THE UNIVERSITY OF ZAMBIA
SCHOOL OF NATURAL SCIENCES
DEPARTMENT OF COMPUTER STUDIES

University of Zambia web based Document archiving

By

Waddington Mwale
Project Supervisor: Dr. Jackson Phiri

A thesis submitted in partial fulfilment of the requirements for the award of Bachelor's Degree of Computer science.

DECLARATION

I, the undersigned hereby declare that the Web Based Document Archiving System is my own work, that it has not been submitted for any degree or examination in any other university to my knowledge, and that all sources I have used or qouted have been indicated and acknowledged by complete references.

Name : Waddington Mwale

Signature.....

Supervisor : Dr J.Phiri

Signature.....

DEDICATION

In memory of my Father, Henry Maheya Mwale, who imprinted us with the importance of an education. He supported us, cared for us, and made us into strong Men and women we are today. He was the pillar in our family . I can never forget the proud smile on his face when one of us had achieved something. And I know if he were around today that smile would again be present now.

ACKNOWLEGMENTS

First and foremost I would like to thank God who gave me the vision to complete this project

My sincere gratitude to my supervisor, Dr J.Phiri who gave me the project idea about the need to archive files in offices at the University of Zambia, I am truly thankful for his help and support which has been invaluable to me in carrying out my thesis. His expertise on the subject, intellect and constant pushing to get only the best from me, have provided me with a strong foundation with which to build my academic skills as a computer science graduate.

Mr. Martin phiri and Selvas Mwanza computer studies lectures who's advice in developing web applications helped me steer this project in the right direction, a work I will forever cherish.

My classmates computer science 4th year students whose technical opinions made this project the work it is today.

ABSTRACT

Organizations in this digital age deal with a large number of papers in their day-to-day activities. Much of this paper is legal or confidential documentation, and it is a statutory requirement that they be preserved for a pre-specified number of years. Statutory or otherwise, paper documents need to be sorted, searched, accessed, copied, modified, access controlled, audit trailer, Corporations of all sizes have increasingly come to realize that document archiving can add value to the organization far beyond mere speed of retrieval by enabling users to retrieve.[1]

This paper presents the design and implementation of a web-based document archiving and management system customized to serve the needs of an organization. The system is designed to replace a paper-based process that has gradually became unmanageable due to the increasingly large volume of documents that are handled in organisations and the need to distribute the documents across different departments or offices. Features of the new systems include online archiving of documents, download and upload and a secure mechanism for web access of documents ,searching , auditing.

Organizations produce piles of documents, images and other information electronically. The location of this information is a time consuming task. Users tend to file papers and to save documents in folders on their own computers. Nobody knows what information is across the organisation and what information is needed.

TABLE OF FIGURES

Table of Contents

1 CHAPTER ONE: Introduction

1.1 Introduction

This chapter provides the project overview, the principal of the system, the research problem, aims and objectives, scope, research problem and the benefits.

1.2 Project Overview

The operations and functioning of institutions heavily depends on the stored data, that is the collection of information about their research, objectives , meeting minutes, staff records, student records for schools e.t.c . Without data, there can be no functioning of an institution. Yet most of the data collected in institutions is quickly lost or gets mixed up with some other different files.

Other than through summaries posted in subsequent publications, most data are never accessible to anyone other than their original collectors, and many data are eventually lost, even to their collector, over the course of time. Recently, institutions have collectively become more aware of the value of data and of the importance of data preservation [1,2] Data archives serve institutions in a variety of ways e.g archived data enable more transparent science, with better error checking and verification of results.[3]

Archiving also enables data to be re-used for broader meta-analyses and to address new questions. Available data can serve a powerful educational role, both in teaching the statistical and technical aspects of research and to engage students in the process of science.

Data archiving is also a powerful mechanism for data security, providing a mechanism by which data can be saved and recessed by the original authors and others even after hard disk failure or other catastrophes on personal laptops or office computers. The broad spectrum of possible use and re-use of archived data speaks to an array of potential users of the archive, including the original collectors of the data, their collaborators, interested users, reviewers, meta-analysts, students, government agencies etc.[4,5,6]

1.3 The principle of the system

Keeping it simple by making the system easily extensibile, portable (develop once, run on all platform), open source (to suit the nature of this project) and showcase of current trend in web development

1.4 Research Problem

1.4.1 Problem Definition

Increased loss of files and retrieving them manually is leading to loss of vital information in offices at the University of Zambia.

1.4.2 Causes And Extent of The Problem

Loss of files is costing the university workers time to start typing the documents again, and researchers are finding it difficult to retrieve papers written by other researchers.

1.4.3 Extent and Severity of Problem

If files are mixed up in book shelves this this can lead to mistaking a certain document for different purpose, e.g network configurations.

1.4.4 Justification for Research

Not only are the office files being lost and straining to manually retrieve them but also university clubs e.g computer society, unzabeca, unzaseda, students union need to archive their documents for future references, continuation of projects left or completed by the graduating students etc.

1.5 Objectives and Aims

To develop a system that archives files, search mechanisims that retrieves a specific file or list of files , enable users to have login accounts to manage their files, admin account that creates and manage all user accounts, enable users perform general file operations such as upload, delete , download, search, view and archive files.

1.6 Scope

Covers User management, file access control at user and department level, upload of file types specified by root admin, department management, file category management, file types

management, file recovery using recycle bin, backup of files, file access log, sending of email notification, system configuration settings by admin, file Backup and general Searching including Filtering search

1.7 Expected benefits

unzaArchive is a web based file archiving application that uses standards and Open Source technologies hence its open source, features will include file management capabilities including file history, search, auditing, uploading, sharing, email notifications etc. The system is designed to replace a paper-based process that has gradually became unmanageable due to the increasingly large volume of documents and the need to distribute them across different departments or offices in an institution.

1.8 Summary

Introduction was aimed at giving the description about the project, It highlighted the major Aim and Objectives of this project, shed more light on the scope and problem statement under discussion and the benefits of the system.

2 CHAPTER TWO: Literature Review and Related Works

2.1 Introduction

Review of research literature highlights the background information of different related topics under discussion from research papers and related file archiving systems currently in use today.

2.2 Background information

Data is the foundation of science, that is the collection of information about the natural world that is obtained through experiment and observation. Without data, science can't exist. Yet most of the data collected, in particular ecological and evolutionary biology is quickly lost. Other than through summaries posted in subsequent publications, most data are never accessible to anyone other than their original collectors, and many data are eventually lost, even to their collector, over the course of time. Recently, scientists have collectively become more aware of the value of data and of the importance of data preservation [1, 3].

2.2.1 Ecological Archives

The evolution of ecology and journals recently have been adopting polices that require that data from their papers be publicly archived. In particular, ecology and evolutionary biology journals and funding agencies have recently adopted policies that either encourage or require data archiving as part of the publication process, such as the Joint Data Archiving Policy being adopted by several journals. Powerful new data repository, such as Dryad ([1, 3, 5]

2.2.2 Evolution and Archiving

To promote the preservation and fuller use of data, *Evolution* and other key journals in evolution and ecology will soon introduce a new data-archiving policy. This policy will state: *Evolution* requires, as a condition for publication, that data used in the paper should be archived in an appropriate public archive, such as GenBank, TreeBASE, Dryad, the NCEAS Data Repository or as supplementary online material associated with the paper published in Evolution. The data should be given with sufficient details, it allows each result in the published paper to be recreated. Authors may elect to have the data publicly available at time of publication, or, if the

technology of the archive allows, may allow access to the data for a period up to a year after publication, with Exceptions granted at the discretion of the editor, especially for sensitive information such as the location of endangered species[4].

2.2.3 Multimedia And Archiving

With Today's tremendous growth of ICT (Information and Communication Technology) and www(World Wide Web), there has been an explosion in the amount and complexity of digital data being generated, stored, transmitted, analyzed, and accessed. WWW allows us to communicate, share and learn information in a global manner. Much of this information is multimedia in nature, which consists of multimedia elements such as digital images, video, audio, graphics, and text data but all that information is only useful if one can access it efficiently. This does not only mean fast access from a storage management point of view but also means that one should be able to find the desired information without scanning all information manually[3,5].

2.2.4 Content Based Image Retrieval

Content Based Image Retrieval (CBIR) had attracted many researchers in different fields in an effort to automate data analysis and indexing. CBIR is like filter information Process and it is used to provide a high percentage of relevant images in response to the query image. The goal of an image retrieval system is to retrieve a set of images from a collection of images such that this set meets the user's requirements. The user's requirements can be specified in terms of similarity to some other image or a sketch, or in terms of keywords. An image retrieval system provides the user with a way to access, browse and retrieve efficiently and possibly in real time.[6]

2.2.5 GenBank Archiving

Archived data are saved for posterity, the archives also function to preserve data in a useable form for the original authors. If datasets are put into a readily interpretable format while the methods and structure of the data are foremost in the scientists' minds, that data can be used later more easily by those scientists and others. The example of GenBank shows the value of the availability of data for all of these reasons. The modern synthetic use of DNA sequence data would not be possible without the near universal use of GenBank as a public archive. Moreover,

GenBank would not be nearly as complete as it is without the communal decision to archive all DNA sequence data, a decision initially introduced by journals. [3].

2.3 Related Works

2.3.1 Pydio /AjaXplorer

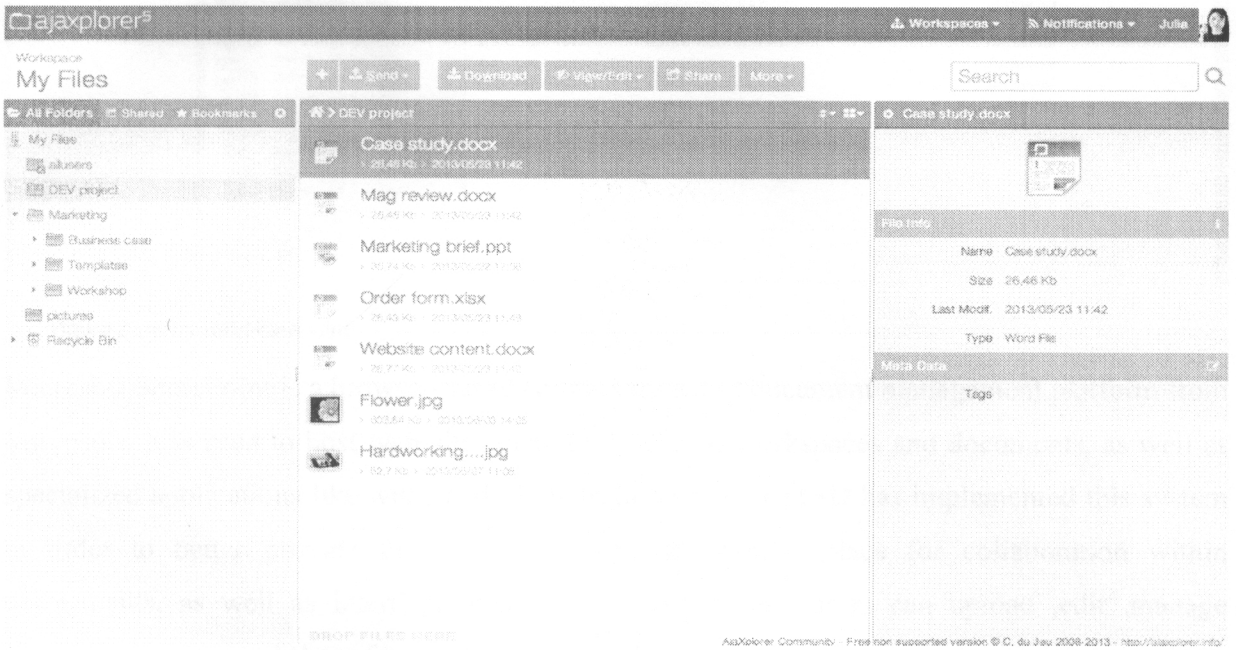


Figure 1 : Pydio document management system

Pydio an open source file archiving system that allows you to instantly turn any server into a powerful file sharing platform. Formerly known as AjaXplorer, users on the system can Gain back control and privacy on their data, reduce costs, and empower them with consumer-grade simplicity allied to enterprise-grade compliance. Pydio deploys on top of your legacy filesystems and instantly connects to existings users directories. Its plugin-oriented architecture makes it ready for all situations.Users can perform operations such as Upload, Search, Delete, profile Update etc.

Learning, deployment and support is from active pydio group members. Users who would want to contribute back, are allowed to hanging around and help other people on the forum, or simply sharing on Twitter, FB, etc.

2.3.2 Sharepoint

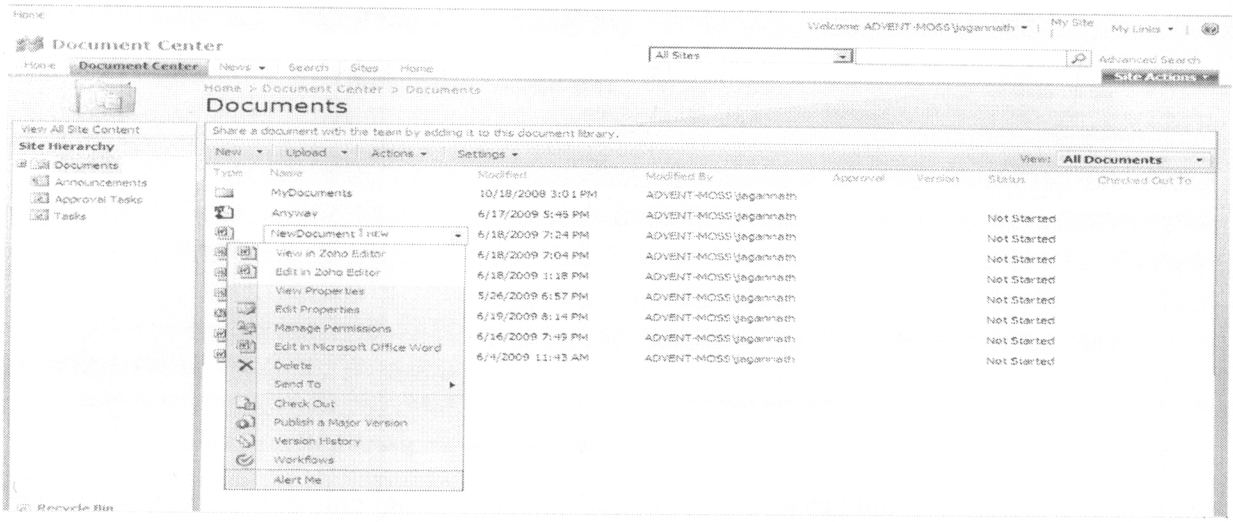


Figure 2 : microsoft Sharepoint

Microsoft SharePoint is a browser-based collaboration and document management platform from Microsoft. It is used to host web sites that access shared workspaces and documents, as well as specialized applications like wikis and blogs from a browser.ITSD has implemented this system in order to better provide the UNCW community with a place for collaboration within departments, as well as interdepartmentally across campus. users can upload ,edit ,manage permissions, view and share files and information.

2.3.3 Moodle

An e-learning system is a system that provides services that are necessary for handling all aspects of a course through a single, intuitive and consistent web interface. Such services are, for example: [16]course content management, [17] synchronous and asynchronous communication, [18] the uploading of content, [19] the return of students' work, [20] peer assessment, [21]student administration, [22] the collection and organization of students' grades, [23] online questionnaires, [24] online quizzes, [25] tracking tools, etc. With the advent of Web 2.0 technologies and services (like wikis, blogs, RSS, 3D virtual learning spaces, etc) e-learning systems will provide services that enable students to shift from passive to active learners where they can actively participate in the on-line learning process. E-learning environments that provide access to synchronous and asynchronous learning resources and activities are going to continue growing [16].

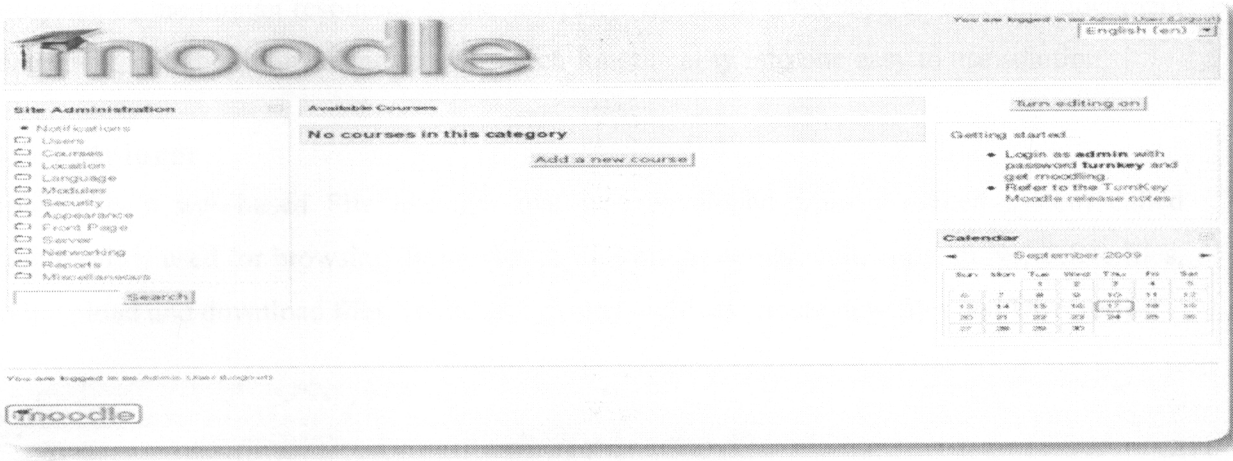


Figure 3 : moodle ,learning management system

Moodle is an online learning management system, where users can build online groups complete with digital assignments, readings, documents, and discussions. Moodle can be used on Macs and PCs and is 100% online, so it can be accessed from any computer at any time[12].

2.3.4 OpenKM

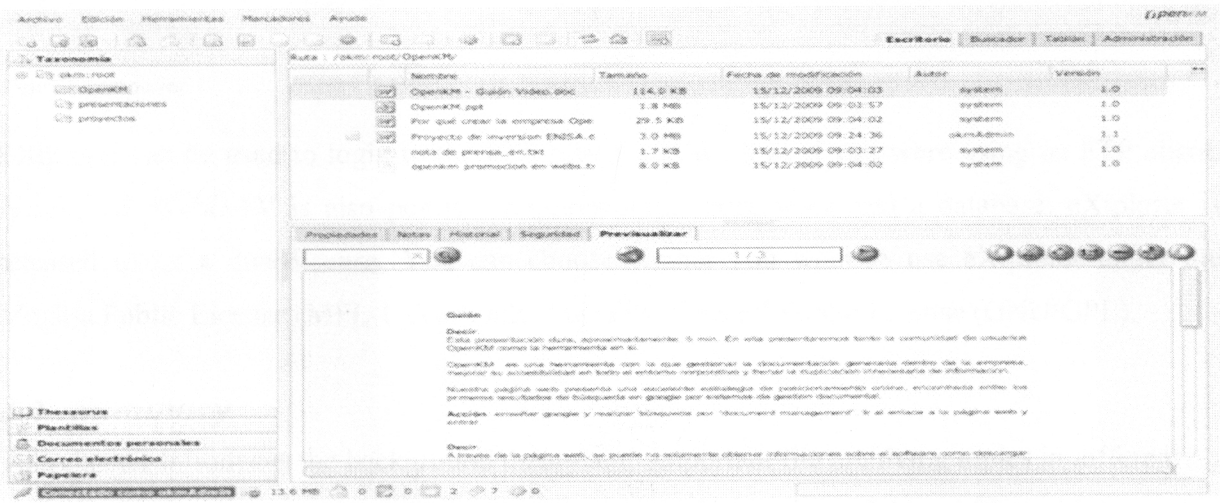


Figure 4 :OpenKM

OpenKM is a web based document management application that uses standards and Open Source technologies. OpenKM provides full document management capabilities including version control and file history, metadata, scanning, workflow, search, and more. It also allows the social activities around content to be used to connect people to other people, information to information, and people to information; helping to manage, more efficiently, the collective

intelligence of the human resources of the company. OpenKM integrates all essential document management, collaboration and advanced search functionality into one easy to use solution

2.3.5 Extplorer

eXtplorer is a web-based File manager that was developed from a related system called quixplorer. It is used for browsing directories & files on the server, edit, copy, move, delete files, search, upload and download files, create and extract archives, create new files and Directories.

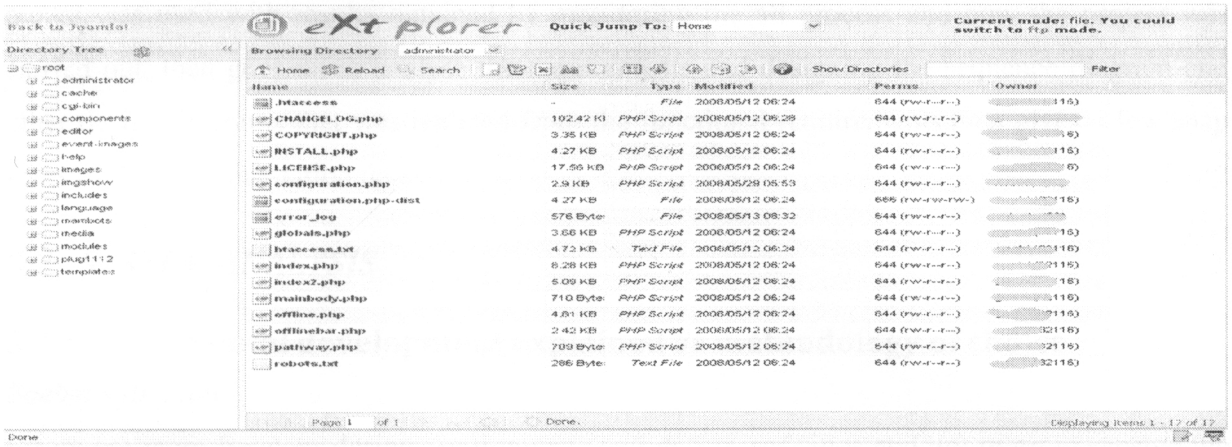


Figure 5 : extplorer

EXtplorer can be used to login to the FTP server and work as if you were using an FTP client. Access via WebDAV is also possible requires some extra work and a database. eXtplorer is released under a dual-license, You can choose wether you want to use eXtplorer under the Mozilla Public License (MPL 1.1) or under the GNU General Public License (GNU/GPL).

2.4 Summary

This chapter introduces the background theory and important file archiving topics in different fields such as genbank, multimedia ecology then expanded on the web based file archiving systems that are currently in used today such as moodle, sharepoint etc.

3 CHAPTER THREE: System Analysis Design and Implementation

3.1 Introduction

This chapter gives the overall system analysis, the design and implementation, using incremental development model by stating Boehm's first law the goes on to explain why the incremental model was used with results obtained by monitoring the all process and how the project was carried out, then goes on to analyse the requirements and designing the sequence diagram and other design diagrams that was realized from the functional requirements,then gives a few snap shots of the implementation details

3.2 SYSTEM ANALYSIS

3.2.1 Incremental development explained as methodology of choice

Boehm's first law

Errors are more frequent during requirements and design activities and are more expensive the later they are removed. Requirements and design errors do outnumber code errors. However cost per defect stays flat from testing through maintenance. The cost per defect metric penalizes quality and achieves lowest values for the buggiest software. For zero defect software the cost per defect is infinity since testing is still necessary. Defect removal cost per function point is the best choice for 1 quality economic analysis. The reason cost per defect seems to rise is because of fixed costs Figure6 .[28]

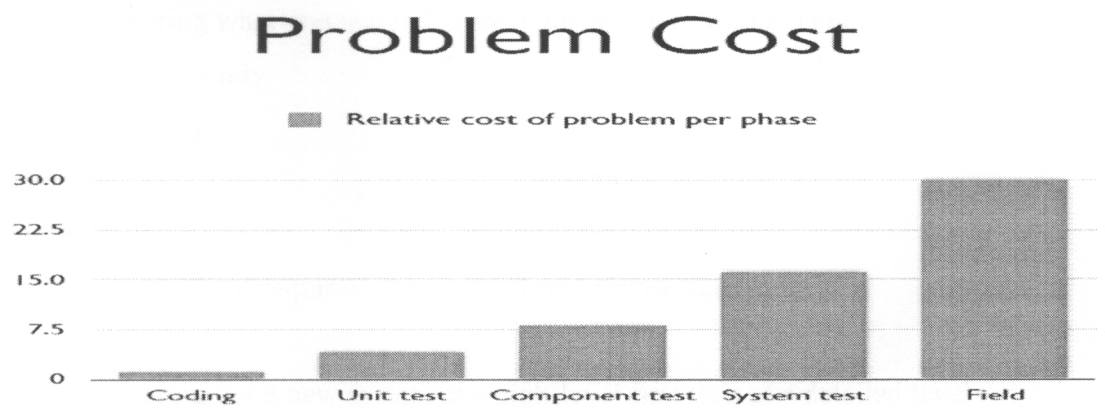


Figure 6 : cost vs per phase

Methodology analysis

Incremental System Development is the methodology that was used to develop Web based Document Archiving. The incremental model recognizes that software development steps are not discrete. Instead, Increment 1 was improved and functionality added until it became increment 2, which became increment n, and so on. These increments were not the versions released but were merely staged compilations of the developing system at a new level of functionality or completeness.

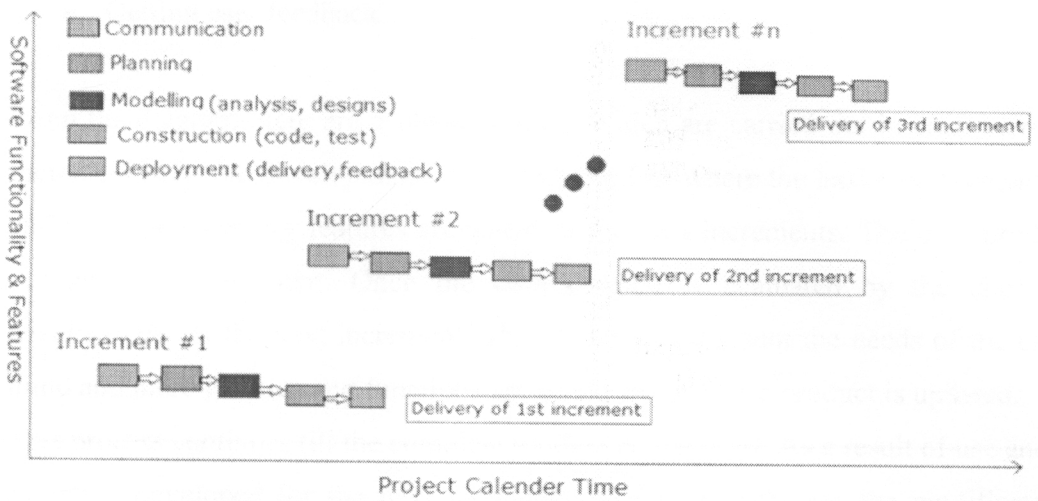


Figure 7 : software functionality features vs project calender time

Increment Tasks

Communication

The first task in increment 1 to was to communicate with users discussing what they really want

Planning

Involved gathering what was needed to solve the problem which included

- feasibility study
- project steps

Modelling

Analysis

Determining details of requirements in the form of a use case model

Design

Drawing blueprints for a new system at a high-level first then at a detailed level

Construction

The actual coding of the model into a software package using php ,sql ,javascript,css,html uses one or more programming languages

Also involved testing the system whenever changes are made and more functionality added

Deployment

Setting up the system and which involved:

- Database
- Accessing on different hosts on a LAN
- Getting user feedback

From the diagram there are 5 phases (tasks) Which are carried out in each increment. The first increment involved development of the core product where the basic requirements are addressed and the supplementary features are added in the next increments. The core product is used and evaluated by the client. Once the core product is evaluated by the client there is plan development for the next increment. Thus in every increment the needs of the client are kept in mind and more features and functions are added and the core product is updated.

This process continues till the complete product is produced. As a result of use and/or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to better meet the needs of the customer and the delivery of additional features and functionality. This process is repeated following the delivery of each increment, until the complete product is produced. The incremental process model focuses on the delivery of an operational product with each increment. Early increments are stripped-down versions of the final product, but they do provide capability that serves the user and also provide a platform for evaluation by the user.

As a major system was nearing completion, builds were scheduled frequently to meet the clients new requirements, It was intended to deliver an operational-quality system at each increment stage, but it did not yet complete the functional specification. One of the biggest advantages of the incremental model when developing web based document archiving was

- the flexibility it had to respond to critical specification changes as development progresses.
- Analysis and development could tackle smaller chunks of complexity.

- The system enabled learning from a new system's development process, and any model that allowed to incorporate this learning into the product was advantageous.

3.2.2 User Case Diagram

Use case scenario

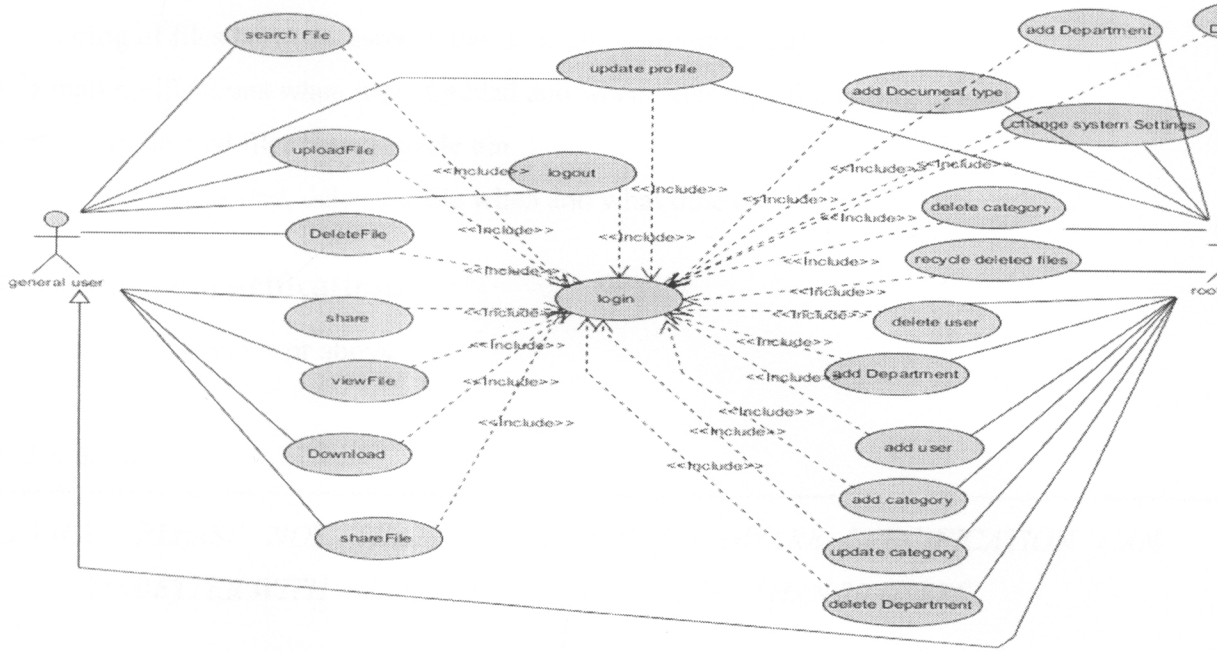


Figure 8 :use case diagram

3.2.3 Functional Requirements

General users

- User should be able to download files
- Searching files by file type, file name, file number, file description, file comment, file author/uploader, department, filtering department, filtering author/uploader name, filtering file category
- Sharing files of with users and departments
- Deleting files and archiving them
- Downloading files

Admin

- Adding, deleting, updating and viewing system users
- Adding, deleting, updating and viewing file categories
- Adding, deleting, updating and viewing department
- Uploading files
- Searching
- Sharing of files between users in the same and different departments
- Email notifications when user is added and when sharing a file during upload
- Recover deleted files from recycle bin
- Audit who uploaded, deleted a file when and what time etc

3.2.4 Hardware specifications

- Processor: Corei3 + or above
- Ram :2GB +
- Hardrive 250

NOTICE: *PLEASE NOT THAT THE ABOVE HARDWARE SPECIFICATION CAN PERFORM BETTER WITH AN IN HOUSE LOAD OF LESS THAN 20 USERS.*

3.2.5 Software specifications

- Operating system : Linux, Windows XP, Windows Vista, Windows 7 or higher
- Browser : Any but better performance with Mozilla Firefox and Chrome
- Database: Mysql 5.0.1 or higher
- Webserver:Apache 2

3.2.6 Non Functional Requirements

Performance/Time:

Processing Time: file upload time taken should be within 5 seconds. Since File size can be various, the processing upload is different. A success message will be shown for upload success or failure.

Response time: This system need a much quicker response time than typical web application.

Usability

Usability is a crucial point in the system. As most users don't have experience of system like this, users are expecting to use the system in a way like a traditional Windows File Browser. The system should simulate the traditional operating system and File Browser.

Security

- All input need to be encoded and validated to prevent SQL injection.
- Users in department groups can assign each other different access rights on files which include read ,write ,admin or forbid be given access can be configured .For example, a user in computer studies department can assign Admin Right to a file to a user in engineering department group.
- Users can update their passwords frequently for security reasons
- User passwords are alphanumeric and must be more than 8 characters
- No user can login in without a password
- All passwords are encrypted and then stored in the database

Portability

The system needs to be portable on all major platforms. This system is restricted to Mysql database, but can execute on any web server, and operating system.

3.2.7 Development tools and software used

3.2.7.1 Programming language

PHP was selected as the server-side language to use. PHP is a widely used scripting language because of its robustness and flexibility. PHP code is embedded in HTML to provide dynamic content. PHP is supported by the two main web servers, namely, Apache and IIS. It is also supported by the Linux Operating System. The PHP framework contains all code that a programmer might need making development with PHP easy. The programmer doesn"t have to reinvent the wheel. It is because of these good qualities and the fact that most Internet Service Providers offer LAMP (Linux Apache MySQL PHP) hosting that PHP was chosen. Options like ASP.NET were avoided because of their platform dependence and the fact that only a handful of ISPs offer Windows servers for hosting. Another reason that played an important role in the

choice of PHP is that development time of the system would be short due to the flexibility of PHP. Finally, PHP is an Open Source Technology offered under the GNU Public License. Therefore, it is easy to acquire and it is widely supported by the Open Source Community. Jquery was used for validating forms , Cascading Style Sheets (CSS) was used for the aesthetics of the website.

3.2.7.2 DBMS

MySQL was the DBMS of choice because of how easily it bonds with PHP. It also has much strength, some of which include robustness and the ability to store huge volumes of data. MySQL also handle high traffic rates well. This is the reason why it is popular with web application programmers. To top it all up, MySQL is, just like PHP, an Open Source product that is provided under the GPL (GNU Public License).

3.2.7.3 Smarty Templating Engine

When it comes to more complicated web-based projects such as web portals, Document archives, e-commerce sites, ERP systems, and so on with the market being so crowded with web projects, in order for your web application to stand a chance, you need to have a great layout which is appropriate for the type of project you are building and of course, stable code tested in all situations. That's why web applications developers specialize either in designing great layouts (web designers), or in programming very well (web programmers).

Programmers have different ways of coding applications and if they use only PHP without a template engine, the designer will need to be familiar with every programmer's way of arranging the code, naming variables, and so on. Also, the programmer will need to understand the designer's templates and generate the code to print HTML code from PHP where the designer wants it.

Let's take the following example:

```
for ( $col = 1; $col < $i; $col++ ) {
    print "<tr> <td> $procesid[$col]</td> <td>$data[$col] </td> <td>$value[$col]</td>
</tr>"; }
```

This is PHP code that prints the values of three arrays in a table. Even a simple example like this one is hard to read and to arrange in a web page exactly as desired. With Smarty, the PHP syntax doesn't contain *print*. Instead, the programmer passes these arrays to the designer by assigning

them to Smarty templates. Then it's the designer's job to make them look good in the web page without worrying about the PHP code. This is one big benefit about using Smarty.

Separating these layers in project development cycles allows you to achieve rapid application development with project maintainability in mind. In the three-tier architecture, the interaction between layers is shown in figure 9 below.

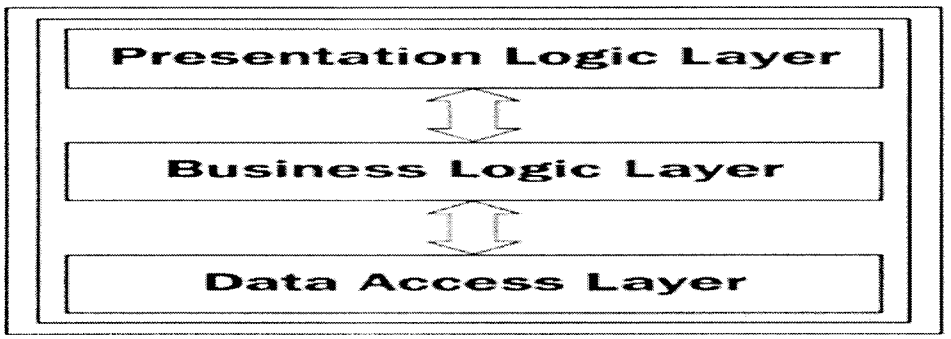


Figure 9 : 3tier architecture

Data Access Layer we find both the data and the ways to extract the data that we want to show to the user. The Data Access Layer may contain:

- SQL language used to extract data from the database
- Files that store data and PHP functions (or other languages) that parse the files
- Data acquisition software (for example, a thermometer on the parallel port)

Now that we've extracted the data, we need to manipulate it in order to get the results we need to display.

Business Logic Layer – Performs the following

- Data validation based on the business plan (for example, view only files)
- Data manipulation according to the business plan (for example, who views what document, access logs for auditing, and so on)

Presentation Logic Layer is where the web page layout how data from the Business Logic Layer is arranged in the web page is defined. This is done using:

- Web page templates
- Text/CSS styles
- Images, banners, and menu styles

Without a templating engine at the Presentation Logic Layer, we find HTML and PHP creating the layout with pure HTML and generating HTML code from PHP. In this case,

we cannot divide the Presentation Logic and Business Logic layers into two separate layers, making the work of designers and programmers very difficult for complex software projects. That's where Smarty comes in.

3.2.7.4 Data Tables Plugin

DataTables is a plug-in for the jQuery Javascript library. It is a highly flexible tool, based upon the foundations of progressive enhancement, and adds advanced interaction controls to any HTML table, interactions involved includes searching columns, retrieving specified number of rows.

Show 10 entries Search:

Name	Position	Office	Age	Start date	Salary
Airi Satou	Accountant	Tokyo	33	2008/11/28	$162,700
Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09	$1,200,000
Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12	$86,000
Bradley Greer	Software Engineer	London	41	2012/10/13	$132,000
Brenden Wagner	Software Engineer	San Francisco	28	2011/06/07	$206,850
Brielle Williamson	Integration Specialist	New York	61	2012/12/02	$372,000
Bruno Nash	Software Engineer	London	38	2011/05/03	$163,500
Caesar Vance	Pre-Sales Support	New York	21	2011/12/12	$106,450
Cara Stevens	Sales Assistant	New York	46	2011/12/06	$145,600
Cedric Kelly	Senior Javascript Developer	Edinburgh	22	2012/03/29	$433,060
Name	Position	Office	Age	Start date	Salary

Showing 1 to 10 of 57 entries Previous 1 2 3 4 5 6 Next

Figure 10 : datatables plugin

3.2.7.5 Operating System and Servers

The system was implemented on Linux. The Lamp package was used for implementing and running the web based file archiving system. Phpmyadmin provided the interface to managing Mysql database.

3.2.7.6 Jquery validation plugin

Jquery validation plugin was customized and used to validate all forms in the system reducing the time taken to validate user input on server side.

3.2.7.7 Tweeter Bootstrap

Twitter's Bootstrap is an excellent set of carefully crafted user interface elements, layouts, and javascript tools, freely available to use in web design project. The Bootstrap stylesheet provides an easy-to-implement 960 grid for efficient layout, as well as expertly crafted styles for typography, navigation, tables, forms, buttons, and more. To take care of everyday JavaScript touches, Bootstrap provides a set of query plugins for drop-down menus, tabs, modal boxes, tooltips, alert messages, and more. Together, these interface elements provide all the trappings of a standards compliant, user-friendly, professionally built HTML5 website, right out of the box. Bootstrap is a very simple way to promote quick, clean and highly usable applications .Providing a clean and uniform solution to the most common, everyday interface tasks developers come across.

3.3 SYSTEM DESIGN

3.3.1 Proposed system Architecture

From the analysis chapter, it has been deduced that the web based file archiving system is distributed in nature since users can access the system from anywhere hence system is based on the Three-tier architecture shown in figure 11.

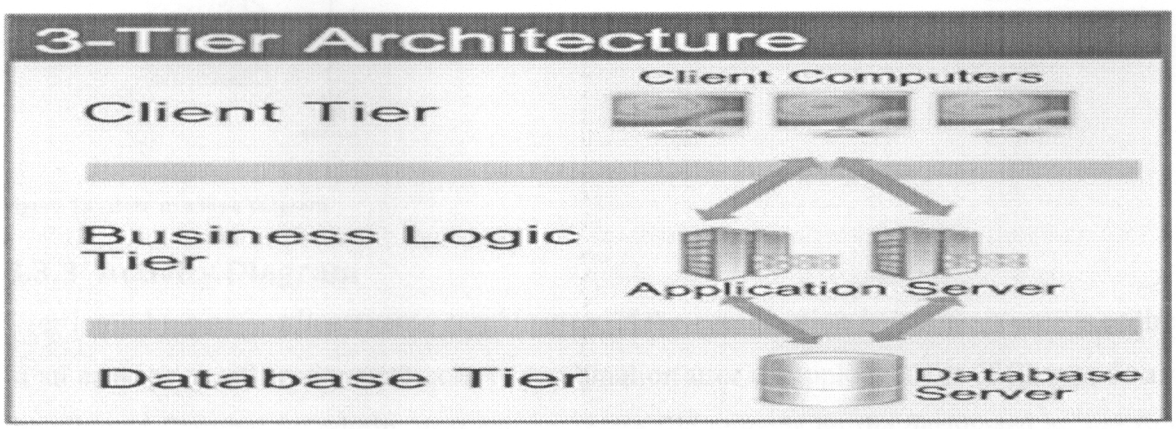


Figure 11: System blue print

• **First Tier:** The GUI is implemented in this tier, either on a thin or a fat client. A *"thin" client* such as a network computer, displays the application front end in a Web Browser, or in which

users can access it. A *"fat" client* normally a PC, offers local data processing capabilities, besides providing the same functionality as a thin client.

• **Middle Tier:** The process logic of the business processes is running on the middle tier in this case php processing queries on the local server. From time to time, middle-tier business logic is integrated on fat clients. This is where, for example file sharing and access control are implemented.

• **Third Tier:** The third tier provides the backend. This is where file and user data of an organization are stored in the database.

3.3.2 State Machine Diagram

The Permission Object contains rights assigned to user if any, system loads permissions related to a specific file, if no permission is assigned the state shows no file assigned

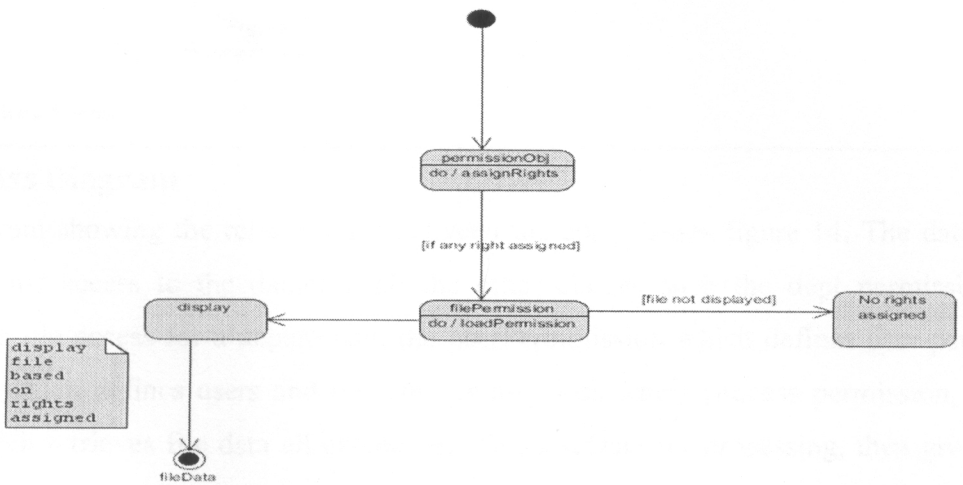


Figure 12 :state machine diagram

3.3.3 Activity Diagram

User login by password/username combination, if the combination is incorrect user is redirected to an error page, otherwise with correct combination user can upload a file if filetype is among the allowed filetypes by admin, user can complete file uploads on the dashboard by verifying if the file is the exact copy they want to upload, if they want to complete they click yes and email notification to user or department, or simply email no one if the file was not shared.

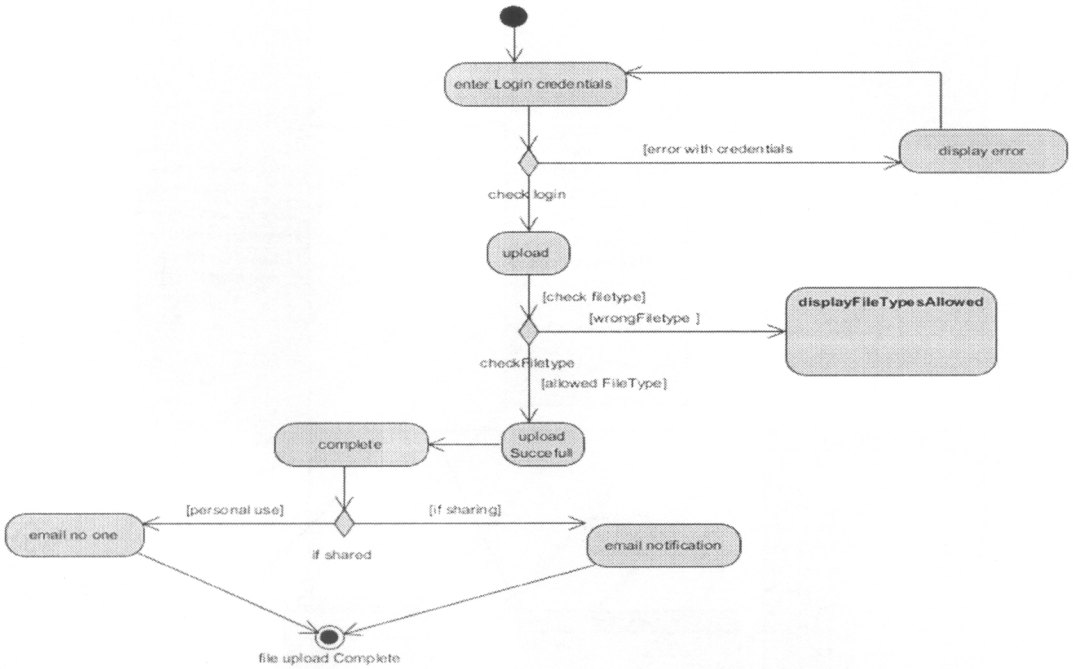


Figure 13 : activity diagram

3.3.4 Class Diagram

Class diagram showing the relationships between the core classes figure 14, The databaseData class contains access to the database all the other classes such the dept permission which implements file access for a department, the userPermmission which defines user permissions, user_class which defines users and the user_perms_class which process permission, File_data classes which retrieves file data all extends the databaseData for processing, then gives the out class the list of files to display.

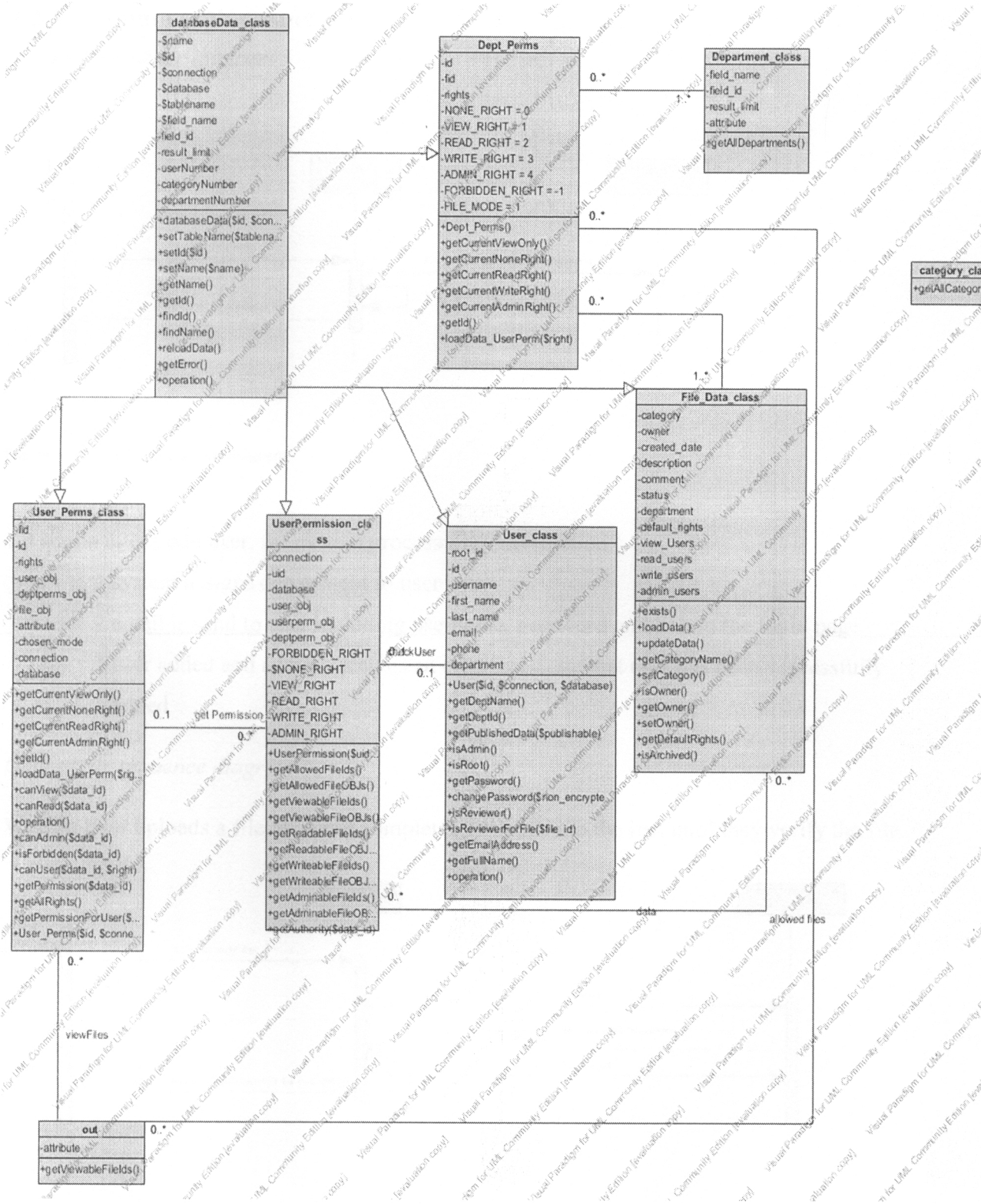


Figure 14 : class diagram

3.3.5 Sequence Diagrams

Add user sequence diagram

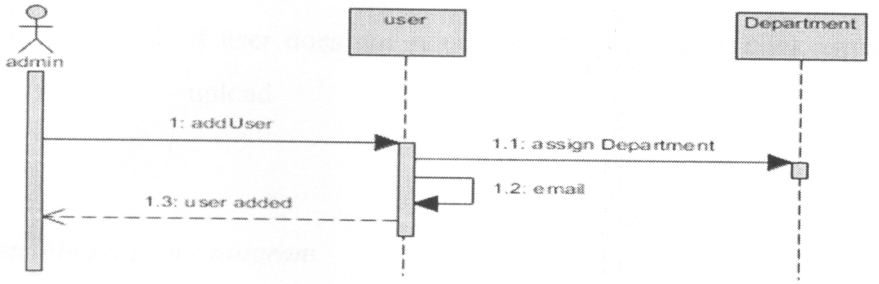


Figure 15: add user sequence diagram

1.0 admin clicks add user, the system process user information

1.1 system assigns department to user

1.2 email is send to user including username, password and link to the login page

1.3 user added and admin receives an echo message that user has been successfully added

Upload file sequence diagram

When a User uploads a file, it is not completely uploaded to the list, until they verify the file.

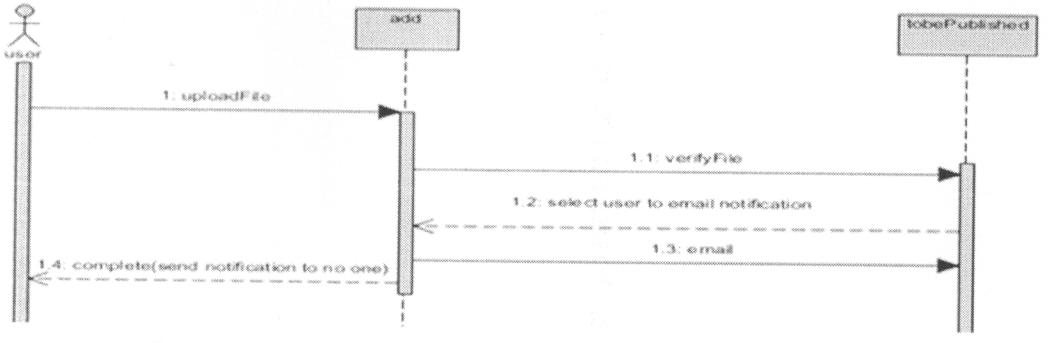


Figure 16 : upload file sequence Diagram

1.0 uploads the file from local disk.

1.1 verifies if the file is correct

1.2 selects user they have shared a file with if they want to send a notification

1.3 email sent

1.4 if user does not select user to email they click email no one and complete upload

Delete file sequence diagram

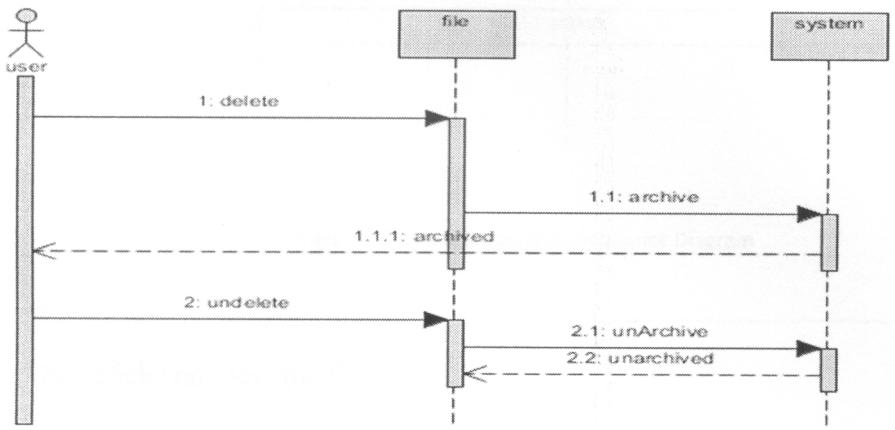


Figure 17 : Delete File Sequence Diagram

File operations sequence diagram

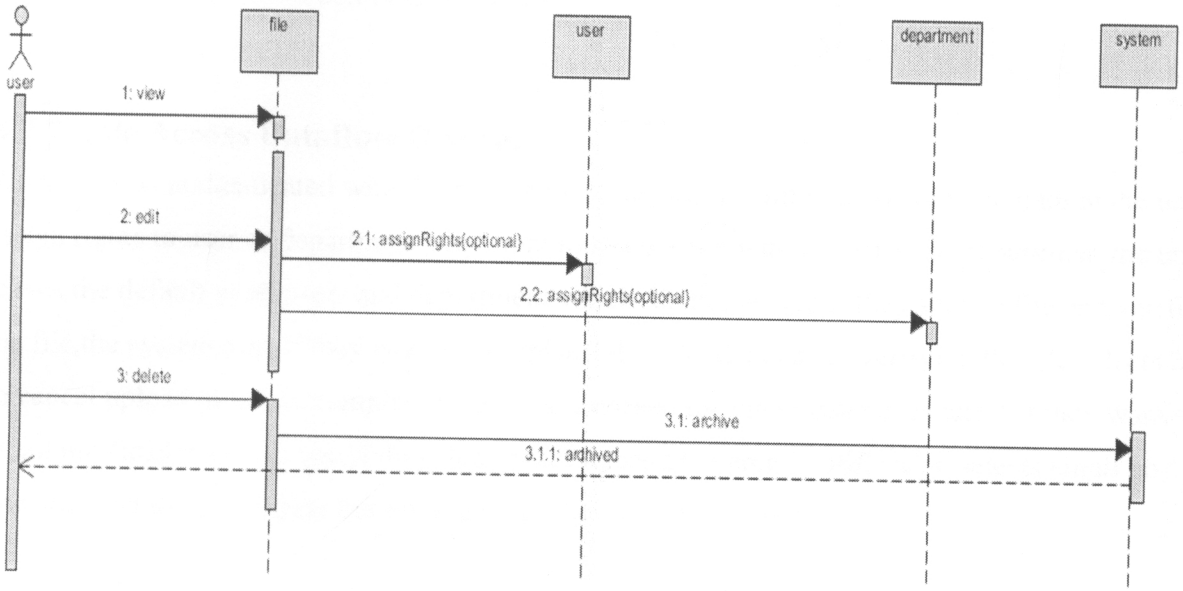


Figure 18 : File Manipulation sequence Diagram

1.0 User clicks on view the file opens

2.0 User click edit

2.1 Assigns rights to user

2.2 Assigns rights to department

3.0 User deletes the file from their dashboard invoking the archive method

3.1 Archive function processed

3.2 The system archives the file and echoes the file has been archived

3.3.6 Interface Design

The proposed system was constructed using Html, css, Javascript , Jquery, PHP and MyQSL using tweeter bootstrap layout components. It is very important to design for the user with the

most basic computer skills and also to bear in mind that different users will access the application with different computer medium.

3.3.7 File Access Dataflow Diagram

When user is authenticated with their username /password combination when uploading the user assign rights to user or department they want to share a file with,if no user or department the user leaves the default to all users and department which denies access if they share a file and verifies the file,the system asks if they want to complete file upload and after verifying the file,selects no to cancel upload or yes to complete which takes user to the email page,user select if they want to email notification to any one,if they don't want to send the email notification selects email no one and clicks the complete button. Figure 19 shows the all process.

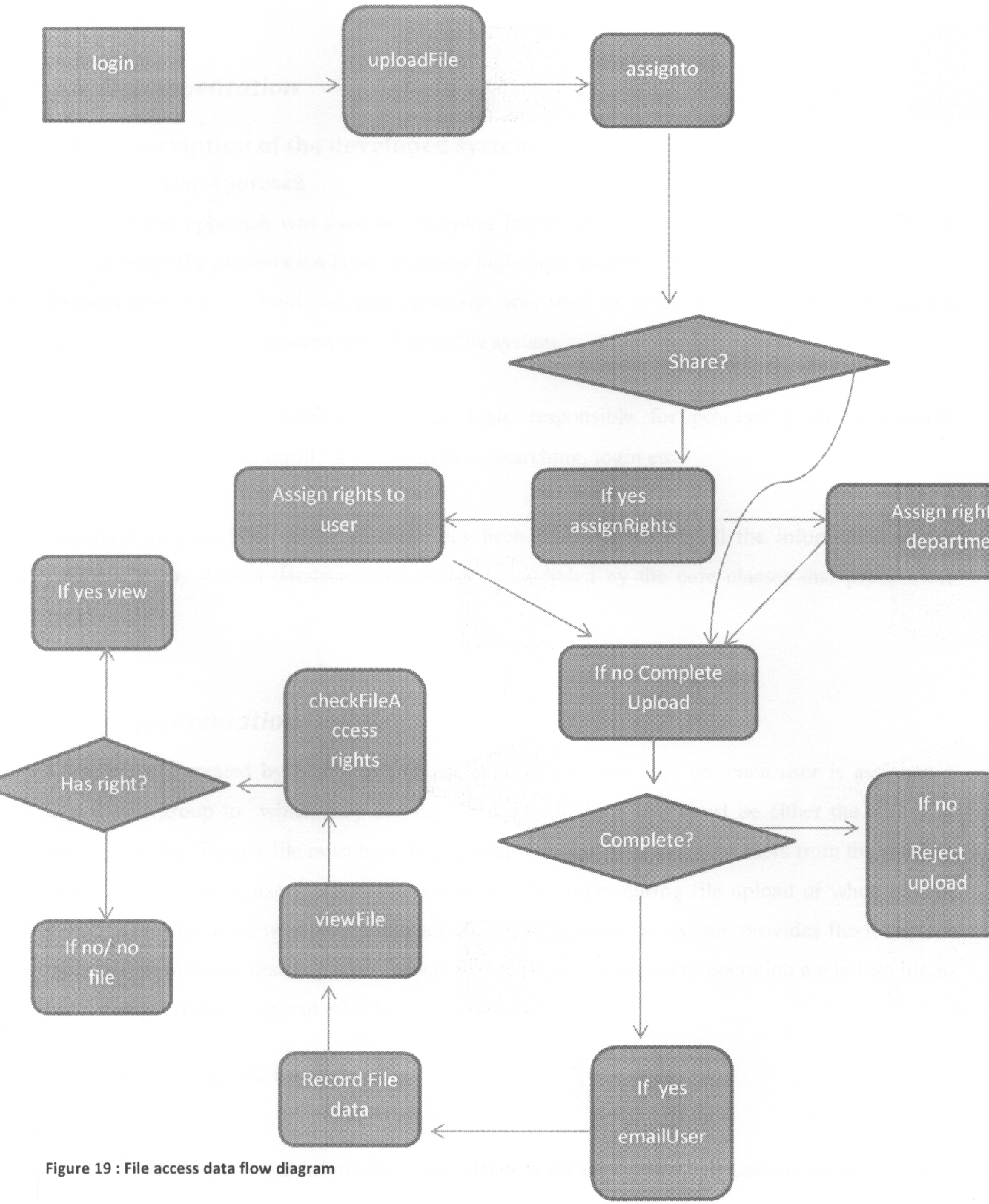


Figure 19 : File access data flow diagram

3.3.8 Entity relationship diagram (ERD)

See Appendix H

3.4 Implementation

3.4.1 Description of the developed system

The Three-Tier Approach

The three-tier approach was used in designing implementing the system. It consisted of three parts namely: the presentation layer, business/logic layer and the database layer.

Presentation Layer- html, css and javascript was used in creating the user interface which facilitates interactions between the user and the system.

Business Logic- The business layer or logic responsible for processing the application functionality such as controlling access to files, searching, login etc.

Database Layer- SQL server database has been used for storing all the information in the database layer. With a database class which is extended by the core classes that process the functionality.

3.5 Implementation Details

Users are are created by admin to prevent unnecessary user sign up, each user is assigned a department group to which they belong. To access a file a user must be either the author ie uploader of the file or a file must have been assigned rights by some other users from the same or different department, users assign file access to other users during file upload or when editing details which includes who has certain access rights e.g read, the system provides flexibility for other plugin methods that might be needed for future growth on every operation e.g before login, after login, before file upload after file upload etc.

3.5.1 Access rights Explained

View

- Read file name and information but no option for view and other functionalities

Read

- View file but hide other functionalities

Admin

- Do anything with a file

Write

- Lock the file

Forbiden

- Hide file

None (default)

- No permission set no access

3.5.2 Implementation screen shots

LOGIN PAGE

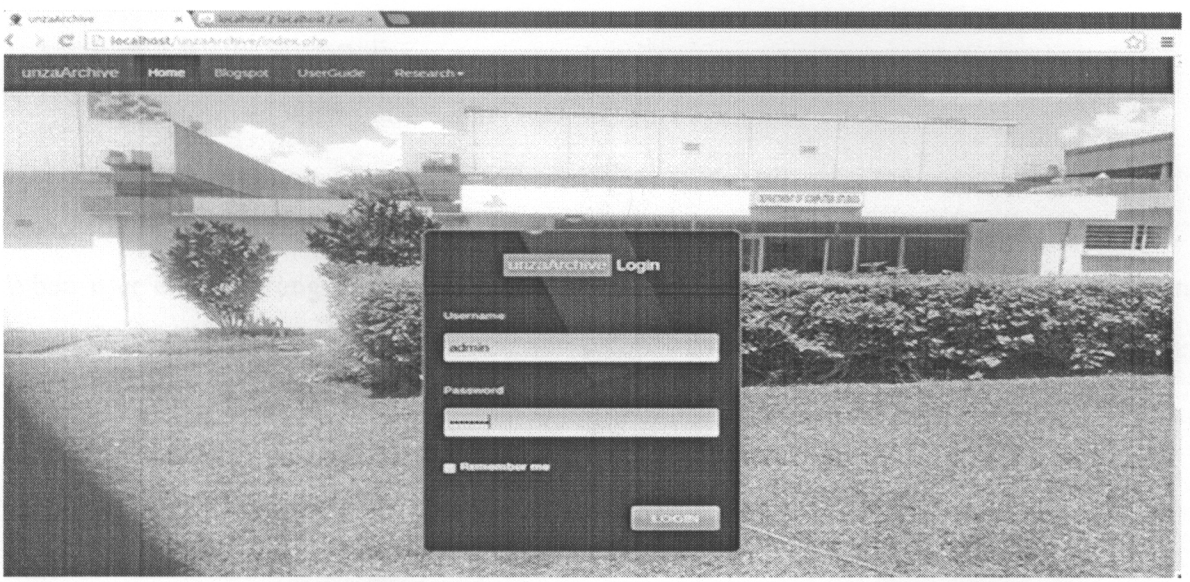


Figure 20:Login page

Login

```
// check login and sha1()
// connect and execute query
$query = "SELECT id, username, password FROM {$GLOBALS['CONFIG']['db_prefix']}user WHERE username = '$frmuser' AND password = sha1('$frmpass
$result = mysql_query("$query") or die ("Error in query. Is the database created ?: $query. " . mysql_error());
if(mysql_num_rows($result) != 1)
{
    // Check old password() method
    $query = "SELECT id, username, password FROM {$GLOBALS['CONFIG']['db_prefix']}user WHERE username = '$frmuser' AND password = password('
    $result = mysql_query("$query") or die ("Error in query: $query. " . mysql_error());
}

// if row exists - login/pass is correct
if (mysql_num_rows($result) == 1)
{
    // register the user's ID
    list($id, $username, $password) = mysql_fetch_row($result);
    // initiate a session
    $_SESSION['uid'] = $id;

    // Run the plugin API
    callPluginMethod('onAfterLogin');

    // redirect to main page
    if(isset($_REQUEST['redirection']))
    {
        redirect_visitor($_REQUEST['redirection']);
    }
```

Figure 21 : Login implementation

When user enters wrong username password combination the system errors out as shown in Figure 22.

Figure 22 : system redirects to error page

When user a user is successfully authenticated the system lands on the home pages below, from the operation drop down menu user can select which operation they want .

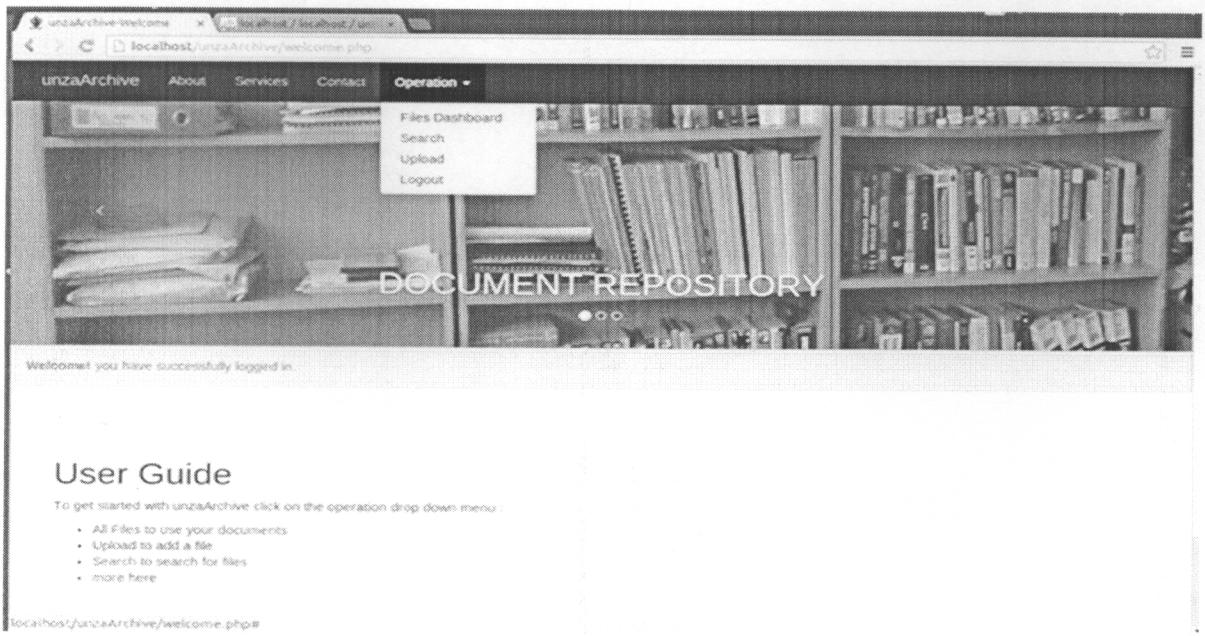


Figure 23: welcome page after correct password/user name combination

When user clicks on Files Dashboard, where they can view their files ,open a file by clicking view and view file information by clicking File name, the systems allows filtering search by filtering using department, author, file category.

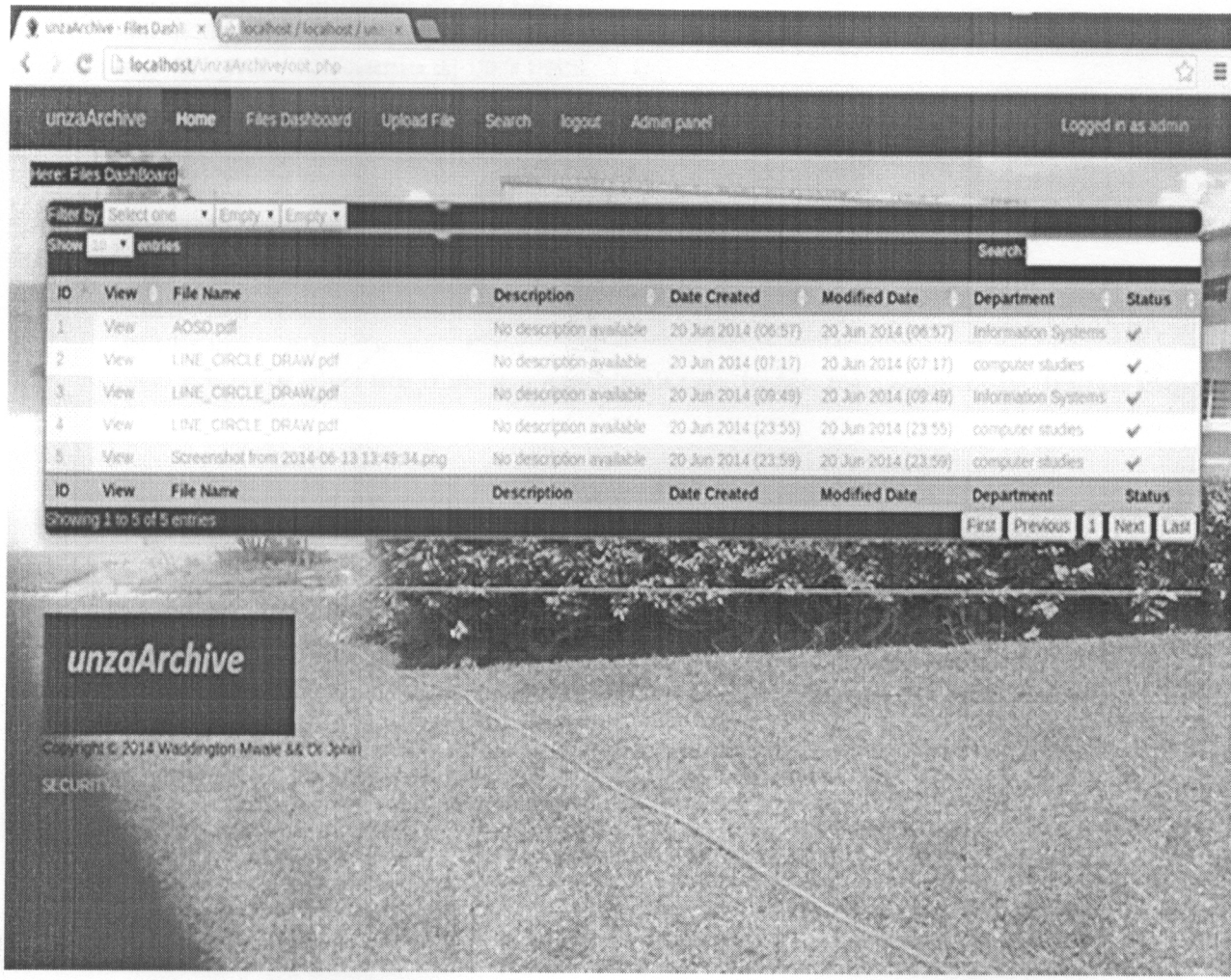


Figure 24 : system loads files user has access to

```
        $this->userperm_obj = new User_Perms($this->user_obj->getId(), $connection, $database);
        $this->deptperm_obj = new Dept_Perms($this->user_obj->getDeptId(), $connection, $database);
        $this->FORBIDDEN_RIGHT = $this->userperm_obj->FORBIDDEN_RIGHT;
        $this->NONE_RIGHT = $this->userperm_obj->NONE_RIGHT;
        $this->VIEW_RIGHT = $this->userperm_obj->VIEW_RIGHT;
        $this->READ_RIGHT = $this->userperm_obj->READ_RIGHT;
        $this->WRITE_RIGHT = $this->userperm_obj->WRITE_RIGHT;
        $this->ADMIN_RIGHT = $this->userperm_obj->ADMIN_RIGHT;
    }
    // return an array of all the Allowed files ( right >= view_right) ID
    function getAllowedFileIds()
    {
        $viewable_array = $this->getViewableFileIds();
        $readable_array = $this->getReadableFileIds();
        $writeable_array = $this->getWriteableFileIds();
        $adminable_array = $this->getAdminableFileIds();
        $result_array = array_values( array_unique( array_merge($viewable_array, $readable_array, $writeable_array, $adminable_array) ) );
        return $result_array;
    }
    // return an array of all the Allowed files ( right >= view_right) object
    function getAllowedFileOBJs()
    {
        return $this->convertToFileDataOBJ( $this->getAllowedFileIds() );
    }
    // // return an array of all the Allowed files ( right >= view_right) ID
    // One might ask why getViewableFileIds() doesn't return the combined
```

Figure 25: load allowed files implementation

To upload a file user click on upload file on fixed header

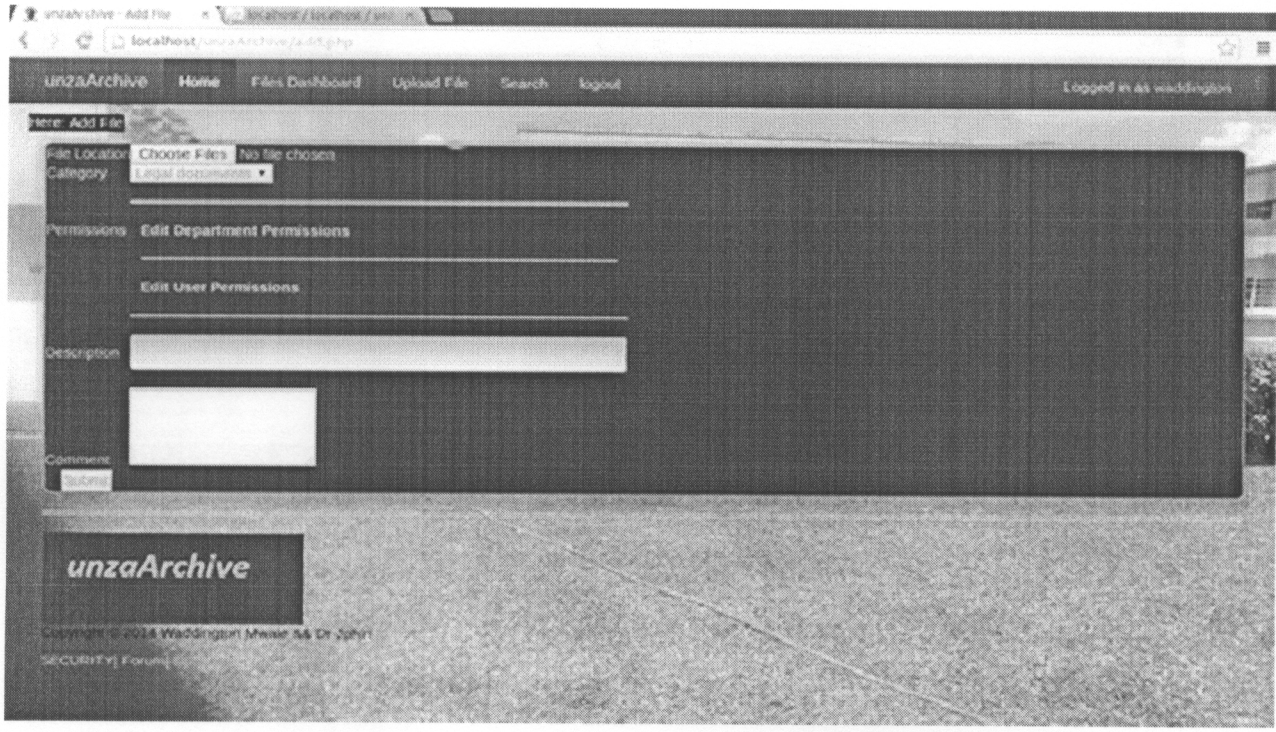


Figure 26 : File upload

Then clicks on choose file to browse the file to upload on the computer

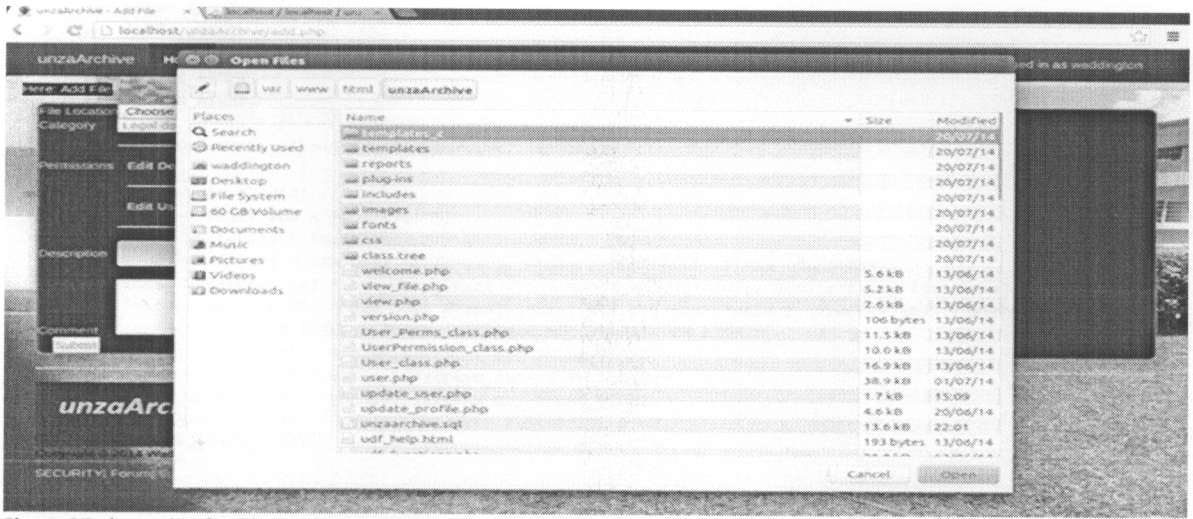


Figure 27 : browsing for file on the computer

Select the file category, e.g meeting minutes, agreements.

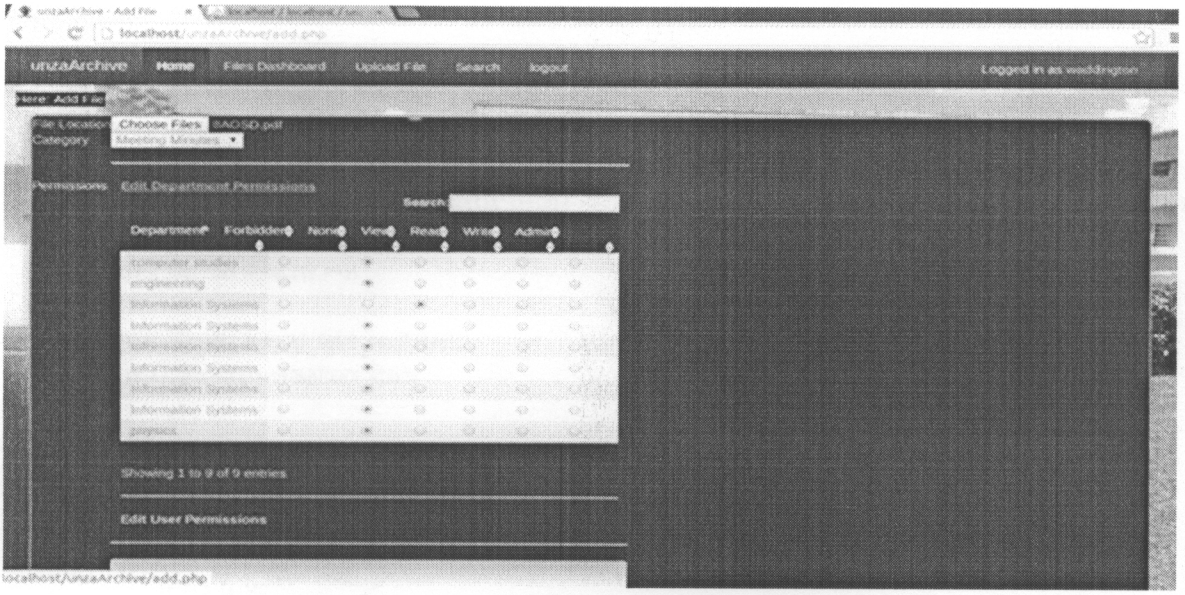


Figure 28: assigning permission

```
{
    var $connection;
    var $uid;
    var $database;
    var $user_obj;
    var $userperm_obj;
    var $deptperm_obj;
    var $FORBIDDEN_RIGHT;
    var $NONE_RIGHT;
    var $VIEW_RIGHT;
    var $READ_RIGHT;
    var $WRITE_RIGHT;
    var $ADMIN_RIGHT;

    function UserPermission($uid, $connection, $database)
    {
        $this->uid = $uid;
        $this->connection = $connection;
        $this->database = $database;
        $this->user_obj = new User($this->uid, $this->connection, $this->database);
        $this->userperm_obj = new User_Perms($this->user_obj->getId(), $connection, $database);
        $this->deptperm_obj = new Dept_Perms($this->user_obj->getDeptId(), $connection, $database);
        $this->FORBIDDEN_RIGHT = $this->userperm_obj->FORBIDDEN_RIGHT;
        $this->NONE_RIGHT = $this->userperm_obj->NONE_RIGHT;
        $this->VIEW_RIGHT = $this->userperm_obj->VIEW_RIGHT;
        $this->READ_RIGHT = $this->userperm_obj->READ_RIGHT;
        $this->WRITE_RIGHT = $this->userperm_obj->WRITE_RIGHT;
        $this->ADMIN_RIGHT = $this->userperm_obj->ADMIN_RIGHT;
    }
    // return an array of all the Allowed files ( right >= view_right) ID
    function getAllowedFileIds()
```

Figure 29 : loading assigned rights

Then assign the permission specific department, user can search the department name on the search bar if the list is too long, usually one ,two or three letters of the department will show the related departments.

Forbidden

```
//  return whether if this user is forbidden to have acc
function isForbidden($data_id)
{
    $query = "SELECT {$GLOBALS['CONFIG']['db_prefix']}$this->TABLE_USER_PERMS.rights FROM {$GLOBALS['CONFIG']['db_prefix']}$this->TABLE
    $result = mysql_query($query, $this->connection) or die("Error in query" .mysql_error() );
    if(mysql_num_rows($result) == )
    {
        list ($right) = mysql_fetch_row($result);
        if($right==$this->FORBIDDEN_RIGHT)
        {
            return true;
        }
        else
        {
            return false;
        }
    }
}

/*
```

Figure 30 : forbidding user implementation

NOTE: *To prevent users from the department where the user belongs forbid user department*

Assign rights to a specified user, if the list is too long search the name on the search bar, enter the description and comments about the file, these field can also be useful when it comes to searching the files, user can also search by comments or description.

3.5.3 Access Levels

General user

General users in unzaArchive are users that perform general system functionalities which include Searching and Accessing files on which they have rights to access, updating their profiles and sharing their files.

Admin

Admin user can perform all functions performed by a general users and they have access to the admin panel for managing users, departments, file categories, system settings and other file recovery functions.

3.5.4 Algorithims

Password encryption

Using SHA1 ($password)

Reason?

Simple-O, an automated essay grading application was developed at the Department of Electrical Engineering University of Indonesia. This application used MD5 + salt algorithm to perform protection for authentication password of users stored in its database. Unfortunately, due to a number of flaws contained in the MD5 algorithm, SHA-1 + salt algorithm was implemented in this application and then the comparison was carried out between those two algorithms. The experiments include time measurements and estimation of brute force attack for each algorithm. Processing time and CPU usage were also measured. In the brute force hash code scenario, it was tried to find plaintext from the chipertext. In this scenario, both MD5 and SHA-1 was implemented and tested using Hashcat tool. The better the algorithm, the more time needed to brute force the chipertext. In this scenario the password tested has 8 to 10 characters. The result from this testing shows that the implementation of SHA-1 algorithm is more robust against brute force attacks than MD5. The difference in processing time between SHA-1 + salt and MD5 + salt ranged from 0.001 seconds to 0.002 seconds for each length variation of the password from 8 to 10 character. While the difference in CPU usage is 0.545%, 0.985%, and 1.69% respectively for the password with 8, 9, and 10 characters length. These results indicate that while giving better security the implementation of the algorithm SHA-1 + salt does not impose on the performance of Simple-O application.[28]

3.6 Summary

This chapter has discussed how web based file archiving system was analysed using the obtained user requirements, then went on to provide the methodology that was used and why it was, described approach to implementing the system using three tier architecture by further describing in detail what technologies have been put into each tier, describing the design detail while working around the three tier architecture, Described the implementation, Technical details of implementation and Screenshots of developed system and sample code.

4 CHAPTER FOUR: Results and Discussion

4.1 Introduction

Results are very important and critical to the success of any project that aims at delivering working software. There are many types of testing that a system may be subjected to, however only the ones in the testing objectives will be carried out for this system, this chapter explains the testing scope, objectives, goals, test plan and result and gives a discussion.

4.2 Scope

The overall purpose of testing is to ensure that unzaArchive meets all of its functional and business requirements. The purpose of this chapter is to describe the overall test plan and strategy for testing the system.

4.3 Test Objectives

The main objectives of testing this system are to ensure complete validation of the business and software requirements.

Test type	Tester	Description
Unit Testing	Developer	Testing units/individual components
Integration Testing	Developer	testing combined individual units as a group
System Testing	Developer	Testing the whole system complete integrated system to evaluate compliance with specified requirements
Functional Testing	Developer	Testing the functionality of the system also known as blackbox testing.
Regression Testing	Developer	Testing the system to uncover error made whenever changes

		are made
User Acceptance Testing	System users	Testing of the system by system users for acceptance

4.4 Testing Goals

The goals in testing this system include validating the quality, usability, reliability and performance of the application. Testing will be performed from a black-box approach. Tests will be designed around requirements and functionality.

4.5 Test Plan and Results

The Test Plan is derived from the Requirements, Functional Specs, and detailed Design Specs. The Test Plan identifies the details of the tests, identifying the associated test case areas within the product. The purpose of the Test Plan is to:

- Specify the approach that Testing will use to test the product, and the deliverables break the product down into distinct areas and identify features of the product that are to be tested.
- Specify the procedures to be used for testing sign-off and product release.
- Indicate the tools used to test the product.
- List the resource and scheduling plans.

Test Case	***Test Purpose***	***Expected Outcome***	***Actual Result***
Login	*Enter wrong username password combination*	*Redirect to error page*	*Error page displayed with error message*
Update profile	*Enter non alphanumeric password*	*If password is not alphanumeric display error message*	*Profile update not succesfull, password should be alphanumeric*
Required fields	*If forms are submitted with missing fields*	*Prompt user to fill in the forms*	*User prompted*

Due to file access complexity a separate test plan table used.

four file acces rights Read,write,admin,none,forbid wer tested

Test Case	*Test Purpose*	*Expected Outcome*	*Actual Result*
Read	*Assign to check if file can't be opened*	*Link view link to open the file available*	*The link view to open the file is displayed*
Write	*Assign write to check if file the cannot be locked for writing*	*Check out option for locking the file*	*The option check out to lock the file is displayed*
Admin	*Assign admin to check if nothing can be done to a file*	*All file operation are available*	*Options to All operations that can be performed on a file are displayed*
View	*Assign view to check if file information can't be viewed*	*Option to View file details displayed*	*Option to view on the file details should be displayed*
Forbid	*Assign forbid to check if file can be accessed*	*File is never accessed*	*Nothing is about a file can be done ,it can't be seen*
None(default)	*For departments assign none and check If they can access the file*	*File not accessible*	*File can't be seen*

Figure 31 : Test results

4.6 Discussion

The work done on the system is of good standard. This is because of research that was carried out on the area of study and academic area to give a clear understanding and guidance on what was to be developed. Much of the research was done by going on the internet and searching

using search engines such as google and yahoo and online library the institution is subscribe to for journal articles.

In order to capture out and understand the correct user requirements for the proposed system, structured interviews were conducted with the members of staff. The response was overwhelming and it almost hundred percent feedback.The software product produced was fairly good, it achieved most of the user requirements, the user interface is good and is very easy to navigate, and even novice users can find their way around the web application easily. The client side validation is excellent. Performance is perfect and the error rate has is low and has been masked such that users can't cannot notice.

Compared with most of the the related systems that were reviewed web based document archiving is not only open source but provides additional functionality such as recycle bin, auditing documents, assigning users to a specific department, defining different file categories such as agreements, meeting minutes, letters.Provides sharing files with any user in the system or department to which which users belong.

The system provides plugin methods that allow the system flexible to additional functionality, e.g after login the method callPluginMethod('onAfterLogin') is called making it easy to define more security features with login.

Compared to most web based file archives which connects or reads and record information to files, web based document archive 'unzaArchive' integrates database connection to speed up the processing time.

4.7 Summary

The chapter discussed how the proposed system was subjected to various types of testing. This brought to light why it is very cardinal to test a new system before it is introduce on the main stream of an organization business, the gave the test plan and results for the system and a discussion of the system.

5 CHAPTER FIVE: Conclusion and Future works

5.1 Introduction

The aim of this chapter is to give logical conclusions of the work done, give an assessment of the completed system, discuss Results / achievements / findings, Problems faced, Limitations and give future recommendations on how the system can be improved.

5.2 Assessment of the completed system

A thorough analysis was done before the development of this system, this helped in developing a system that has proven to satisfy all its user requirements beyond expectation. The current system and tools being used were also critically analyzed, consultation was also carried out with various stakeholders throughout the development of the system. A three tier architecture has been used to implement this system, this is to allows the system to be scalable, flexible and easy to maintain. Thorough testing of all system functionality was carried out by following a comprehensive test plan and training done by me the developer of the software.

5.3 Problems encountered

When reviewing of similar existing systems that have been developed most of them did not integrate the database a very important feature that I included in my system to improve on performance and security. The system was developed by integrating modules from different related systems that were not using the database but recorded information to files, integrating database functions by replacing all file access methods with database access queries considering the different ways that files were accessed from the different modules was a very huge challenge though in the end it was a success.

Most the related systems have combined presentation logic with business logic e.g server side scripting with interface html and css, so i had to completely separate logic ,presentation and data access logic using smarty templating engine, a templating engine that needs understanding of smarty programming syntax which took most the time to understand and to be implemented.

Sending emails to other email servers was a huge challenge though I managed to send to gmail.

Finding research papers was difficult because not much of research has been done in this field, so I had rely on online journal, magazines and other IEEE articles to understand the theory.

5.4 Limitations

Even though the system is able to provide all general file archiving functions the system can only

- Run on a web server with integrated mysql server
- Needs reasonable internet connection or local LAN with not more than 20 users
- Needs a machine dedicated to file archiving to improve on performance
- To be used by a group of users on a LAN they need to be connected to the machine acting as server

5.5 Lessons learnt

The project has helped me to appreciate what I have learnt in other courses so far as it made me apply different concepts to develop the system. The project has also allowed me to learn to use different system developmental tools. Though it is extremely challenging and time consuming, the practical experience gained from it is worthwhile.

Project Management skills have helped me to work and on time and taking the process of planning seriously for the product to be delivered on time. The project management knowledge has helped on how to schedule meetings with different stakeholders and making sure every meeting is according to the agenda and date / time set.

5.6 Future works

- Archiving user details when they are deleted from the system
- Sending email notifications to all email sites , though the system can send to Gmail
- Setting file expiration Date specifically to a file, unzaArchive system implements system wide file expiration date.

- Shift all root admin functions to department level admin and leave only the system settings function to root admin.
- Integrating the system with scanner.

5.7 Summary

In this chapter I elaborated my learning experiences from the project, discussed the results and gave future recommendations for the extension and improvement of the system, logical conclusions of the work carried out and an assessment of the software product was done.

6 REFERENCES

[1] Michael C. Whitlock "Data archiving in ecology and evolution": best practices, (2011) ,0169-5347

[2] Demetris Antoniades, Evangelos P. Markatos, Constantine Dovrolis,One-Click Hosting Services: A File-Sharing Hideout(2009)

[3] Cong Wang and Kui Ren, Wenjing Lou, Jin Li(2010) Toward Publicly Auditable Secure Cloud Data Storage Services,19-23

[4] Mark J. Costello "Motivating Online Publication of Data", (2009) , Vol. 59 No. 5:

[5] Michael C. Whitlock, Mark A. McPeek, Mark D. Rausher, Loren Rieseberg, Allen J. Moore "Data Archiving", (2010), Vol. 175, No. 2,

[6] H. B. Kekre, Sudeep D. Thepade "Image Retrieval using Augmented Block Truncation Coding Techniques", (2009),384-385

[7] Mark D. Rausher, Mark A. McPeek, Allen J. Moore, Loren Rieseberg, and Michael C. Whitlock (2009) DATA ARCHIVING Evolution 64-3: pp603–604

[8]. Strasser, C. DataUp: Enabling data stewardship for researchers. IConference 2013 Proceedings, (2013). pp. 657-658). doi:10.9776/13300

[9] Emilio M. Bruna ."Scientific Journals can Advance Tropical Biology and Conservation by Requiring Data Archiving". BIOTROPICA, (2010) , 42(4): pp399–401

[10] Elias, T. "Universal instructional design principles for Moodle". *International Review of Research in Open and Distance Learning,* (2010). *11*(2), pp110-124.

[11] Fetaji, B., & Fetaji, M. "E-learning indicators": A multi-dimensional model for planning and evaluating e-learning software solutions. *Electronic Journal of e-Learning,* (2009). *7*(1),pp 1-28.

[12] Gibbs, G. & Simpson, C. "Conditions under which assessment supports students' learning". *Learning and Teaching in Higher Education,* (2005). *1*(1), pp3-31.

[13] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT plaforms: vision, hype, and reality for delivering computing as the 5th utility", Future Generation Computing Systems,(2009).vol. 25(6), pp. 599-616

[14]B.J. Liu, F. Cao, M.Z. Zhou, G. Mogel, L. Documet, "Trends in PACS image storage and archive", Computerized Medical Imaging and Graphics ,(2009).

[15] P.T. Jaeger, T. Lin, and J.M. Grimes, "Cloud computing and information policy" computing in a policy cloud", Journal of Information Technology & Politics, vol 5(3), pp. 269-283, 2010

[16] J. Massy (2005). "The eLearning industry and market in Europe". Available at http://ec.europa.eu/education/archive/elearning/doc/studies/market_annex1_en.pdf. [Accessed April 15, 2010].

[17] F.D. Davis (1989). Perceived Usefulness,Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, vol. 13, pp. 319-340.

[19] F.D. Davis, R.P. Bagozzi, and P.R. Warshaw(1989). User Acceptance of Compute Technology: A Comparison of Two Theoretical Models. *Management Science*, vol. 35, pp. 982 1003.

[20] W. King and J. He (2006). A meta-analysis of the technology acceptance model. *Information & Management*, vol. 43, pp. 740-755.

[21] J.H. Sharp (2006). Development, Extension and Application: A Review of the Technology Acceptance Model. *Proc ISECON*, vol. 23.

[22]M.K. Lee, C.M. Cheung, and Z. Chen (2005).Acceptance of Internet-based learning medium: the role of extrinsic and intrinsic motivation.*Information & Management*, vol. 42, pp. 1095-1104.

[23] E.M. van Raaij and J.J. Schepers (2008). The acceptance and use of a virtual learning environment in China. *Computers & Education*,vol. 50, pp. 838-852.

[24] E. Ngai, J. Poon, and Y. Chan (2007). Empirical examination of the adoption of WebCT using TAM. *Computers & Education*, vol. 48, pp. 250-267.

[25] S. Liu, H. Liao, and J.A. Pratt (2009). Impact of media richness and flow on e-learning technology acceptance. *Computers & Education*, vol. 52, pp.599-607.

[26] Moons, K. G., Kengne, A. P., Woodward, M., Royston, P., Vergouwe, Y., Altman, D. G., & Grobbee, D. E. (2012). Risk prediction models: I. Development, internal validation, and assessing the incremental value of a new (bio) marker. *Heart*, *98*(9), 683-690

[27] Capers Jones, VP and CTO, Namcook Analytics LLC.Laws of Software Engineering Circa 2014. Version 7.0 February 17, 2014

[28] Ratna, Putri, et al. "Analysis and comparison of MD5 and SHA-1 algorithm implementation in Simple-O authentication based security system." *QiR (Quality in Research), 2013 International Conference on.* IEEE, 2013.

7 APPENDICES

7.1 *Appendix A* *GANNT CHART*

ID	Task Name	Duration	Start	Finish
1	**increment 1**	**51 days?**	**Mon 06/01/14**	**Mon 17/03/14**
2	communication	1 day	Mon 06/01/1	Mon 06/01/1
3	planning	2 days	Tue 07/01/1	Wed 08/01/1
4	modelling(analysis,desig	3 days	Thu 09/01/1	Mon 13/01/1
5	construction(code,testin	44 days	Tue 14/01/1	Fri 14/03/1
6	deployment(delivery,fee	1 day?	Mon 17/03/1	Mon 17/03/1
7	**increment 2**	**32 days?**	**Tue 18/03/14**	**Wed 30/04/14**
8	communication	1 day	Tue 18/03/1	Tue 18/03/1
9	planning	2 days	Wed 19/03/1	Thu 20/03/1
10	modelling	3 days	Fri 21/03/1	Tue 25/03/1
11	construction	25 days	Wed 26/03/1	Tue 29/04/1
12	deployment	1 day?	Wed 30/04/1	Wed 30/04/1
13	**increment3**	**43 days?**	**Thu 01/05/14**	**Mon 30/06/14**
14	communication	1 day?	Thu 01/05/1	Thu 01/05/1
15	planning	2 days	Fri 02/05/1	Mon 05/05/1
16	modelling	3 days	Tue 06/05/1	Thu 08/05/1
17	construction	25 days	Fri 09/05/1	Thu 12/06/1
18	deployment	2 days	Fri 13/06/1	Mon 16/06/1
19	documentation	10 days	Tue 17/06/1	Mon 30/06/1

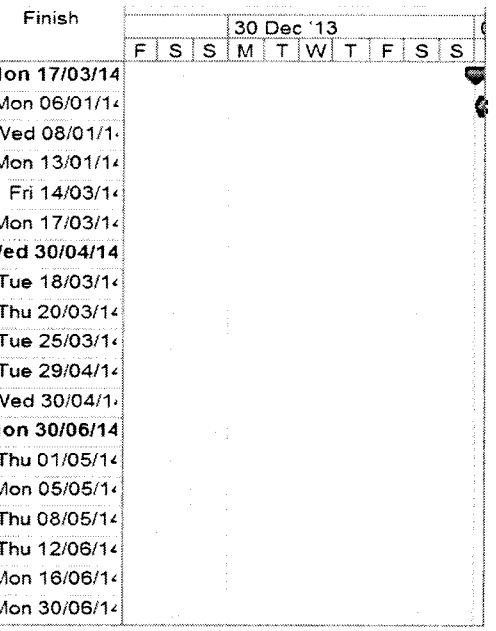


Figure 32 : Gannt chart

7.2 *Appendix B* Sample code

```
//Configuration file config.php

<?php

// Eliminate multiple inclusion of config.php

if( !defined('config') ){

define('config', 'true', false);

// config.php - useful variables/functions

// ** MySQL settings - You can get this info from your web host ** //

/** The name of the database for unzaArchive*/
```

```
define('DB_NAME', 'DATABASE NAME');
/** MySQL database username */
define('DB_USER', 'root');
/** MySQL database password */
define('DB_PASS', 'DATABASE PASSWORD');
/** MySQL hostname */
/* The MySQL server. It can also include a port number. e.g. "hostname:port" or a path to a
 * local socket e.g. ":/path/to/socket" for the localhost. */
define('DB_HOST', 'localhost');
/* * Prefix to append to each table name in the database (ex. unz_ would make the tables
 * named "unz_users", "unz_data" etc. Leave this set to the default if you want to keep
 * it the way it was. If you do change this you manually go through and re-name the database tables to match.
 * @DEFAULT 'unz_'
 * @ARG String
 */
$GLOBALS['CONFIG']['db_prefix'] = 'unz_';
/*** DO NOT EDIT BELOW THIS LINE ***/
/** Absolute path to the unzaArchive directory. */
if ( !defined('ABSPATH') )
        define('ABSPATH', dirname(__FILE__) . '/');}
//  Database class
<?php
/*databaseData_class.php - sets up database schema and provides various db functions*/
if( !defined("databaseData_class") );{
 define("databaseData_class", "true", false);
    //DO NOT INSTANTIATE THIS ABSTRACT CLASS
  class databaseData {
```

```
var $DB_PREFIX;
var $TABLE_ADMIN = 'admin';
var $TABLE_CATEGORY = 'category';
var $TABLE_DATA = 'data';
var $TABLE_DEPARTMENT = 'department';
var $TABLE_DEPT_PERMS = 'dept_perms';
var $TABLE_DEPT_REVIEWER = 'dept_reviewer';
var $TABLE_LOG = 'log';
var $TABLE_RIGHTS = 'rights';
var $TABLE_USER = 'user';
var $TABLE_USER_PERMS = 'user_perms';
var $FORBIDDEN_RIGHT = -1;
var $NONE_RIGHT = 0;
var $VIEW_RIGHT = 1;
var $READ_RIGHT = 2;
var $WRITE_RIGHT = 3;
var $ADMIN_RIGHT = 4;
var $name;
var $id;
var $connection;
var $database;
var $tablename;
var $error;
var $field_name;
var $field_id;
var $result_limit;
var $userNumber;
```

```
var $categoryNumber;
var $departmentNumber;
var $adminNumber;
function databaseData($id, $connection, $database) {
  $this->connection = $connection;
  $this->database = $database;
  $this->setId($id); //setId not only set the $id data member but also find and set name
  $this->result_limit = 1; //expect unique data fields on default  }
function setTableName($tablename) {
  $this->tablename = "$tablename"  }
function setId($id)  {
  /*setId($id) sets the data member $id and it also look
        a name that is correspondent to that id and set it to
        the data member field $name*/
  $this->id = (int) $id;
  $this->name = $this->findName()  }
function setName($name)  {
  /*setName can only be used under the assumption that
        the name field in the DB is unquie, e.g. username*/
  $this->name = $name;
  $this->id = findId();  }
function getName()  {
  return $this->name;}
function getId()  {
  return $this->id;  }
function findId()
$query = "SELECT {$this->database}.{$GLOBALS['CONFIG']['db_prefix']}{$this->tablename}.$this->field_id FROM {$this-
```

```
>database}.{$GLOBALS['CONFIG']['db_prefix']}{$this->tablename}  WHERE  {$this-
>database}.{$GLOBALS['CONFIG']['db_prefix']}{$this->tablename}.$this->field_name='$this-
>name'";

    $result = mysql_query($query, $this->connection) or die ("Error in query: $query. " .
mysql_error());

    if( mysql_num_rows($result) > $this->result_limit AND result_limit != 'UNLIMITED') {

      /*if the result is more than expected error var is set*/

       $this->error='Error: non-unique'; }

    elseif (mysql_num_rows($result) == 0) {

       // record must exist.  Error message is stored

       $this->error = 'Error: unable to fine id in database';}

    else {

       list($id) = mysql_fetch_row($result);    }

   return $id;      }

  /* logic in findName() is simular to findId().  Please look at findId()'s

     comments if you need help with this function */

  function findName()      {

     $name = '';

     $query  =  "SELECT  {$this->database}.{$GLOBALS['CONFIG']['db_prefix']}$this-
>tablename.$this->field_name  FROM  {$this-
>database}.{$GLOBALS['CONFIG']['db_prefix']}$this->tablename  WHERE  {$this-
>database}.{$GLOBALS['CONFIG']['db_prefix']}{$this->tablename}.$this->field_id  =  $this-
>id";

     $result = mysql_query($query, $this->connection) or die ("Error in query: " .$query .
mysql_error());


     if(mysql_num_rows($result) > $this->result_limit AND result_limit != 'UNLIMITED') {

       $this->error='Error: non-unique'; }

     elseif (mysql_num_rows($result) == 0)  {

       $this->error = 'Error: unable to find id in database';      }

     else      {
```

```
        list($name) = mysql_fetch_row($result);
        return $name;          }     }
    //assuming that userid will never change
    function reloadData()  {
      /* Since all the data are set at the time when $id or $name
            is set.  If another program access the DB and changes any
            information, this OBJ will no longer contain up-to-date
            information.  reloadData() will reload all the data */
      $this->setId($this->id);      }
   function getError()  {
      /* Get error will return the last thrown error */
return $this->error;}
    function combineArrays($high_priority_array, $low_priority_array)       {
      /* combineArrays() uses a linear search agolrithm with the
            cost of n*n, n being the size of the biggest array.  combineArrays()
            gives $high_priority_array the advantage by merging the
            low_priority_aray onto it.  One can look at these two arrays
            as 2 sets and cobineArrays acts as a union operator.
            For briefness, let's $high = $high_priority_array and
            $low = $low_priority_array */
      $found = false;
      $result_array = array();
      $result_array = $high_priority_array; //$high is being kept
      $result_array_index = sizeof($high_priority_array);
      //iterate through $low
      for($l = 0 ; $l<sizeof($low_priority_array); $l++) {
        /* each $low element will be compared with
```

```
                    every $high element*/
            for($r = 0; $r<sizeof($result_array); $r++) {
              if($result_array[$r] == $low_priority_array[$l]){
                /* if a $low element is already in the
                                        $high array, it is ignored */
                $r = sizeof($result_array);
                $found = true;
              }          }
            /* if certain $low element is not found in $high, it
                    will be append to the back of high*/
            if(!$found)
            {
              $result_array[$result_array_index++] = $low_priority_array[$l];        }
            $found = false;       }
          return $result_array;  }
        function convertToFileDataOBJ($fid_array) {
          $filedata_array = array();
          for($i = 0; $i<sizeof($fid_array); $i++)       {
            $filedata_array[$i] = new FileData($fid_array[$i], $this->connection, $this->database,
"data");      }
          return $filedata_array;      }  }
// end inclusion control
}
```

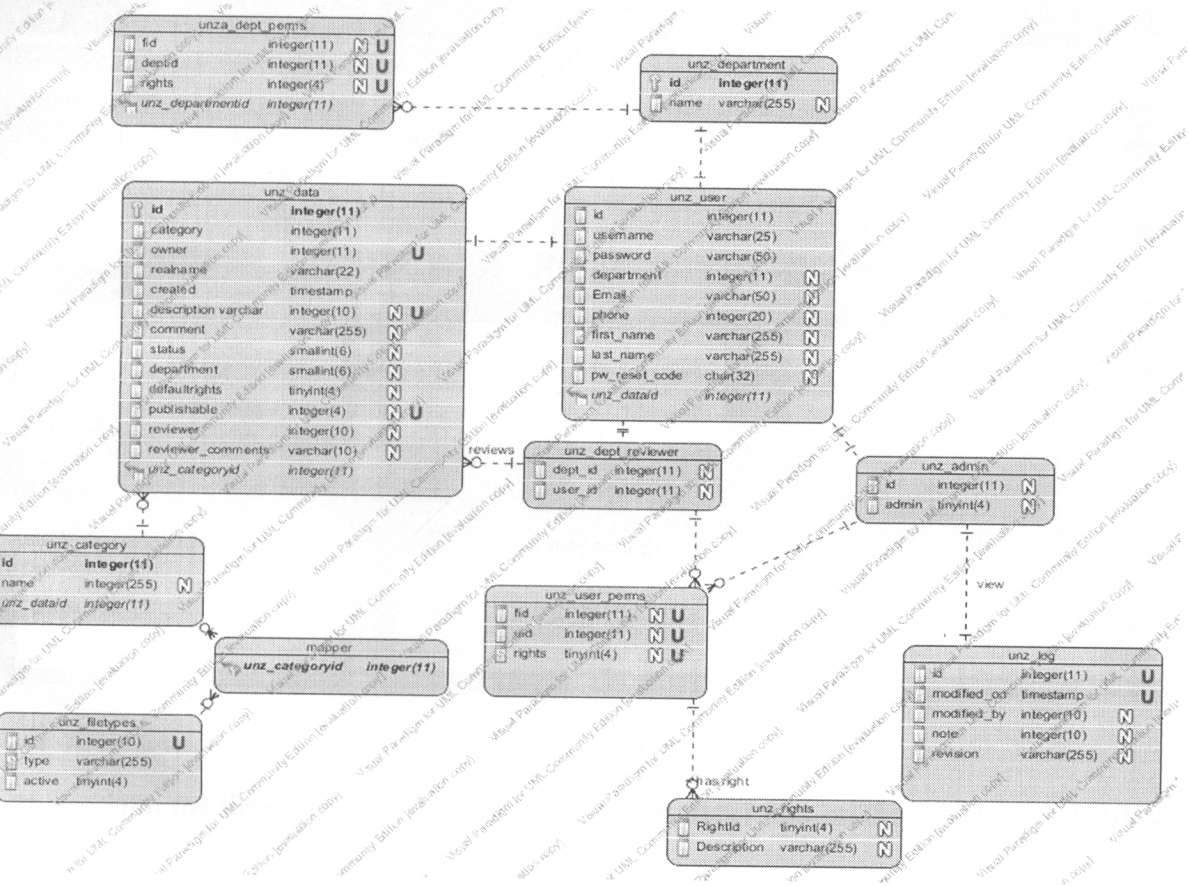
7.3 *Appendix C* Installation

New Installation

- Untar/Unzip files into any dir in your webserver documents dir

- Create a MySQL database/username/password.
- Make a directory for the files to be stored that is accessible to the web server but not available by browsing. Ensure the permissions are correct on this folder to allow for the web server to write to it .e.g C:/docs on windows or create one using command mkdir /usr/local/unzaArchive/doc on linux
- Edit the config.php to include your database parameters
- Edit the database.sql file. You need to change the values set in the unz_settings table, specifically for these two entries:
  dataDir-for storage of files, base_url – application url , Change to reflect the correct locations.
- Import database.sql file into your database
- With your web server up Visit http://localhost/unzaArchive username admin (no password)

4 *Appendix D* Entity Relationship Diagram



re 33 : Entity Relationship Diagram

5 *Appendix E* ScreenShots of Admin panel and Searching

min panel

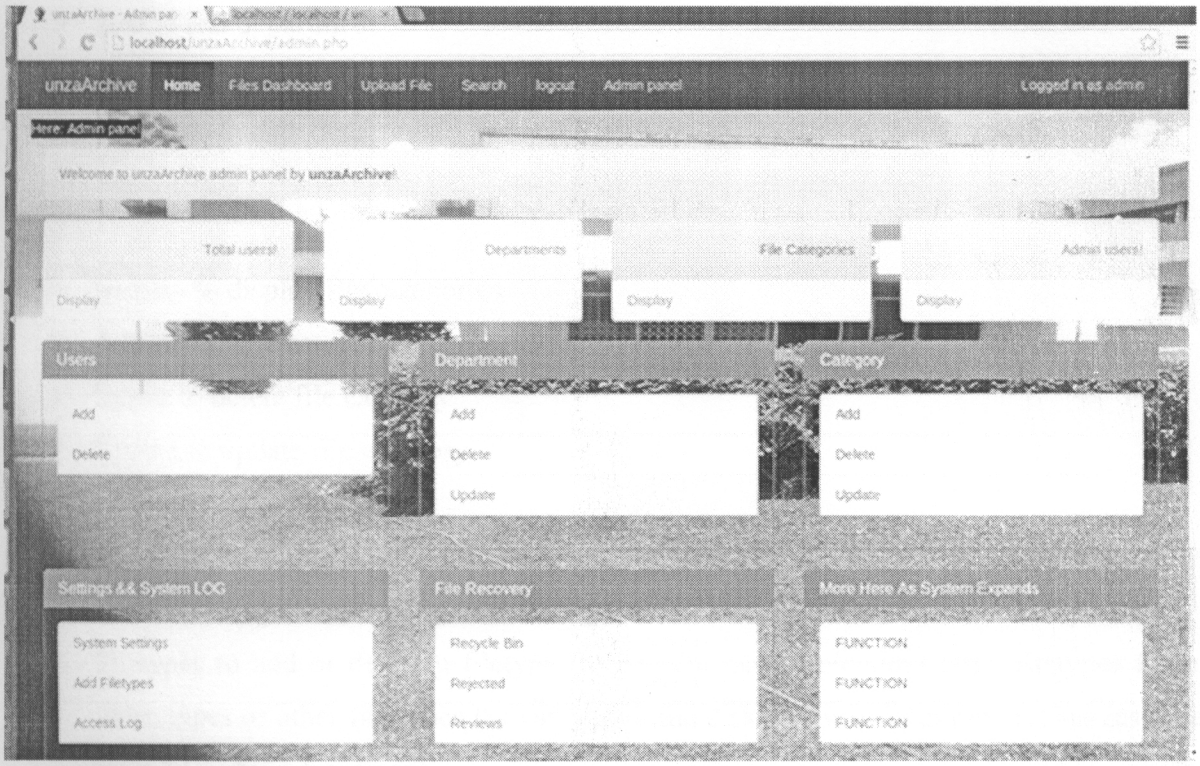


Figure 34 : Admin panel

If user is admin the they will be able to see the Admin panel link on the top fixed bar Admin panel links

Note : due to different organizational polices some of the system functionalities on admin panel can be removed for privacy.

Display is used to display users in the system, show a list of users from which a specificied user can be searched and clicked to view his/her details.

Users

- Add-adding a user to a system
- Deleting users
- Department

Departments

- Display to display departments list

- Add to add department
- Delete to delete a department
- Update to update department

Category

- Display to display file categories
- Add to add file category
- Delete to delete file category
- Update to update file category

Settings and system log

- System settings to change system settings
- Filetypes to add or delete a filetype (to prevent users uploading large filetypes for e.g video types or other file types the can shrink the disk size this functionality was restricted moved to admin functions)
- Access log used to view what has been happening to files, e.g deleted files ,uploaded files

File Recovery

- Recycle bin to recycle deleted files
- Rejected files to view files rejected during file upload.
- Reviews to view files on which upload has not been completed

Search

- User can search files by author/uploaded last and first name, department, category , Description, filename, comment and file number

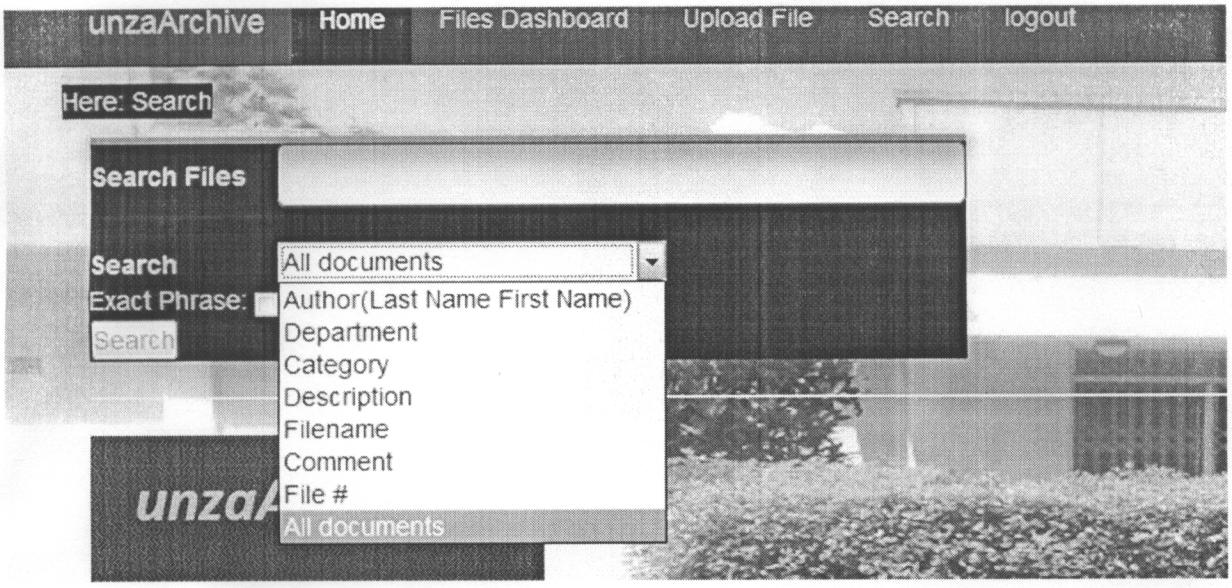


Figure 35 : Search