REPUBLIC OF THE PHILIPPINES

PANGASINAN STATE UNIVERSITY

College of Engineering and Architecture

Computer Engineering Department

CpE 222 Software Design

[PSU-UC Computer Engineering Department Document Management System]

REQUIREMENTS SPECIFICATIONS

[Arce, Hans Nathaniel]

[Balbin, Ervin]

[Cayabyab, Alfonso]

[Episcope, Maria Carmela]

[Esporsado, Dannylyn]

[Ignacio, Cris John Paul]

[Ines, Avegail]

[Rivera, Marlon]

[Sarlatan, Edrian Ely]

[Soriano, John Angelo C.]

June 24, 2022

Chapter 1 Introduction

1.1 Purpose

The purpose of this document is to provide the necessary background information, system scope, and specifications for the development and implementation of a paperless document management system. The main usage is to add the documents under the repository of Pangasinan State University Urdaneta Campus, Department of Computer Engineering.

1.2 Scope

The Document Management System Specification provides generic requirements for ensuring adequate document management functions in the system. By providing the specification it leads to understanding on the complexity of the functions the clients can expect from a quality document management system.

- The type of software product to be developed is

  a DBMS, the product will address the PSU-UC Computer engineering department physical document repository by turning it into a digital document repository. Letting users browse the system for a document or search for the desired document through the search function. Basically, the system will be developed and implemented with the main usage of accessing, tracking, and editing stored documents. DMS is a type of electronic cabinet that is used to store and organize digital files.

- **Authorization Functions**

  Consist of a login and register class in order to access the repository system. Users can also update their login details in the main interface of the system.

- **Document Functions**

  Document function Is a generalization of various subfunctions that correlates to the overall function which is the document function requirement Users can add, update, delete and even share the document.

1.3 Project Constraints and Limitations

The main constraint of this project is time, thus from the initial round-robin discussion of functionalities of the project a lot of it was dismissed, mainly the tag or the classification of the different type of documents in the repository.

- Document Classification – instead of having a tag for the document by its document type a search function can be used to write the the desired document, in short, the document can only be easily accessed if it was uploaded and was named in a manner of “docType | docAuthor – docName”.
- Document Status – in one of the flowcharts in chapter 2 system analysis namely document control. document control ensures that document status is maintained for controlled documents. There are three general statuses for a document:
    - Not yet approved
    - Approved
    - No longer approved

  For retrievability and correlation with use, some further delineation of status designations has been established. The status is used as the basis for determining document use limitations.

1.4 Hardware Requirements

(These are the minimum hardware requirements for the software or project to work properly.

PC localhost: At least 4GB of RAM

At least 20GB of hard disk

Cloud host/ localhost: Heroku or pythonanywhere for cloud host implementation or SQLite for localdisk

1.5 Software Requirements

Windows 10, JavaScript, Python 3.8 or latest, Django 4.0.3 or latest, Bootstrap v5

1.6 Definitions of Terms

Document – a piece of written, printed, or electronic matter that provides information or evidence or that serves as an official record.

Management – is an organizational function that oversees a group of individual projects linked together through a shared organizational goal or common area of impact.

JavaScript – an object-oriented computer programming language commonly used to create interactive effects within web browsers.

Python – is an interpreted, object-oriented, high-level programming language with dynamic semantics.

Django – an advanced Web framework written in Python that makes use of the model view controller (MVC) architectural pattern.

Database – a repository of information managed by a database engine which ensures integrity of data and fast access to the data

Framework – A framework in programming is a tool that provides ready-made components or solutions that are customized in order to speed up development.

Website - A website is a set of data and information about a particular subject which is available on the internet.

SQLite3 – a software library that provides a relational database management system.

Flowchart – is a diagrammatic representation of an algorithm. Since this is an SRS a cohesive representation of charts that maps out the relation between function is needed in order to understand the process of the system.

Data Flow Diagram – data flow diagram (DFD) maps out the flow of information for any process or system.

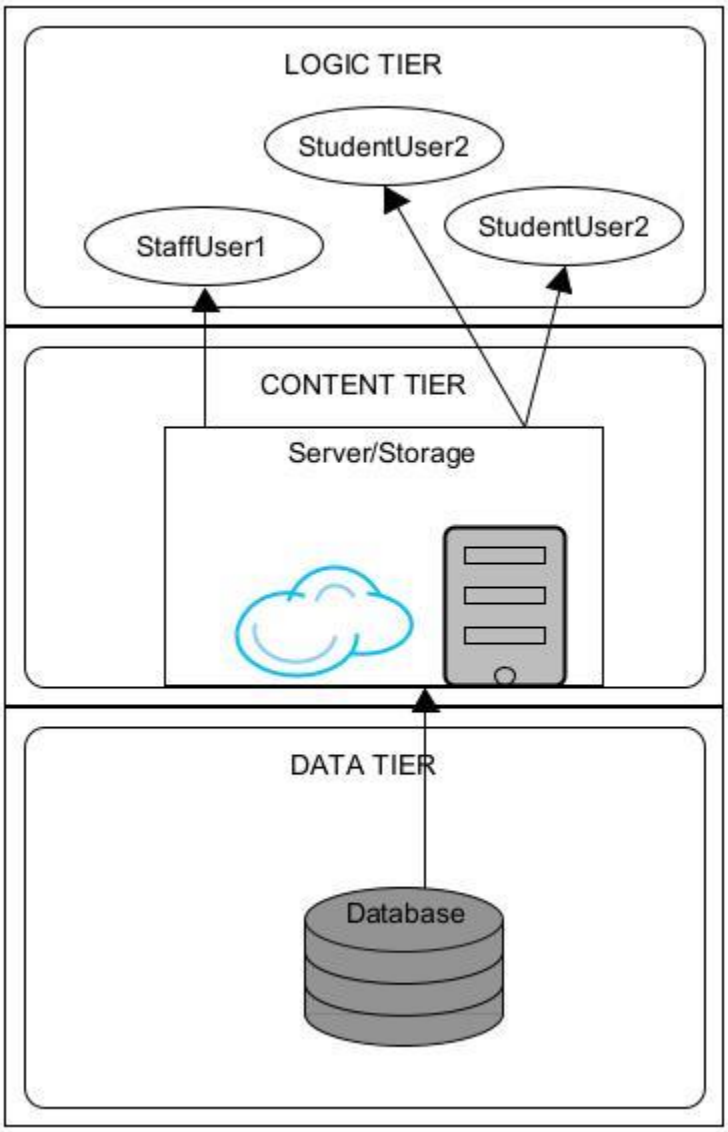
Entity Relationship Diagram – is a visual representation of the table's structure and the relationships between logically related tables.

1.7 References

- Functional Specifications *for Electronic Records Management Systems Software, 2006, ISBN 1-920807-34-9*
- *Estrera, Paul Joseph M.: Electronic Document Management System for Higher Education Institution. 2017, ISSN No.: - 2456 – 2165*
- *Thamrin, Taqwan.: Document Management System Based on Paperless. June, Publication*
- *K. Konishi, N. Furukawa and H. Ikeda," Data Model and Architecture of a Paper-Digital Document Management System", DocEng'07, August 2831, Winnipeg, Manitoba, Canada, ACM 2007.*
- "*Djang Tutorial*". *Playlists* [The Net Ninja Yt Ch], Youtube. Retreived May 27,2022 from https://www.youtube.com/playlist?list=PL4cUxeGkcC9ib4HsrXEYpQnTOTZE1x0uc&ab_channel=TheNetNinja
- "*Django Online Library Management System (BookApp)*". *Playlists* [Learn Python], Youtube. Retrieved May30, 2022 from https://www.youtube.com/playlist?list=PLpKjJ2hdIt2C7yLtX1GeNDnASIM2pAMPx

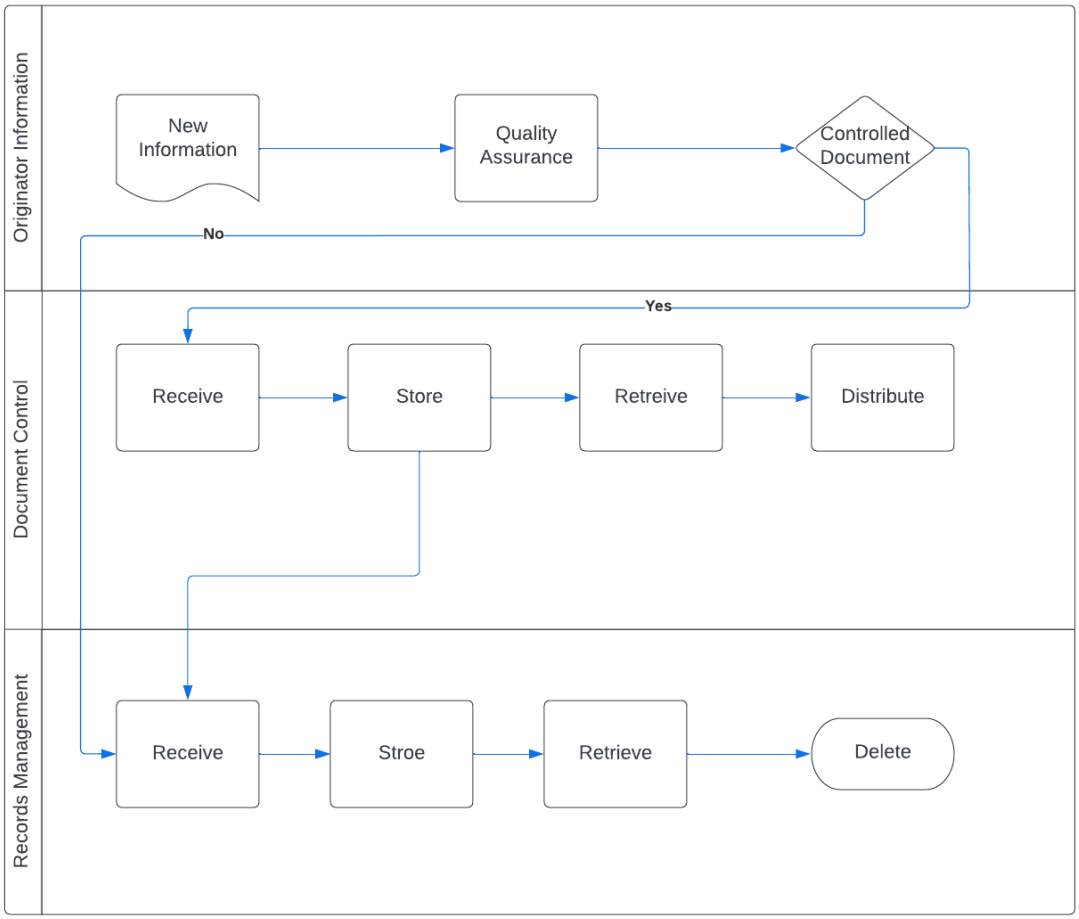
Chapter 2 System Analysis

2.1 System or Software Architecture

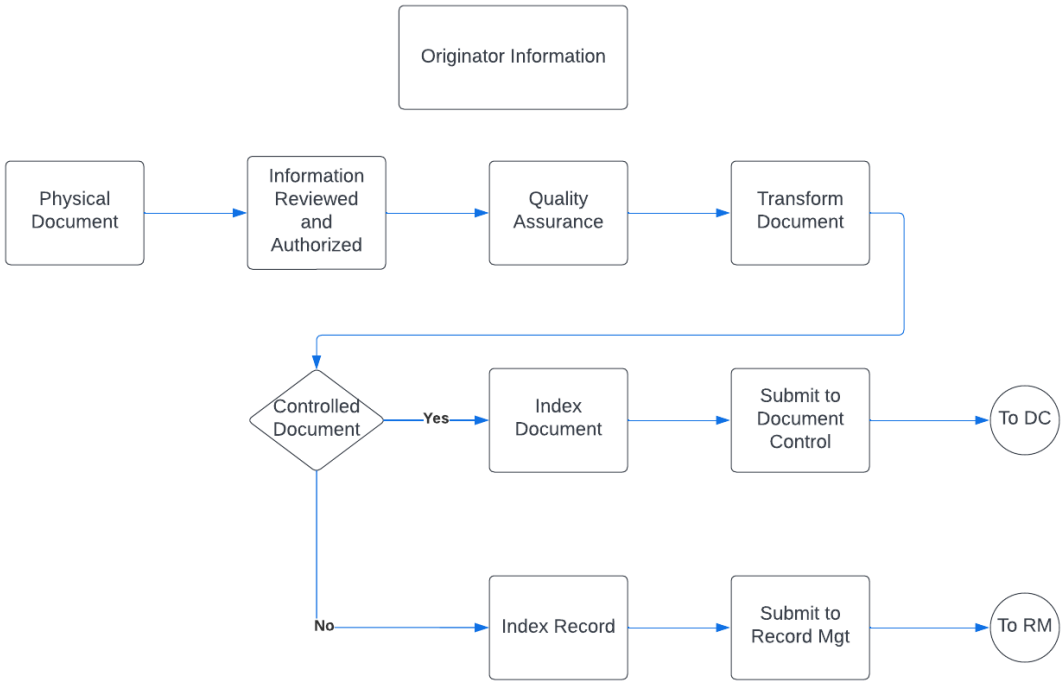


2.2 Flowchart

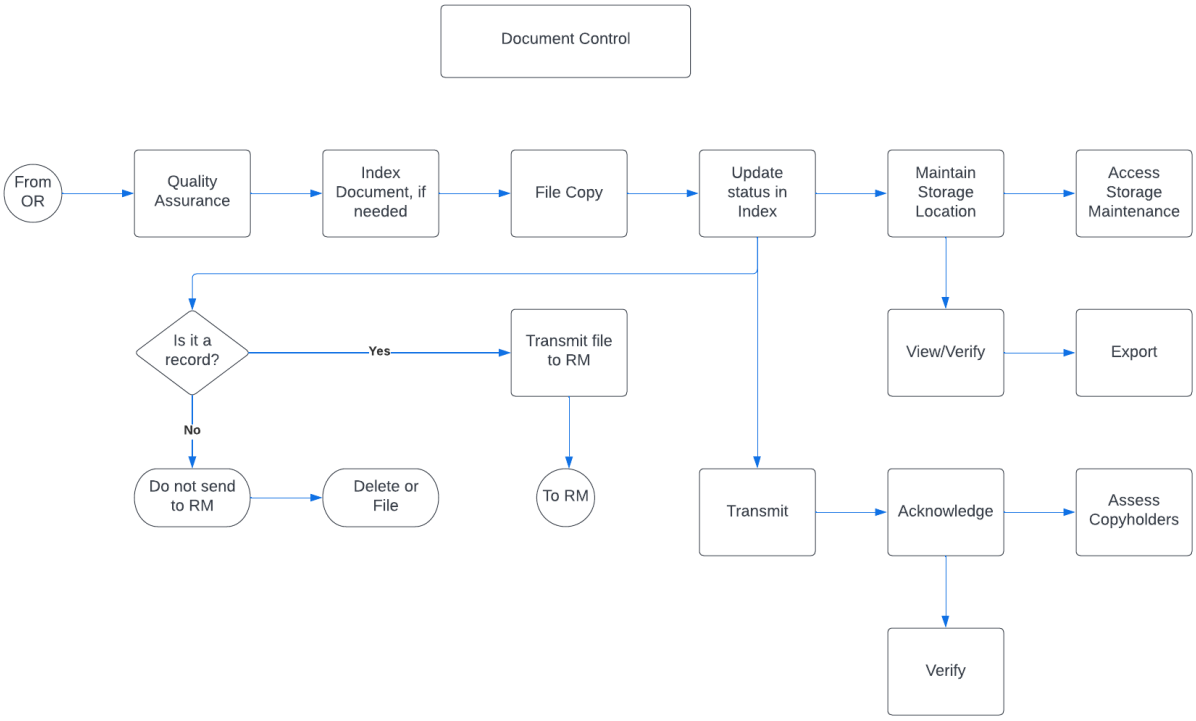
Top-Level Flowchart Document Control and Records Management



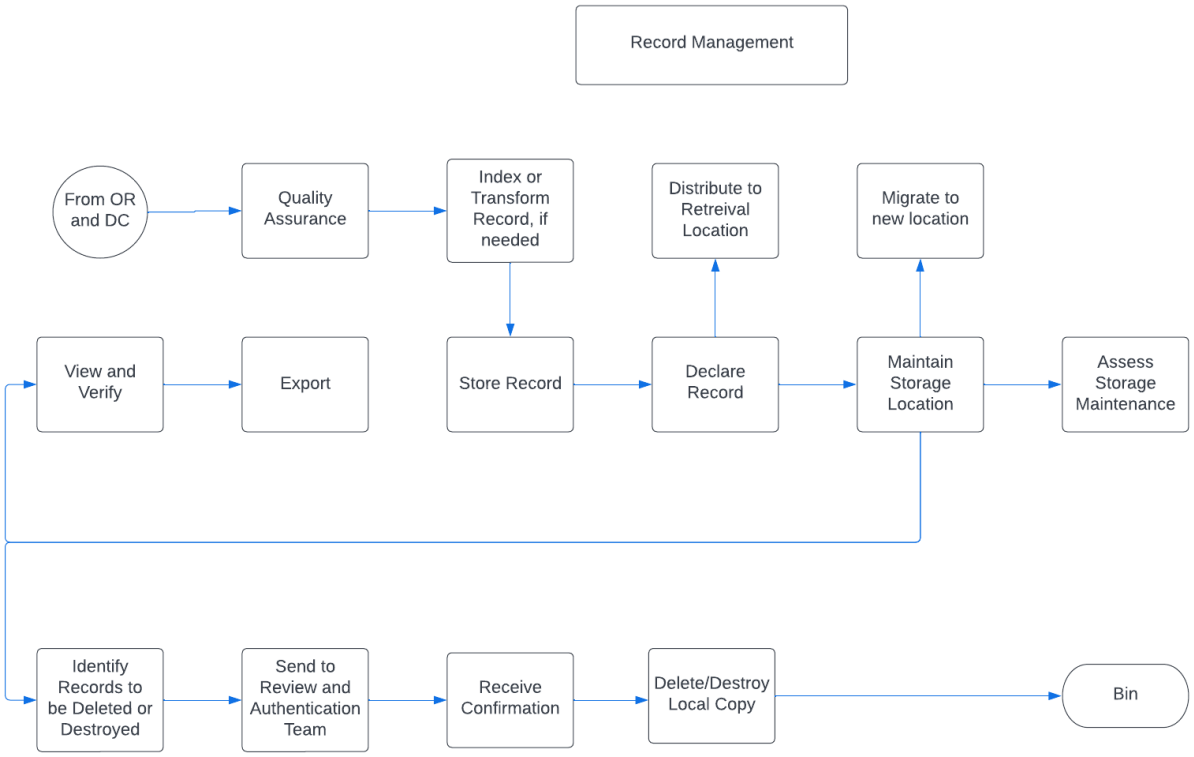
2.2.1 Functionality 1-Originator Flowchart



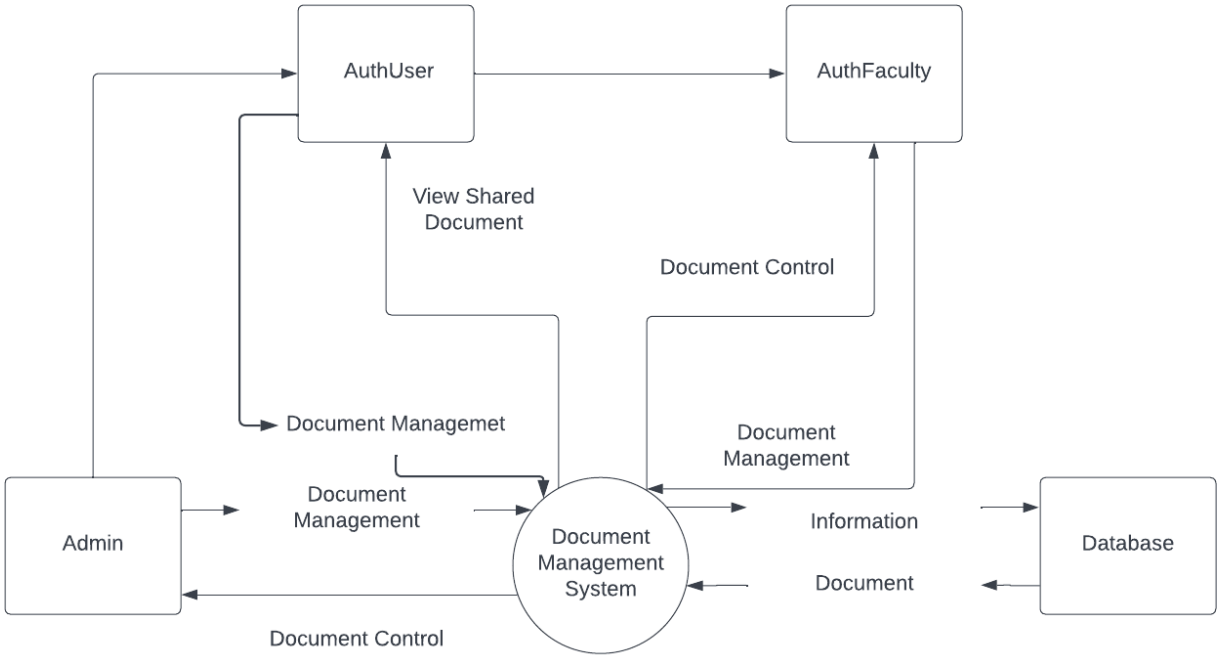
2.2.2 Functionality 2-DC Flowchart



2.2.3 Functionality 3-RM Flowchart

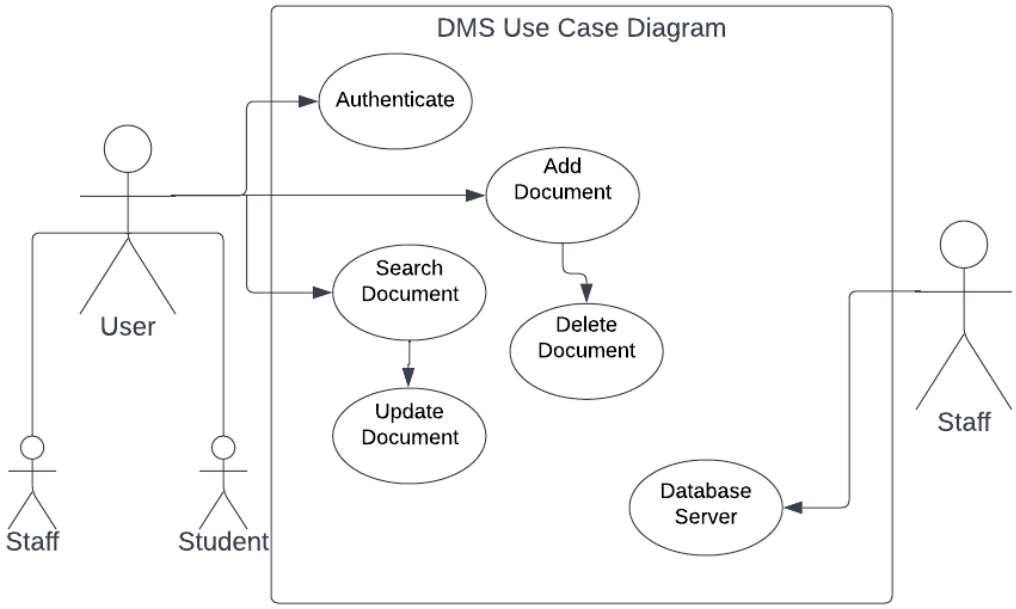


2.3 Data Flow Diagram

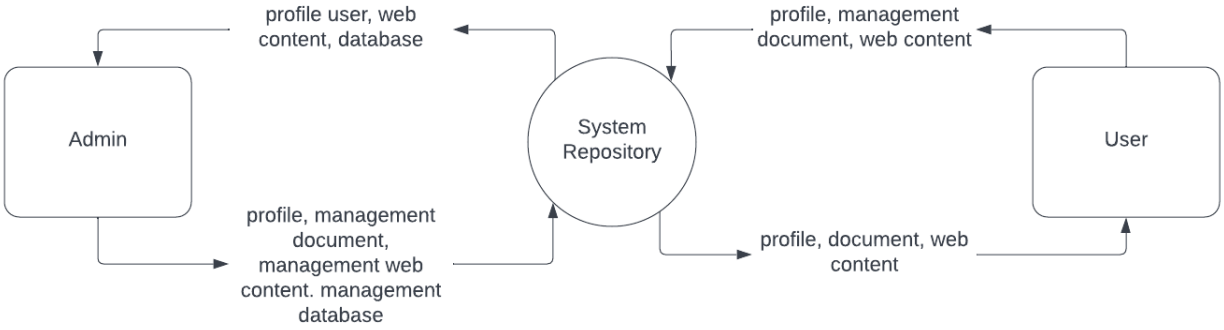


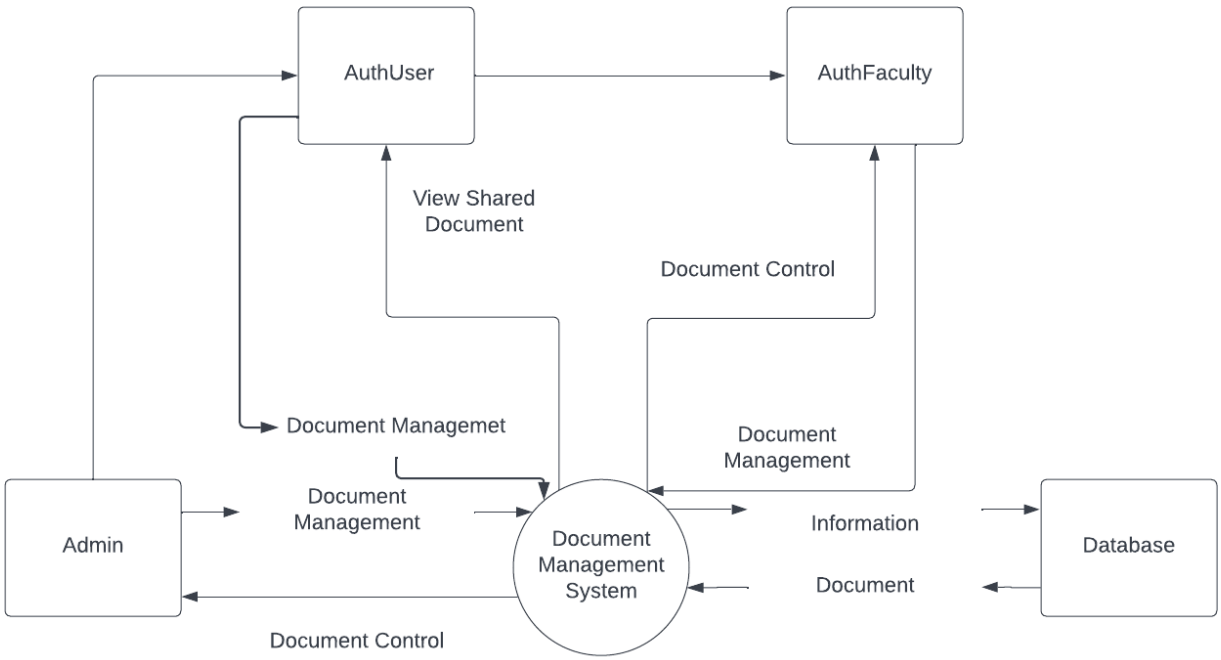
2.4 Entity-Relationship Diagram

2.5 Use Case Diagram



2.6 Data Flow Diagram





2.7 Data Dictionary

Field Name	Data Type	Field Size
id	**Integer**	
password	**Character**	**128**
Is_superuser	**Bool**	
username	**Character**	**150**
last_name	**Character**	**150**
email	**Charater**	**254**
is_staff	**Bool**	
is_active	**Bool**	
first_name	**Character**	**150**
title	**Character**	**250**
description	**Text**	
file_path	**Character**	**100**
session_key	**Character**	**40**

Chapter 3 Functional Requirements

3.1 Functionality 1 – Document Control

3.1.2 The system shall display the documents

3.1.3 The system shall allow users to edit documents

3.1.4 The system shall allow users to update documents description

3.1.5 The system shall allow users to share the document

3.1.6 The system shall allow users to delete documents

3.1.7 The system shall allow the users to download the document

3.1.8 The system shall allow shared document to be downloaded

3.2 Functionality 2 - Provide Search Function

3.2.2 The system shall display document description when sharing it.

3.2.3 The system shall display matching documents based on the search

3.2.4 The system shall enable user to navigate between the search results

3.2.5 The system shall notify the user when no matching document is found on the search

3.3 Functionality 3 – Maintain User Profile

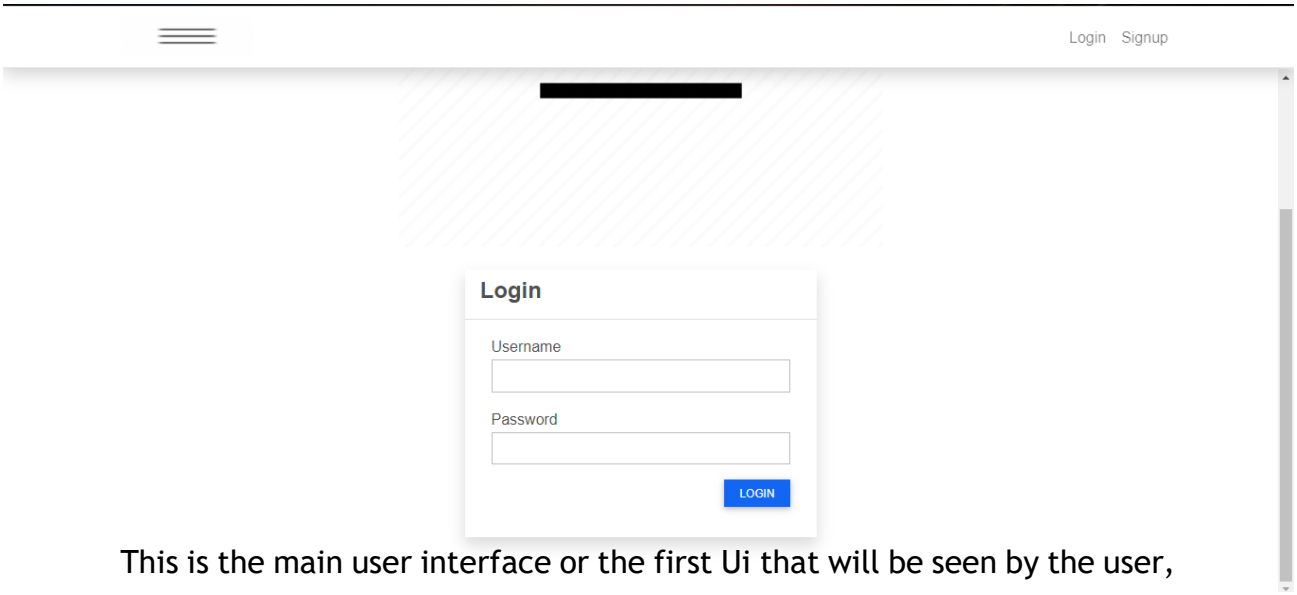
3.3.2 The system shall allow user to create profile and set his credential

3.3.3 The system shall allow user to update the profile information

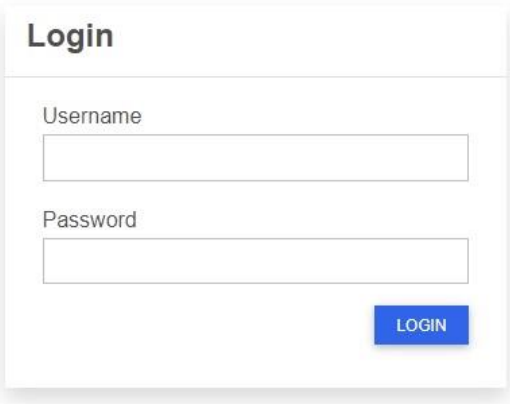
3.3.4 The system shall allow user to register when authorized by appropriate staff

Chapter 4 User Interface/Physical Layout

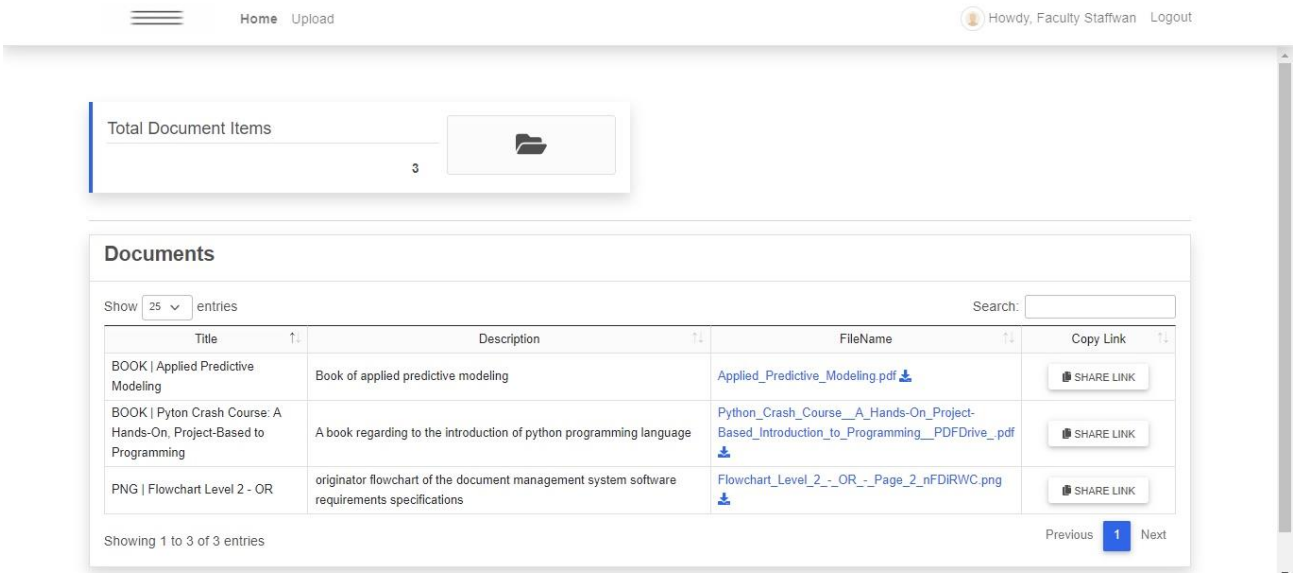
4.1 Main User Interface



This is the main user interface or the first Ui that will be seen by the user, its main functionality is to restrict unauthorized users from accessing the system unless the said subject was approved by the highest authority to register and use the system.



4.2 Control 1 – Main Home Page Menus



After accessing the main user interface, the web will be redirected to the main system wherein it consists of **4 menus** in the **header** and another **4** in the **body.** The leftmost button in the header is named **"home"** wherein it redirects the user to the main dashboard, the 2nd button named **"Upload"** is where the document manipulation functions can be found such as the add, update, and delete of such documents. Next is the **user profile** in the right side alongside the **"logout"** button, by clicking the user profile it will be redirected to the credentials of the users wherein he/she can edit the credentials

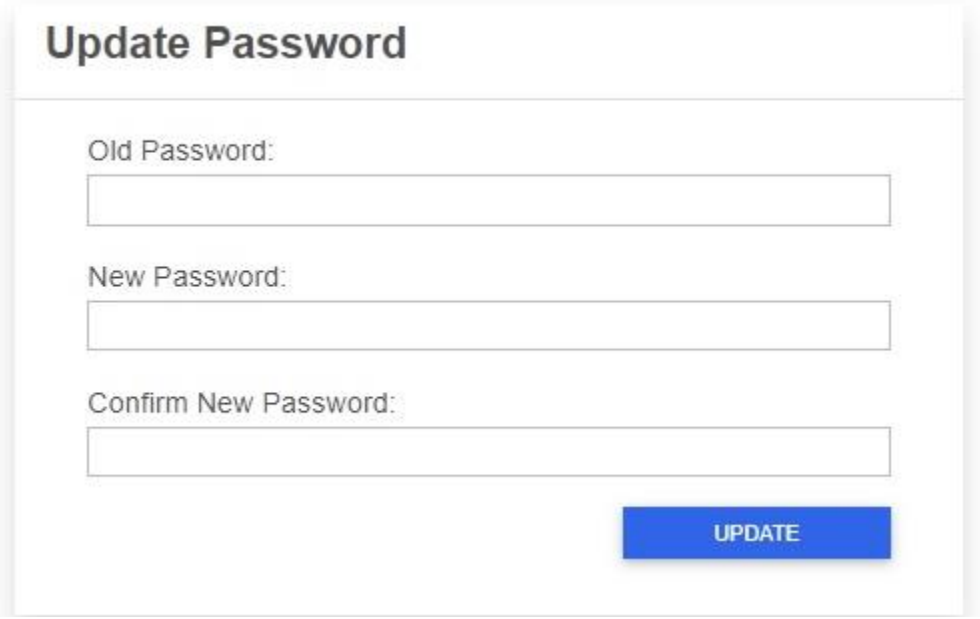


Fig: Edit Password

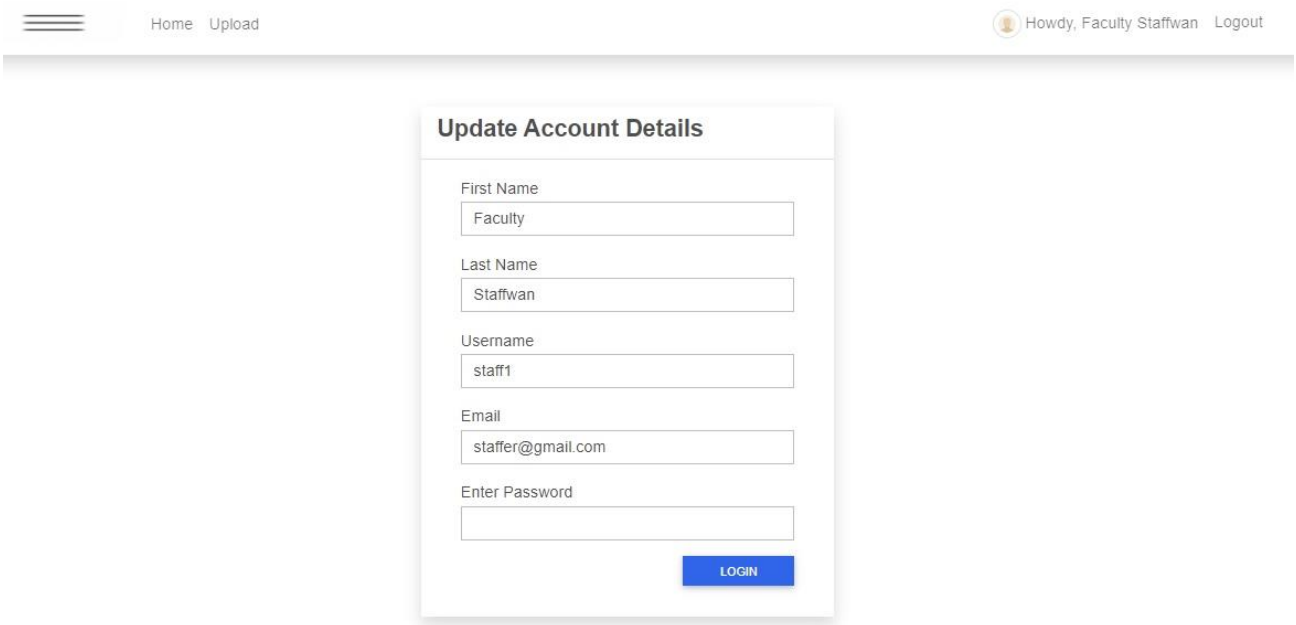
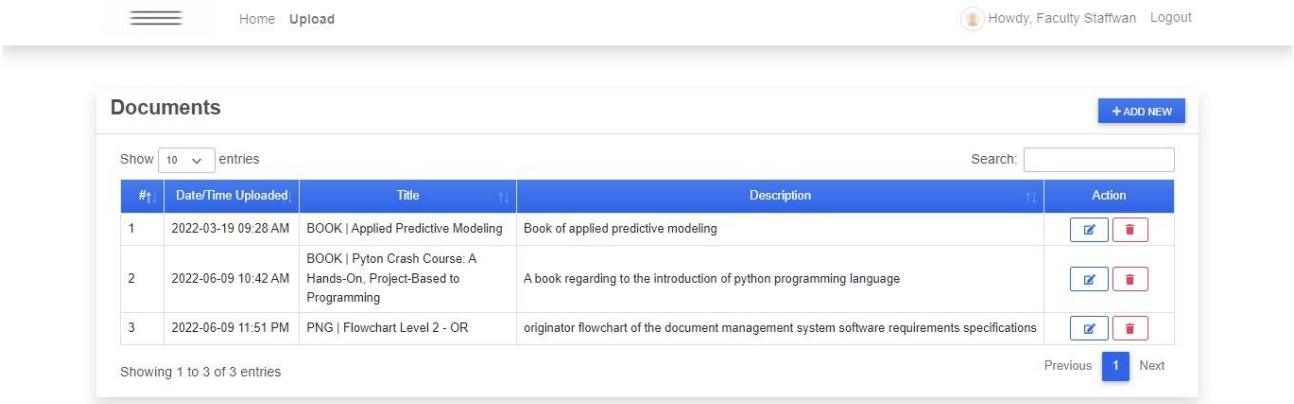


Fig: Edit User Profile Credentials

4.2.1 Sub-controls 1 – Document Management



Consists of **3 buttons** and **1 keystroke. "Add New"** is the button wherein the documents can be uploaded and named unto getting stored into the system repository. After adding a document, the table shows its title, description, and action. In the action table, is where the **"Update"** and **"Delete"** function of the documents is located.

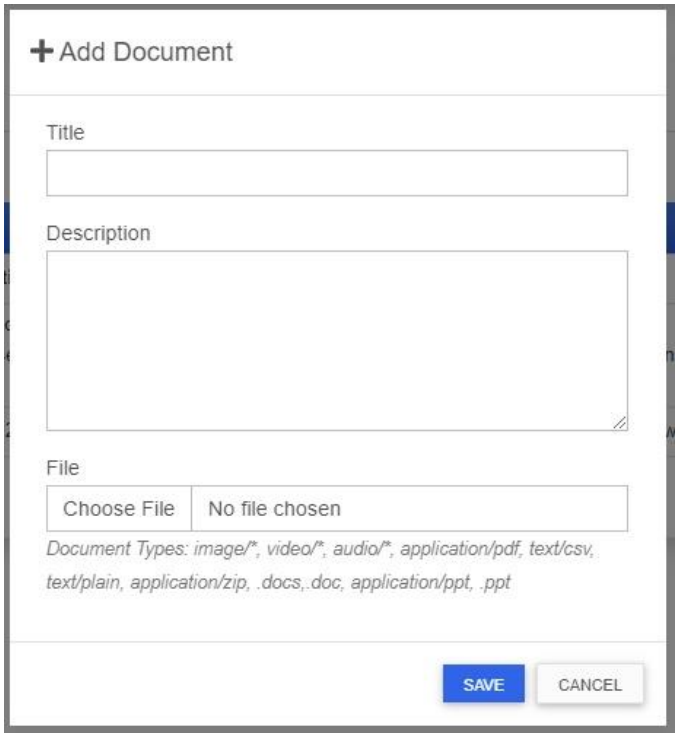
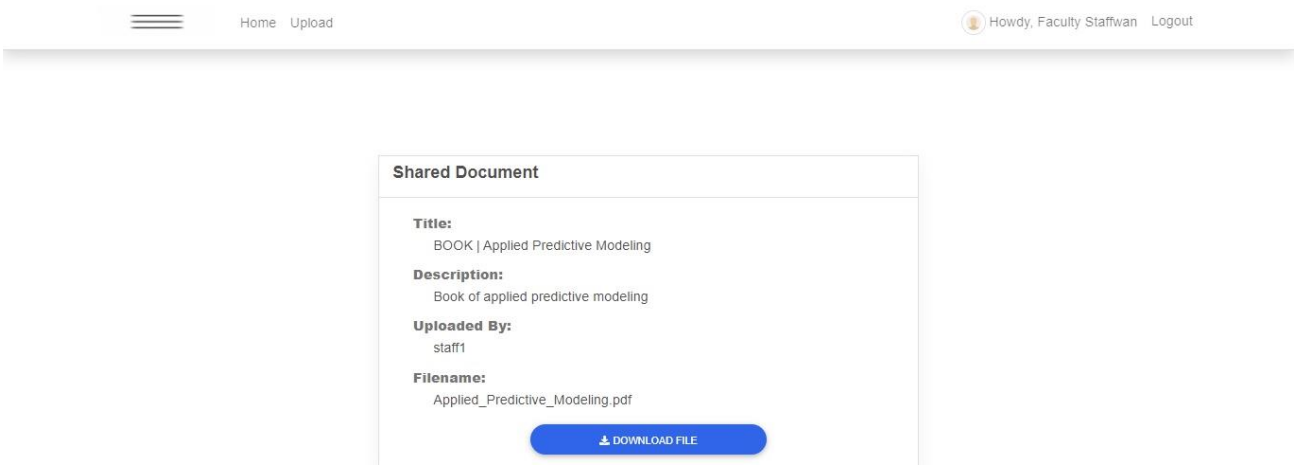


Fig: Add New Document

4.2.2 Sub-controls 2 – Shared Document



It follows the **what-you-see-is-what-you-get** approach wherein the only thing this user or previewer can do is read the information provided and download the document.

Chapter 5 Non-Functional Requirements

5.1 Performance Requirements

5.1.2 Usability

5.1.2.1 Graphical User Interface

5.1.2.1.1 The system shall provide a uniform look and feel between all the web pages.

5.1.2.1.2 The system shall provide use of icons and toolbars.

5.1.2.1.3 The system must produce meaningful error messages that offer a simple method of correcting the error or cancelling the process.

5.1.2.1.4 The system must ensure that a minimal movement between screens is required to perform system operations.

5.1.2.1.5 The system must support navigation and screen interaction by both mouse and keyboard.

5.1.2.1.6 The system must ensure that its facilities are as intuitive and easy to use as possible by allowing:

- Functions tot be performed with as few mouse clicks or keystrokes as possible; and
- The completion of routine functions from one screen

5.1.3 Reliability

5.1.2.2 Back-end Internal Computers

5.1.2.2.1 The system shall provide storage of all databases on redundant computers with automatic switchover.

5.1.2.2.3 The system shall provide RAID V Disk Stripping on all data base storage disks.

5.1.4 The system shall be based on web and can be run from a web server or local server

5.1.5 The system shall depend upon hardware components of the user

5.2 Security Requirements

5.2.1 The system shall confirm all transaction to the system

5.2.2 The system web browser shall never display the user's password. It shall always be echoed with special characters representing typed characters.

5.2.3 The system's back-end servers shall never display the user's password.

5.2.4 The system's back-end servers shall only be accessible to authenticated administrators.

5.2.5 The system's back-end databases shall be encrypted.

5.3 Software Quality Attributes

5.3.1 Performance: time behavior, capacity;

5.3.2 Usability: appropriateness, recognizability, learnability, operability, user error protection, user interface;

5.3.3 Reliability: availability;

5.3.4 Security: confidentiality, authenticity;

5.3.1 Functional Suitability: functional completeness, functional correctness, functional appropriateness;

5.3.1 Maintainability: modularity, reusability, analyzability, modifiability;

5.5 Business Rules

5.3.1 Preservation: The system must be able to manage digital objects according to the minimum mandatory requirements of this document over time, to ensure their integrity and support long-term preservation.

5.3.1.2 The system must support for mapping to and from the system model fields.