Department already exist" message. Clicking on "View" will take you to a page where you will see a list of every department that has been added and available on the system. At the end of each row is a "Trashcan" icon which can be clicked to delete a department when necessary. However normal users will only be able to view departments and the trash icon will not be clickable on their end.

References: References, for the ARCHIVO system, are used to group files and to search easily, find and identify them. Just as in the real world when a file comes into an office to be managed and stored, it is put into a particular folder along with similar/related files and kept in a particular section; in other to find that file, even if the name of the file cannot be recalled, it is easy to remember and find the section where the file was kept and the folder it was kept in. These section, folder and file name(s) make up the references. Now clicking on the "Reference" button on the sidebar menu will dropdown a submenu displaying "Add Reference" and "View Reference". Clicking on Add Reference will lead you to a page to create folder names and references for files to be stored in. Since ARCHIVO 1.0 was tailored specifically for Veritas University, Abuja, the Organization name - VUNA, is static and cannot be changed, however you will need to specify the department for which you want to create a reference for. Next you will be required to enter "Sub Details" for the reference you are trying to create; sub details are used to narrow down the description of a reference, for example, if you want to store documents for the first semester results final year students of computer science, the reference would be **VUNA/CSC/400L/FIRSTSEM/RESULTS**, where 400L is a sub detail and FIRSTSEM is a sub detail. Beside the sub details text box, there is "Add Details" button; this button can be clicked to add the number of sub details you want on your reference. Lastly to create a reference, you will need to name the folder in which the documents will be stored, following the example above, RESULTS is the folder name. By clicking on View Reference, you will be taken to a page where you will see a list of every reference that is available on the system. The end of each row has a Trashcan icon which can be clicked to delete a reference. Normal users will only be able to use the View Reference and the trash icon will not be clickable on their end.

Archives: Archiving files is one of the main purposes of the system; hence the "Archives" button on the sidebar menu. This button also has a submenu that displays "Add File" and "View File". Clicking on Add File will display a page which will require you to select the folder in which you would like to store the file you want to archive; the folder must have already been created. Next, in the "Subject Matter" text box you will need to give the file a name with which it will be stored in the system with; the subject matter should reflect what the file is about, then, on clicking "Choose file", a window will be

opened for you to select the actual file you want to archive from your computer, then lastly you will need to select the date of the day you are archiving the file. The file you want to archive will then be uploaded into the system's database when you click the "Save" button at the end of the form. On the system, only users in the same department can view files that were archived under that department. On clicking View File, another page will be displayed, showing all the files that have been archived on the system. On each row there is a column with a "Download" icon which when clicked will download the file on that row into your computer. There is also a trashcan icon and then "Send" icon on each row on the list. Clicking the send icon on a particular row will take you to a page for you to select the department and the user you would like to send the file on that row to. The archive functionality is only available to the administrator and super users.

File Transfer: This is another main feature of the ARCHIVO system. On the sidebar there is the "File Transfer" dropdown button, also with "Send File" and "View Files" on its submenu. The Send file button takes you to a page that allows you choose the desired file from your computer and give it an appropriate subject matter, then lets you select the department and user you would like to send the file to. While the View Files on the other hand takes you to a page with a list of every file sent to and from your account. You can also download and delete files here. This feature is available to all ends of the system.

Notification: When a file is sent to any user on the system, the user should be informed that he/she has a message, hence the "Notification" button. If another user sends you a file, the Notification Button will have a marker indicating that you have a new message. Now clicking this button will bring out its own submenu with "New Messages" on it. New Messages will take you to a list of every recent file sent to you that you have not seen yet. This feature is present and functional on all ends of the system.

Users: The last button on the sidebar is "Users". Clicking this button will present "Add Users" and "View Users". A click on Add Users will take you to a page that will require you to select the department of the user you want to create on the system, input the users full name, position as a staff of the university, the e-mail address of the user, the passport of the user and the user type; that is either as a super user or normal user. Then a click on the Save button will create the user account on the system and save the user data in the system's database. View user button will take you to the page of the list of users created on the system. Users can be deleted by clicking the trashcan icon on the row of each user. This feature is only available to the system's administrator.

Other Features and Functions: The main menu page of the system has the "Add Reference", "Add To Archive", "File Transfer", "Add Department" and "Add User" buttons, these functions are all available to the system's administrator but vary for other users as earlier mentioned. On the sidebar is the "Dashboard" button; clicking this button will take you back to the first page of the system after login, that is the main menu page, from any part of the system that you are. Same thing will happen when the "ARCHIVO" sign is clicked. On the top of any page you will see the "Logout" button which you can click to exit your account when needed anytime. Every View page has a search box at the top of their table which produces results of what you are searching for as you type. There is also the "Show Entries" box which you can use to set the amount of rows that will be displayed at a time. The appearance of the system is quite simple and its features are based on the requirements of the end users. The website has been tested and works fine on browsers like Firefox, Google chrome, Internet Explorer.

6. Conclusion, Limitations and Future Work

This study aimed at developing a complete, fully working web-based data archival management system for Veritas University, Abuja. The system was designed primarily to handle the long term storage and management of files in the institution as well as improve on already existing systems by implementing the feature/function of direct file transfer from one user to another. To the best of the researchers' knowledge, as of the time of this project, such a system had not been created. This system proves very useful and putting it to use will help create an environment with greatly reduced use of hardcopy files and a smoother and more efficient work flow as it solves a good number of practical problems. Although this project was developed specifically for Veritas University, Abuja, with little changes and modifications made, it can be used by anyone and in any organization. This system is a viable one, in the sense that it will go a long way in making work environments more convenient, efficient and productive for its users. Because of its importance the researchers strongly recommend the use of the system as its use is not restricted only to schools but also can be used in banks, hospitals and so on. This project achieved the initially set out goals and objectives. Few functions from the gathered requirements, like the online editing of documents through the system and other generally necessary features like new message notifications and change of password, were however not implemented as intended. Therefore, to improve the ARCHIVO 1.0 system, the change of password, notification and file editing functions will be implemented in the next version release. Also, a summative usability study will be carried for benchmarking the system and for further improvement in further releases.

References

[1] Aadamsoo, A. (2010). *Web Based Project Management System* [Scholarly project].Retrieved June 29, 2019 from https://www.theseus.fi/bitstream/handle/10024/16996/Aadamsoo_Anne-Mai.pdf

[2] Fiona, A.C. (2010). Development of a Data Management System For Students' Final Year Projects Case Study: Department of Information Systems [Scholarly project]. Retrieved May 5, 2019 from http://103.23.20.161/index.php/semnasif/article/view/1196

[3] Hussain, A., Mkpojiogu, E.O.C., & Abdullah, I. (2016). Requirements engineering practices in UUMIT centre: an assessment based on the perceptions of in-house software developers. *Journal of Telecommunication, Electronic & Computer Engineering (JTEC)*. 8(8), 27-32.

[4] Garderen, P.V., Jordan, P., Hooten, T., Mumma, C., & McLellan, E. (2012) The Archivematica Project:Meeting Digital Continuity's Technical. Retrieved June 23, 2019, from http://www.unesco.org/new/fileadmin/MULTIMEDIA/HQ/CI/CI/pdf/mow/VC_VaGarderen_et_al_26_Workshop1.pdf

[5] Hussain, A., & Mkpojiogu, E.O.C. (2017). Predicting the perceived worth of software products requirements with customer satisfaction. *Advanced Science Letters.* 23(5), 4269-4273.

[6] Joshi, S.R. (2011). *A Web Based Content Management System* [Scholarly project]. Retrieved July 31, 2019 from http://sdsudspace.calstate.edu/bitstream/handle/10211.10/1113/Joshi_Satya.pdf

[7] Kemppainen, T. (April, 2015). *Data archive project* [Scholarly project]. Retrieved May 5, 2019 from https://www.theseus.fi/bitstream/handle/10024/93069/Kemppainen_Timo Pekka.pdf?sequence=1&isAllowed=y

[8] Meimaris, M., Papastefanatos, G., & Pateritsas, C. (n.d). An Archiving System for Managing Evolution in the Data Web. Retrieved May 5, 2019 from http://ceur-ws.org/Vol 1377/paper2.pdf

[9] Mkpojiogu, E.O.C., & Hashim, N.L. (2017). Quality-based prioritization: An approach for prioritizing software requirements. *Journal of Telecommunication, Electronics & Computer Engineering (JTEC)*, 9 (2-2), 17-21. E-ISSN: 2289-8131.

[10] Mutale, B.M., & Phiri, J. (2016). Web Based Document Archiving Using Time Stamp and Barcode Technologies – A Case of the University of Zambia. *International Journal of Innovative Research in Science, Engineering and Technology,* 5(4) 2319-8753. Retrieved July 5, 2019 from

https://pdfs.semanticscholar.org/076a/668fd87d0dd8e7fa5f5c4710b818d4e697bc.pdf

[11] Malik, S., Al-youbi, S.A., Al-youbi, S.A, Hussein, R., Dweik, G., & Arif, S. (2018), Web-based Archive Management and Student Guidance for Final Year Projects, *Journal of eLearning and Higher Education,* 2018 (2018), Article ID 871144, DOI:10.5171/2018.871144

[12] Mkpojiogu, E.O.C. & Hussain, A. (2017). Can scoring rubrics be used in assessing the performance of students in software requirements engineering education? *Journal of Telecommunication, Electronic & Computer Engineering (JTEC).*9 (2-11), 115-119

[13] Nicholas, A.O., Ambrose, A.A., Adewole, A., & Edeh, E.E. (2014). *Development of a File Tracking System for Tertiary Institutions* [Scholarly project]. Retrieved July 31, 2019 from
http://eprints.covenantuniversity.edu.ng/5567/1/1948.pdf

[14] Yadav, D. (2016). Opportunities and Challenges in Creating Digital Archive and Preservation: An Overview. International Journal of Digital Library Services (IJODLS) 6 (2) ISSN:2250-1142. Retrieved from http://www.ijodls.in/uploads/3/6/0/3/3603729/ijodls727.pdf

[15] Hussain, A., Mkpojiogu, E.O.C., & Kamal, F.M. (2016). The role of requirements in the success or failure of software projects. *International Review of Management and Marketing*, 6(S7), 306-311; ISSN: 2146-4405. http://econjournals.com/index.php/irmm/article/view/3272

[16] Zhang, R., Zhang, M., & Pu, G. (n.d). Web-based Medical Data Archive System. Retrieved July, 2019 from https://pdfs.semanticscholar.org/5a48/4a4ee6594ba96c5a62c2b7ab28abfa207550.pdf