Publication Management System

CLIENT SERVER NETWORK ARCHITECTURE COURSE WORK

UP751231

Table of Contents

1. Introduction

Research work is a major cornerstone of any academic department in a higher institution. Academic departments accumulate several types of publications from research papers to journal articles, project reports, etc. all related to several research work being carried out. Over the years, these documents increase in number and become difficult to manage. It is therefore necessary to develop a system to catalogue and manage the publications in an efficient way, while enabling access to different classes of users in a secure and controlled manner.

In the past, this would have required a physical building to house all the publications, with physical cataloguing system. However, with the advent of modern technologies, we can use the principles of a Client/Server architecture to develop a web based replacement for such a system with the paper publications replaced with machine readable files.

This report details the specification of an architecture for a Web based Publication Management system for the School of Engineering called the "Engineering Publication Library" (EPL). It provides all the information on the technologies used, classes of users and what they have access to, database design, considerations made while making the design, etc.

At the end of this report, a developer would have all the information necessary to implement the Engineering Publication Library.

2. Literature review

For several years, different kinds of information has been organized online using Content management systems. A content management system (CMS) offers a way to manage large amounts of web-based information that escapes the burden of coding all of the information into each page in HTML by hand (Eden, 2006).

Digital libraries are a form of content management systems specifically for the content found in libraries. They are sets of electronic resources and associated technical capabilities for creating, searching, and using information (Kresh, 2007).

Library content management systems have particularly been useful in helping organize huge collections of books for easy access. It has made research work much easier for students and researchers alike.

There are several different Online Publication libraries that are currently in use, each with a slightly different implementation of the same overall idea. Two of the most widely used are BibSonomy and Eprints.

2.1 **Eprints**

Eprints is an open source digital repository platform developed and maintained by the University of Southampton. (eprints, 2015)

It provides functionality for hosting large collections of publications in different file formats. Eprints is developed using the LAMP (Linux, Apache, MySQL, and Perl) architecture. (Wikipedia, eprints. 2015). It is used by several universities and other higher institutions to host several types of content. The University of Portsmouth uses Eprints to host "Parade@Portsmouth" (parade@portsmouth. 2015), which is a publications library for dissertation and Theses.

Eprints supports features such as search, listing publications based on categories, user accounts, etc.

2.2 BibSonomy

BibSonomy is a web resource which allows a researcher to categorize and archive both bookmarks and literature references while reviewing a resource of interest in his web browser.

It provides the ability to curate a list of available publication references related to a research work while carrying out literature review.

BibSonomy takes the idea of a publication library system a step further by adding a social component to it. It provides the ability for users to add relevant publications from several sources to the system while providing supporting metadata such as relevant area of research for the publication, tags and keywords, short summary of what the publication is about, etc. (Benz D., et-al. 2010).

This added social functionality improves the overall usability of the publication management system. It also provides an ability for the researchers to actively curate the content available on the platforms they use for research work.

The publication library architecture discussed in this report borrows ideas from both of these implementations, and creates something uniquely suited to the needs of the School of Engineering for which it is designed.

Digital Publication Management systems provide several advantages over physical libraries, some of which are discussed below.

2.3 Advantages of a Digital Publication Management System

- **No physical boundary:** The user of a digital library need not go to the library physically in order to carry out research work. This increases flexibility on the part of the researcher.

- **Round-the-clock availability:** A major advantage of digital libraries is that people from all over the world can gain access to the information at any time, as long as an Internet connection is available. This has enabled long distance learning, access to research materials to the physically impaired, etc.
- **Multiple access:** The same resources can be used at the same time by a number of users. Since the publications are not physical, there is no innate restriction to the number of people that can have access to it at any given time. This resolves the issue of having limited number of copies of a given resource in a physical library.
- **Structured approach:** A digital library provides access to much richer content in a more structured manner. It is therefore easier to find the exact resource being searched for.
- **Information retrieval:** There is flexibility in the use of search terms, i.e., key words to find resources. A digital library can provide user-friendly interfaces for sorting resources in several ways.
- **Space:** A physical library doesn't have any physical space requirements, and in that regard, is unrestricted in the amount of resources it can hold.
- **Networking:** A particular digital library can provide the link to any other resources of other digital libraries very easily; thus a seamlessly integrated resource sharing can be achieved across different libraries with very little effort.
- **Cost:** The cost of maintaining a digital library is lower than that of a traditional library. A traditional library must spend large sums of money paying for staff, book maintenance, rent, and additional books. Digital libraries are not completely without cost though. They incur costs for the conversion of print materials into digital format, for the technical skills of staff, and for the costs of maintaining online access (i.e., servers, bandwidth costs, etc.). However, after initial costs of setup, the digital libraries would cost significantly less that a physical library to run and keep updated (Kresh, D 2007)

3. Designing a Web Based Application

When designing a web based application, there are several concepts that have to be understood, and considerations that have to be made. Most importantly, the objectives of the design work need to be clearly outlined from the start.

3.1 Design Objectives

We need to define the objectives of the Engineering Publication Library. Properly defining the objectives would provide a guideline in making design decisions. At the end of the design, we should have an architecture that fulfils every design objective highlighted here.

The design should state all the different classes of users available in the Engineering Publication Library which include Public users, Basic Registered users, Students, Staff and

Administrators. It should also be able to define all the features available to each class of user, as well as their limitations.

The Engineering Publication Library (EPL) should be able to host several types of publications including Student Project Reports, Masters and PhD theses, Conference Papers, Journal Papers, Research Reports, etc.

The EPL should have the ability to perform search queries, download publications, create user accounts, manage user accounts, upload new publications or delete existing one, block or delete existing users, change user classes, make publications accessible to public users, keep statistics about usage of the system, enforce download quota assigned to the different classes of users, etc.

All this functionality has to be implemented in a way as to ensure that the interface is straight forward and easy to understand, the database is designed in the most efficient manner, and the entire system functions as expected.

Below, some of the concepts to be understood and considerations to be made while executing the design.

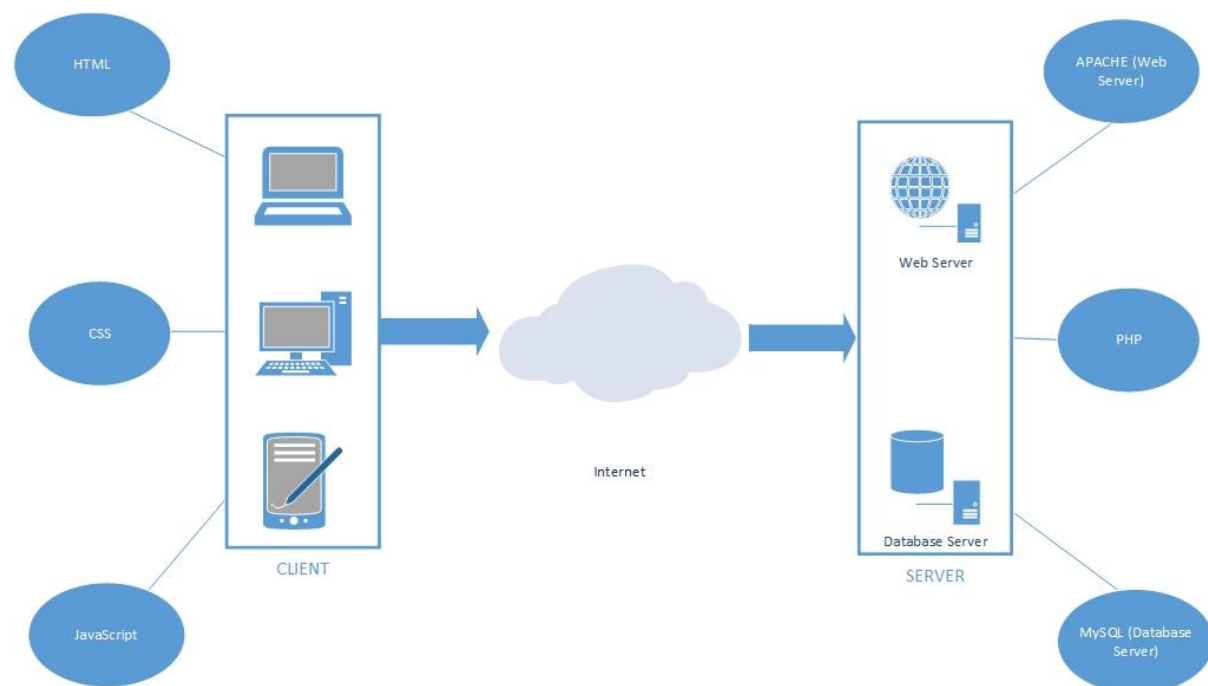## 3.2 Client Server Architecture

Figure 1: Client Server Architecture showing the technologies

Client/Server computing is a technology solution to many data management problems faced by organisations today. It is a computing model for development of computerized systems

that distribute functions between two types of independent and autonomous processes: Server and Client.

According to Yadav, SC, & Singh, SK (2009), A Client is any process that requests specific services from the server process, while a server is a process that provides requested services for the client. Client and Server processes can reside on the same computer or in different computers linked by a network.

A Website is a typical example of a system utilizing the Client/Server Architecture. The Web browser serves as the client, accepts input from the user and sends it to the server for processing. The server then sends the processed response back to web browser, which displays it to the user.

Websites are developed using a range of technologies. There are several languages that can be used, each providing similar functionality in its own unique way.

Web development technologies can be split into Client side and Server side.

3.2.1 Client side

The Web browser is the application on the client side of the client/server web architecture. It supports several technologies which receive data from the web server and structure it in a manner which can be presented to the end user. It also has the ability to do some processing on the client side, as well as send input from the user back to the server for further processing. Some of the main technologies involved in this process are discussed below.

HyperText Markup Language (HTML)
HTML is a markup language for describing web documents. (W3Schools, 1999). Web browsers read HTML and render it to become the web page displayed to the user. HTML forms the basic building blocks of all websites.

Cascading Style Sheets (CSS)
 Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. It is most often used to set the visual style of web pages and user interfaces written in HTML. (Wikipedia, CSS. 2015)

Javascript (JS)
JavaScript is a lightweight, interpreted, object-oriented language with first-class functions, and is best known as the scripting language for Web pages. JavaScript runs on the client side of the web, which can be used to design / program how the web pages behave on the occurrence of an event. (MDN, 2015).

3.2.2 Server Side

Webpages are stored on servers, and are requested by the Web Browser. When the browser requests for the webpage, that request is fulfilled by a Web server. Most websites

require different kinds of data to be functional. This data is stored in a database server. A server side scripting language is also required to process requests on the server and retrieve data from the database.

In this project, Apache is used as web server, MySQL as database server and PHP as server side scripting language.

Apache

The Apache HTTP Server Project is an open-source HTTP (HyperText Transmission Protocol) server for modern operating systems including UNIX and Windows NT. It provides a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards (Apache, 2015).The Apache server receives HTTP requests from clients, processes it and sends the appropriate response back.

MySQL

MySQL is a fast and powerful, yet easy-to-use, database system that offers just about anything a website would need in order to find and serve up data to browsers (Nixon, 2014). It supports several types of variables and can be queried for information stored within it.

Hypertext Pre-processor (PHP)
PHP is the language that you use to make the server generates dynamic output that is potentially different each time a browser requests a page.

3.3 Design Considerations

Several considerations have to be made when designing any application. There are some considerations unique to web based applications. These factors have to be considered to ensure that the final application not only functions as expected, but will continue to function in the long term and is able to support changes that will inevitably be added over the lifespan of the system. Security is increasingly becoming an important factor for any networked system so, this has to be considered. For our Electronics Publication Library (EPL), we need to consider some important factors in making design decisions.

3.3.1 Maintenance and Management

In order to ensure that our system not only works, but is useable for a long period of time, we need to ensure that it is easy to maintain and manage. This consideration informs several decisions made during the design process. In designing the EPL, we have chosen to use popular open source web technologies such as PHP, JavaScript, HTML, etc. These are technologies that are very popular and known by a lot of people. This ensures that our system can be debugged and expanded upon by others without needing people with specialized expertise.

A conscious decision also has to be made to design the database in a way that is in accordance with accepted best practises. This makes it easy to understand how the data for the entire site is stored, even by those who were not part of the original development. The interface of the EPL has to be designed with Maintenance and management in mind. The Admin interface should be designed to be as simple as possible, while providing the necessary tools an administrator would need. This is necessary to ensure that anyone can become and administrator for the EPL website without needing to be a web developer.

3.3.2 Hosting

Web hosting is a very important aspect to consider when designing a web based system. It is important to use technologies that are widely supported in the industry in order to have a wide range of options when deciding on a web hosting service. The EPL is designed using the LAMP (Linux, Apache, MySQL, and PHP) stack, which is supported by almost all web hosting companies. This provides an incredible range of choices for hosting the website. Other factors to be considered when making a decision on hosting are on the amount of storage, memory and bandwidth required. Most web hosting companies provide several tiers of subscriptions with varying amounts of the 3 requirements above, so those have to be considered carefully.

Another option that provides more freedom but requires more work is to use a Virtual Private Server (VPS) solution. A VPS vendor provides a server with a discrete set of specifications. These servers are usually bare, and require administration to install all required server software and do all the configuration. This method is much more complex than regular hosting, but it provides more control to the developer, especially if the application is developed using non-standard tools and technologies.

The EPL will be hosted on a shared hosting platform similar to those found on commercial offerings. It can be hosted on any of the top web hosting providers, with minimal cost. Taking a survey of the top two web hosting providers, Hostgator.com provides its "Business" hosting plan at $115 per year while "Godaddy.com" provides its "Ultimate" hosting plan at £84 per year.

3.3.3 Portability

It is most likely that over the lifespan of the EPL, it might need to be moved to different servers and/or domains. This means that the website has to be designed with portability in mind. The most important factor to consider for portability is the way the links are structured in the code. The website is a collection of files that are mostly stored in the same location, with some exceptions. Since these files are within the same file structure, when calling them within the code, it is prudent to use the relative file path instead of the canonical, or full file path which involves the domain name.

The issue of portability doesn't only occur as a result of a change in domain name. Some files or assets in the EPL application might be moved to a different subfolder within the same server, or the existing folder could be renamed. In order to prevent these changes from breaking file links within the application, all files should be referenced using their relative file path.

Another consideration to be made for portability is the technologies used. We have to use technologies that is most commonly available among major hosting providers. This provides the ability to move the website between service providers easily without needing to seek special functionality. Most servers in the world run the popular Apache Open source Web server. It would therefore be harmful to the portability of the website if it was developed with a technology such as ASP which is only supported by Microsoft's IIS Web server. This would hinder the ability to move the website easily from one service provider to another, hence hindering portability.

3.3.4 Upgrading

Over the years, as the needs of the department evolves, changes might need to be made to the EPL. The Website has to be designed in such a way as to allow new features to be implemented without having to make drastic changes to the existing functionality. The use of standard web technologies makes it possible for these changes to be made by almost any one.

The design of the EPL has to abstract the data away from the website as much as possible. This makes it possible to change the entire website while still retaining the same database. Upgrading is a very important consideration to be made while developing a website as it ensures that the website doesn't become outdated and unusable after just a few years.

3.3.5 Security

Security is a very important consideration in developing websites. It is even more apparent today, as more and more organisations have their websites hacked into, and data stolen. Important considerations have to be made during design to ensure a secure product. The user password has to be stored in the database as a "salted Hash". This means that nobody can recover the original password, including the administrators, and authentication is only done by comparing the stored hash with the hash of the password provided by the user at login. This is an effective way of preventing a malicious people or programs from gaining access to an account by sniffing packets.

Another important security measure is to mitigate SQL Injections. SQL Injections are a very popular security vulnerability in websites where SQL commands can be injected as regular text input which are then inadvertently ran as SQL queries and display sensitive data to end users without appropriate access. There are several popular and effective ways to mitigate

this vulnerability such as reducing the permissions of the database account used by the website, escaping words that are special to SQL, etc. These techniques will be used to ensure that the EPL is not susceptible to SQL injection.

Another way to ensure security of the website is to enforce rules on passwords. Enforcing certain rules on passwords upon user creation will ensure that the passwords used on the system meet a minimum level of complexity which would make them difficult to bruteforce. There are several other techniques that can be used to secure the website, however, security has to be balanced with functionality and ease of use in order to provide the best end user experience.

4. System function analysis

The Engineering Publication Library should be able to perform several functions. Some of those functions would only be available to specific classes of users, while others would be available to the general public. Below, the functions of the system are expanded upon.

4.1 Host Academic Publications

The primary function of the EPL is to be a Library of Academic publications for the Engineering department. This means that it should have an index of all publications created by both staff and students of the department. These publications can be added by both staff and administrators of the EPL, however, they can only be removed by the administrators.

4.2 Robust Search

In order to make the library accessible, there has to be a robust search functionality implemented. This involves searches using keywords such as the name of the publication, names of authors, date of publication, etc. There would also be an "Advanced Search" function that provides the ability to further streamline the search criteria.
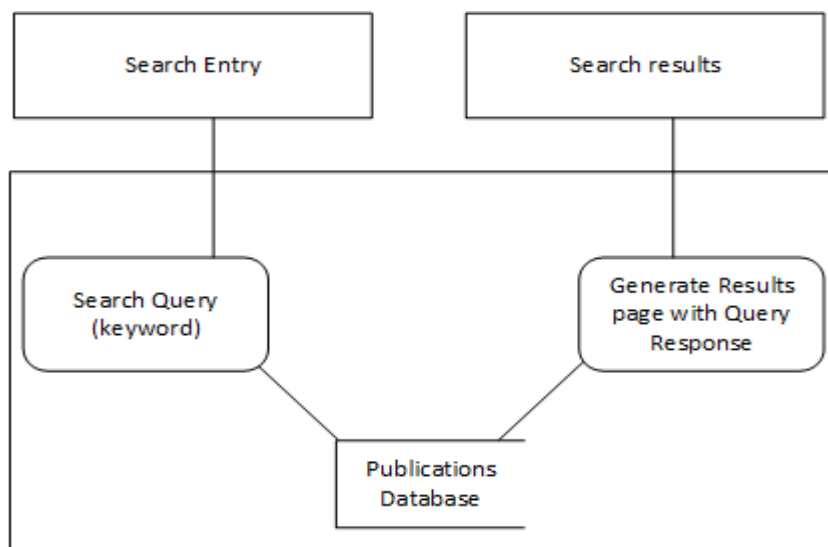

Figure 2: Information Flow Diagram showing the search functionality in the EPL

4.3 User creation/Management

Visitors to the EPL would have the ability to create user accounts, login to those accounts and make changes to their user profile. A new user account created will be on the "Registered user" class, which can be elevated by the administrator.
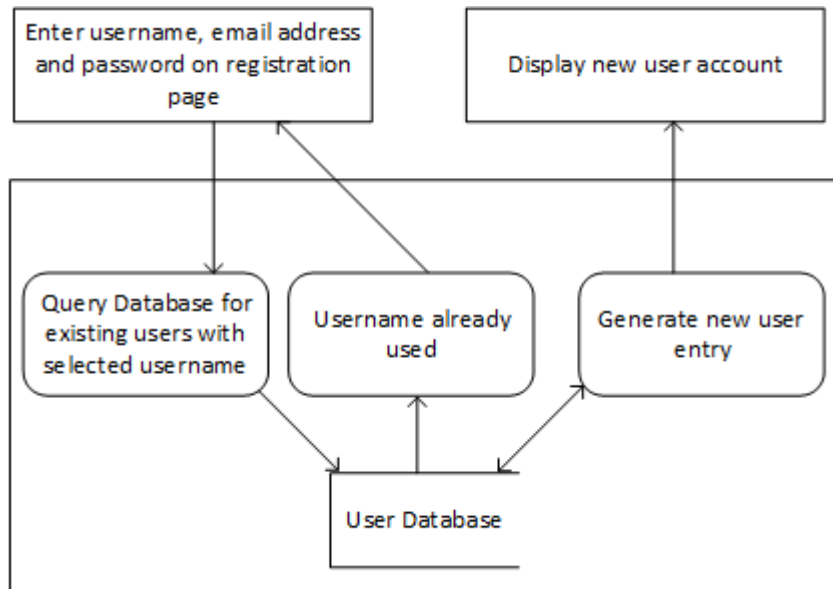
Figure 3: Information Flow Diagram showing the account creation feature of the EPL

4.4 Category listing

To further aid discoverability, the publications should be displayable using categories such as Author names, Year published, range of date, etc. This provides the ability to find publications without knowing much information about it. It also enables the ability to filter results down to specific criteria.

Figure 4: IFD showing the "List by Category" functionality of the EPL

4.5 Download Publications

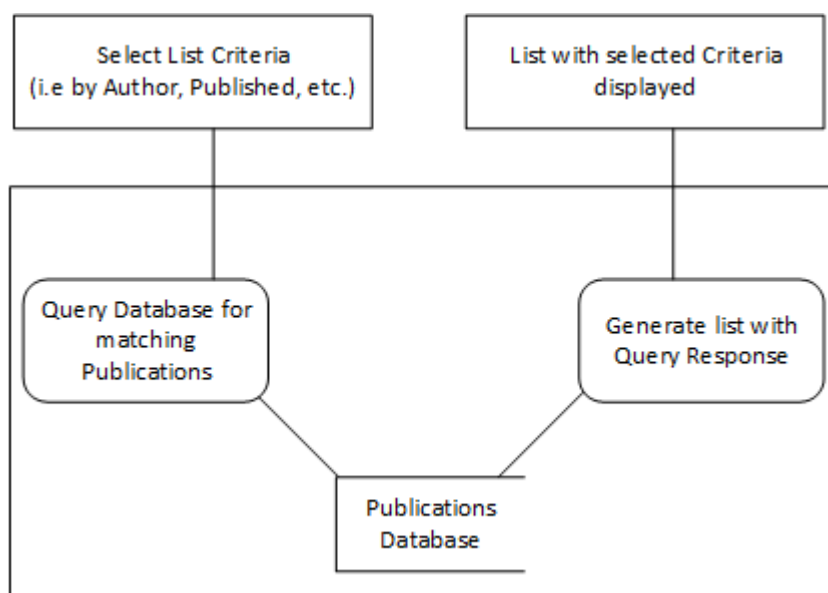One of the major features of the EPL is to allow downloading of Publications. This feature can be carried out by all classes of users on the system, with varying levels of restrictions. Unregistered (Public) users can only download publications that have been specifically made accessible to the public by Staff or Administrator. Staff and students can download any document but are restricted by the number of downloads they can make within a calendar month. Administrators have no restrictions on what or how many times they can download.

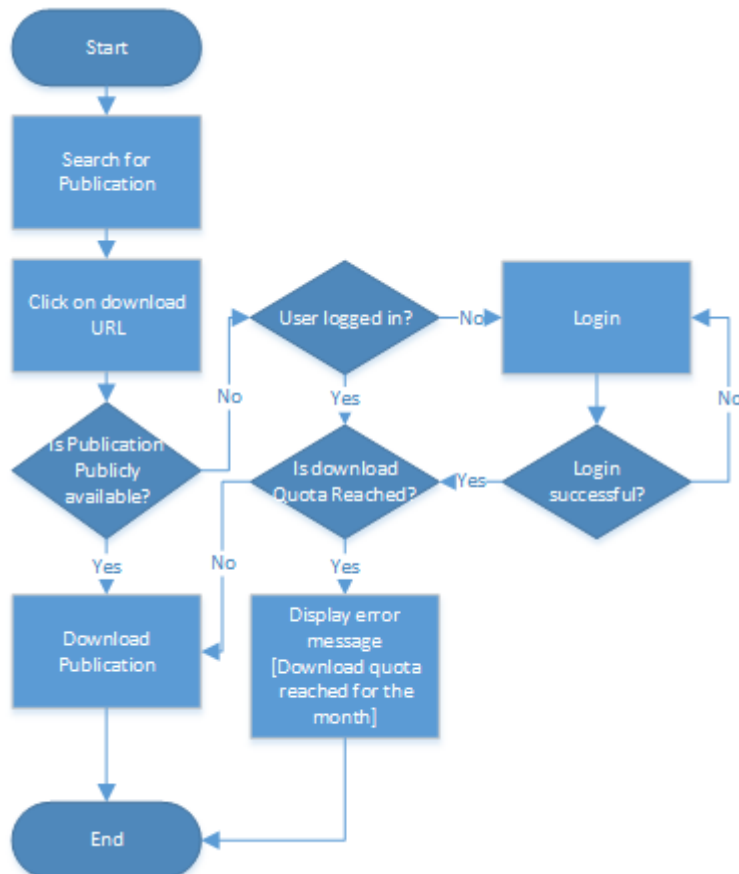*Figure 5: Flowchart showing the ability to set make publications publicly available for download.*

4.6 Administrator functions

The administrator has a unique set of functions which enable him to have total control of the system and other user accounts. These functions can be found on the Admin panel interface which is only accessible to accounts on the class of Administrator. These admin functions are listed below.

Change account levels

By default, when a user creates an account, it is designated as a "Registered user" account. This means that all restrictions to the registered user class of accounts would apply to that account. This account can be elevated to a higher class by the administrator.

Block or delete users

An administrator can block user accounts temporarily. This would prevent that user from logging into the system. The Admin can also delete user accounts completely. This could be done when a user graduates from the University for example.
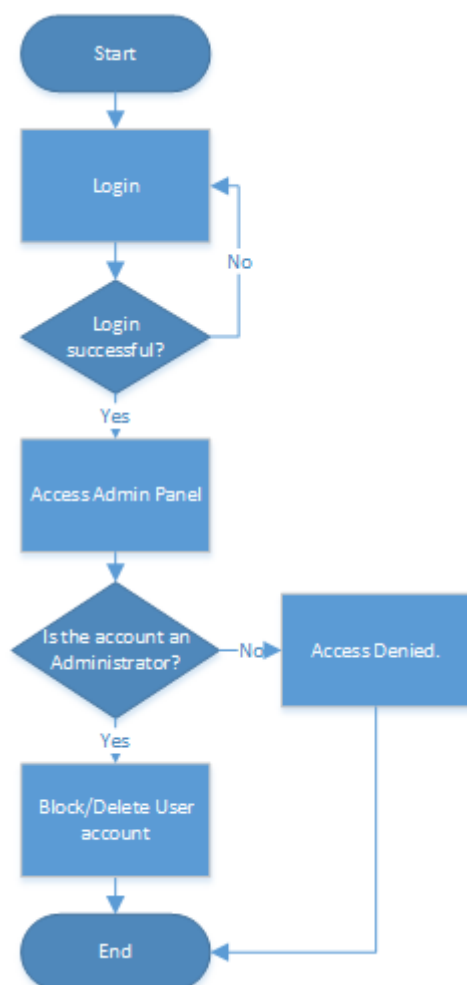
Figure 6: Flowchart showing the Admin Account deletion/blocking feature of EPL

Change restrictions on classes of accounts

The different classes of accounts are restricted to a fixed maximum number of downloads per calendar month. This number can be increased or decreased for the different classes of accounts by the administrator.

Upload/Remove library publication

A library publication can be added by Staff and Administrator, however, only the administrator has the ability to delete publication already in the library.

Figure 7: Information Flow Diagram showing the new publication upload admin feature of EPL

4.7 Statistics

The EPL will keep records of important data such as user logins, number of publications downloaded by a user per month, overall website usage statistics, popular search terms, etc. These statistics can be used to implement limits placed on specific classes of users such as disabling the ability to download publications after a user has exceeded his monthly limit.
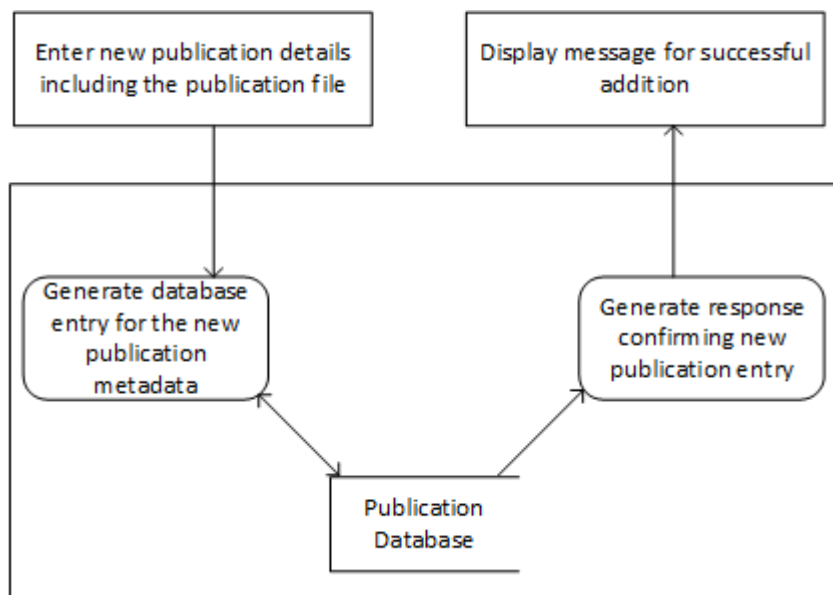
These are the major features of the EPL, and they will be used to guide our design process to ensure that at the end, we have a website architecture that supports all the above features, and more.

5. User Identification and Requirement Analysis

The EPL will be used by different classes of users. While discussing the System functions, some of those classes have already been mentioned. Each class of user would have access to a subset of the overall features of the Library. Below, we need to explore the different classes of users available and what each of the users will be able to do on the system.

5.1 Unregistered users (Public)

The EPL website will be available to the general public, however, it will have very limited functionality. The General Public will be able to do the following:

- Access the website
- Search for Publications

- Download publications made available to the Public
- Register for a user account

Figure 8: Use-case diagram for an Unregistered User in the EPL

In general, publications cannot be downloaded by unregistered users, however, certain publications can be made accessible to the public by EPL Staff with Administrator accounts.

5.2 Basic

"Basic" class is assigned to any user account that does not belong to a student, staff or Administrator. This could be any member of the public that registers for an account on the website, former students or past member of staff etc. This class of users have the least level of access among the registered users. They have ability to download any publications in the library, however, are restricted to just 10 publications per calendar month. Users in this class can have their account elevated to a higher class by the Administrator.
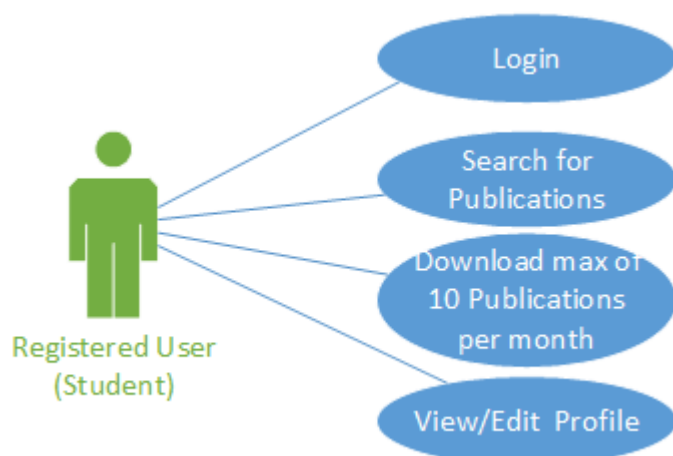
Figure 9: Use-case diagram for a Basic Registered User in the EPL

5.3 Student

Students have the ability to login with a unique set of credentials to the EPL website. Once logged in, they would have a greater level of access to the Library, but still have certain restrictions. Students can do the following:

- Access the website
- Login with their student credentials
- Search for Publications
- Download maximum of 20 publication per calendar month
- View/Edit personal information
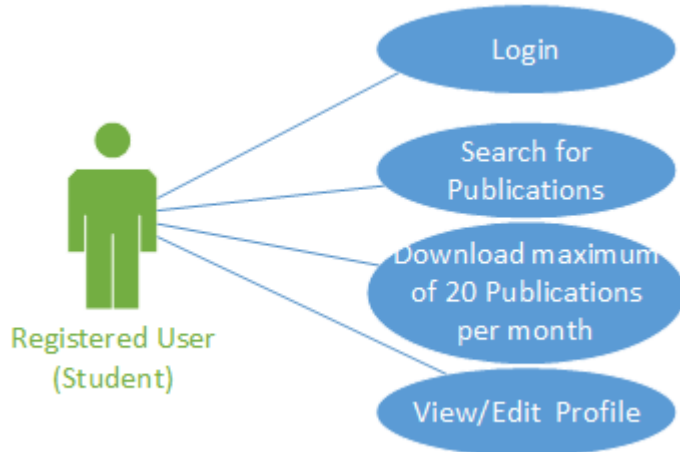
Figure 10: *Use-case diagram for Student Users in the EPL*

5.3 Staff

Staff have the ability to login with a unique set of credentials to the EPL website. Once logged in, they would have a greater level of access to the Library than Students, but still have certain restrictions. Staff can do the following:

- Access the website
- Login with their student credentials
- Search for Publications
- Download maximum of 30 Publications per calendar month
- View/Edit personal information
- Upload Publications
- Make publications publicly available
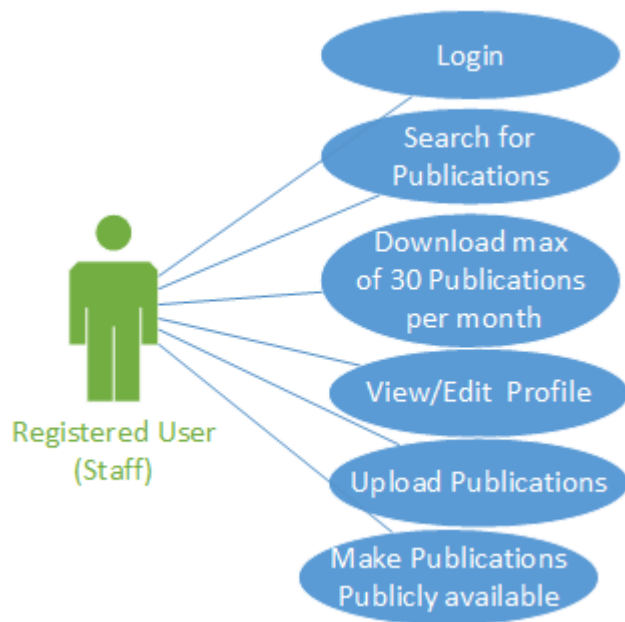
Figure 11: *Use-case diagram for Staff Users in the EPL*

5.4 Administrators

Administrators have the highest level of access of all users on the site. The Administrator account does not have any restrictions on the number of downloads that can be made from the EPL. In addition to all the abilities of all other users, The EPL staff can also do the following:

- Search for Publications
- Login
- Download Publications
- Block/Delete existing user accounts
- View overall usage statistics for the website
- Grant download access to specific publications to the unregistered users
- Upload new publications to the database
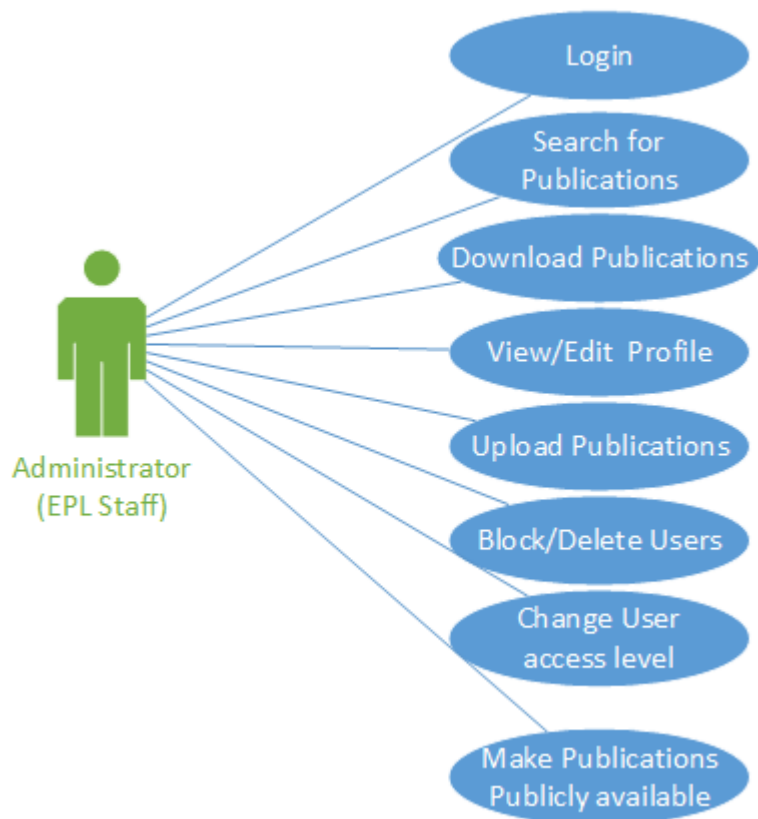
Figure 12: *Use-case diagram for Administrators of the EPL*
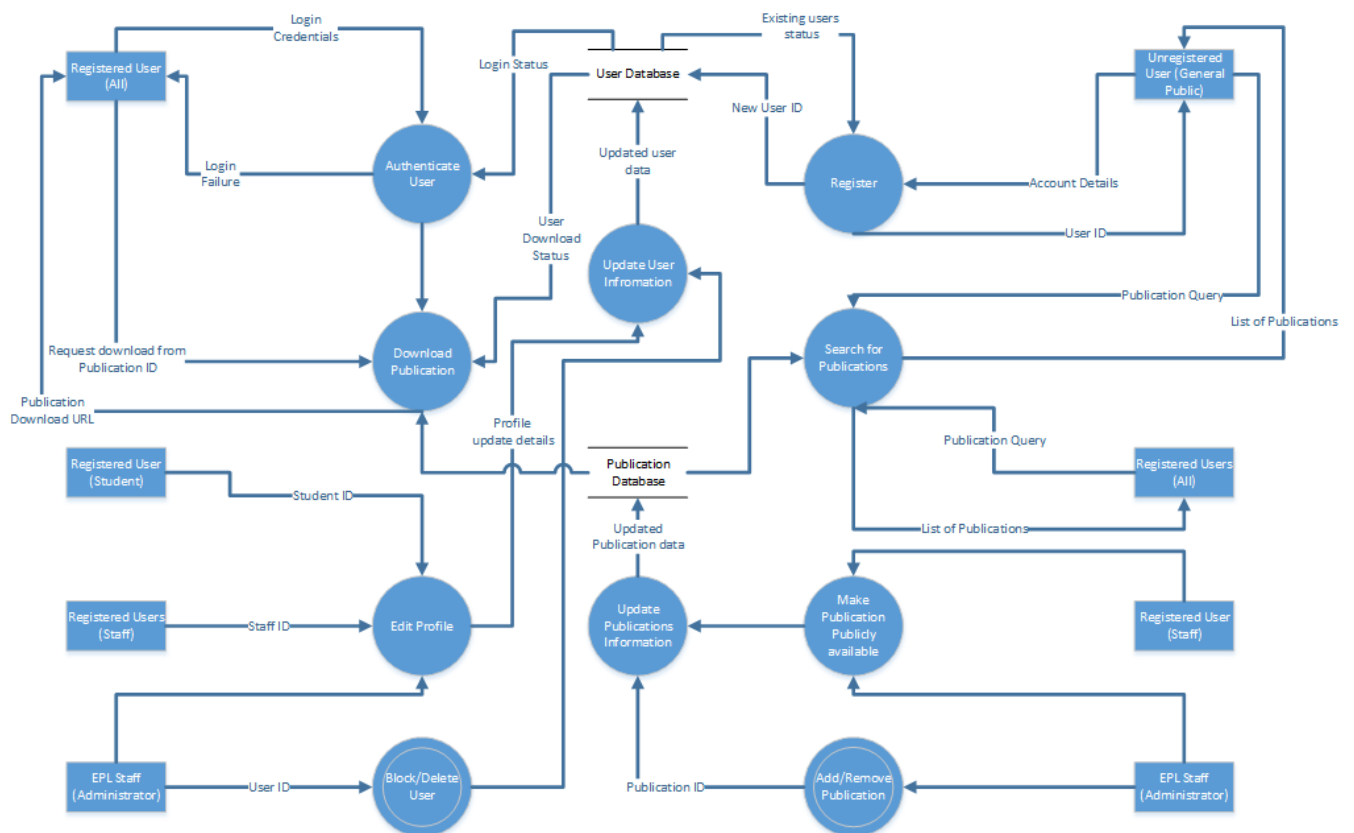
Figure 13: *Data Flow Diagram of the Engineering Publication Library*

6. Database requirements analysis and design

The EPL website needs to manage a large amount of data with different requirements such as needing to be accessed more frequently, requiring isolation from other data for security reasons, etc.

As a result of this, these data have to be assigned to tables on the database in a manner that minimizes the number of queries required to retrieve them when needed. Before designing the database, we need to perform a requirements analysis in order to determine the data needed by the system.

6.1 Database Requirement Analysis

The requirement analysis is done by reviewing the different data required by both users and publications in the system.

6.1.1 Users

Data required by for users

Names

- Title,
- First Name
- Middle Name
- Last name

Address

- House Number
- Street
- Town
- City
- County/State/Province
- Postcode
- Country

Email address

Date of Birth

Privilege level (depending on the type of user)

Username

Password

6.1.2 Publications

The EPL will host different types of content, with each type requiring a different set of information to be stored in the database. Below, the different types of content are listed along with the data required by each type.

Student Project reports

Student project reports are publications written by undergraduate students and usually submitted as part of the requirements for being awarded a degree by the institution. These publications are usually only published internally within the institution. The attributes of Student Projects reports are as follows:

- Title of Report
- Name of supervisor
- email address of supervisor
- Name of student Author(s)
- Address of student(s)
- Phone number of student(s)
- email address of student(s)
- date of publication
- date added to EPL

Masters and PhD Theses

These reports are also similar to the undergraduate student project reports. They are written by Post graduate students and are usually only published internally. As a result, the attributes attached to Masters and PhD theses are identical to those of the Student Project Reports.

Conference papers

Conference papers are publications written to be presented at an academic conference. These types of publications have a different set of attributes which are listed below

- Title of Paper
- Paper Abstract
- Author(s)
- Address of Author(s)
- Email address of author(s)
- Phone number of authors(s)
- Details of each author including institutional affiliation, email and postal address
- Title of conference
- Address of conference location
- date of conference

- date added to EPL

Journal papers

Journal papers are academic research papers written for publication in an academic journal. As a result, Journal papers published in the EPL would have also been published in certain journal publications. This means that they will have a unique list of attributes as listed below

- Title of Paper
- Paper Abstract
- Author(s)
- Details of each author(s) including institutional affiliation, email and postal address and phone number
- Title of Journal
- Publisher
- range of pages for the paper in the Journal
- Date of publication
- date added to EPL
- Issue number

Research Reports

A Research Report is an academic report written to fully describe the processes involved in carrying out a research work, as well as the observed results. Research reports are usually much more comprehensive than other forms of publications listed above.

The attributes of research reports are listed below

- Title
- Abstract
- Author(s)
- Details of each author(s) including institutional affiliation, email and postal address and phone number
- Duration of research
- date of publication
- date added to EPL

Other Reports

This is a category reserved for publications that do not meet any of the other criteria. These publications would require extra data to specify what kind of publication it is.

The attributes are listed below.

- Title
- Abstract
- Author(s)

- Details of each author(s) including institutional affiliation, email and postal address and phone number
- Type of publication
- date of publication
- date added to EPL

All the above listed attributes have to be entered into a database, and be retrievable when needed. In order to achieve this, the database has to be designed specifically with these data types in mind. Below is an approximation of the database table layout that can be used for the EPL publications

6.2 **Database Design**

The database of the EPL has been designed to provide the ability to store all data related to the different kinds of publications as well as the data for all users on the system. The system has been designed to eliminate duplication of data across all the tables in the database. This is achieved by linking different tables together with both primary and foreign keys.

Below, the different tables, columns and data types are discussed, as well as the reason for the design.

There are two major classes of data to be stored in the database.
- Publication data
- User account data

6.2.1 Publication data

Publication data are all the information required by publications stored in the EPL. This is used to uniquely identify each publication, as well as enable the ability to properly search through the library to find relevant information. As a result, all data related to publications need to be stored in the database in a way that eliminates duplication of data and allows for easy searching.

The design for the publications database is as follows

Publications Table (pub)

This table contains all the data that is general to all types of publications. It is also the primary table for the publication content and its ID serves as the primary key for obtaining other content from the other tables.

Column	Data Type	Description
pub_id	INT	Primary key for the Publications
title	VARCHAR(100)	Title of the publication
abstract	VARCHAR(500)	Short abstract/description of the publication

url	VARCHAR(50)	Download link for the publication
publisher	VARCHAR(50)	Publisher of the publication (i.e conference, journal, institution, etc.
isbn	VARCHAR(20)	ISBN number of the publication (if any)
date	DATE	Date of publication
date_added	DATE	Date publication was added to EPL
public	TINYINT(1)	This is a Boolean value that is set to enable or disable a publication from public access

Table 1: Showing the column assignment and data type for the "Publications" table.

Conference paper table (conf)

This table contains entries that are unique to conference paper publications. Data is only entered into this table when creating a new entry for conference paper. The data in this column can be identified by the Publication ID (pub_id) serving as the Foreign Key.

Column	**Data Type**	**Description**
conf_id	INT	ID for the conference paper entry, serving as the Primary Key
address	VARCHAR(250)	Address of the conference where the paper was presented
pub_id	INT	Publication ID from the "PUBLICATION" Table serving as the Foreign Key

Table 2: Showing the column assignment and data type for the "Conference paper" table.

Student report table (stdr)

This table contains entries that are unique to "Student Report" type of publications. Data is only entered into this table when creating a new entry for Student reports. The data in this column can be identified by the Publication ID (pub_id) serving as the Foreign Key.

Column	**Data Type**	**Description**
stdr_id	INT	Primary key for student reports
supervisor	VARCHAR(75)	Name of supervisor for the student report
supervisor_email	VARCHAR(35)	email address of supervisor for the student report
student_degree	VARCHAR(10)	Degree level of student report (i.e MSc or PhD)
pub_id	INT	Publication ID from the "PUBLICATION" Table serving as the Foreign Key

Table 3: Showing the column assignment and data type for the "Student Report" table.

Journal paper table (journ)

This table contains entries that are unique to "Journal Paper" type of publications. Data is only entered into this table when creating a new entry for Journal Papers. The data in this column can be identified by the Publication ID (pub_id) serving as the Foreign Key.

Column	Data Type	Description
journ_id	INT	Primary key for Journals
pages_number	VARCHAR(20)	The range of pages where the original publication can be found within the journal
issue	VARCHAR(20)	Journal issue number where the publication can be found
pub_id	INT	Publication ID from the "PUBLICATION" Table serving as the Foreign Key

Table 4: Showing the column assignment and data type for the "Journal Paper" table.

Research Report table *(res)*

This table contains entries that are unique to "Research Report" type of publications. Data is only entered into this table when creating a new entry for Student reports. The data in this column can be identified by the Publication ID (pub_id) serving as the Foreign Key.

Column	Data Type	Description
res_id	INT	Primary Key for Student Report publications
duration_start	DATE	Start date for the research work
duration_end	DATE	End date for the research work
pub_id	INT	Publication ID from the "PUBLICATION" Table serving as the Foreign Key

Table 5: Showing the column assignment and data type for the "Research Report" table.

Other publications table (otherpub)

This table is for other publications that cannot be classified as one of the other ones specified.

Column	Data Type	Description
otherpub_id	INT	Primary key for other publications
pub_type	VARCHAR(50)	Type of publication
pub_id	INT	Publication ID from the "PUBLICATION" Table serving as the Foreign Key

Table 6: Showing the column assignment and data type for the "Other Publications" table.

Author table (author)

This table has all the information about the authors of the publications. This information is separated out into its own table as an author might have multiple publications. This is also helpful as it provides the ability to update the author's details without having to edit all the author's publications manually.

Column	Data Type	Description
author_id	INT	Primary Key for authors
title	VARCHAR(10)	Author's title (e.g Mr, Mrs. Miss, etc.)
FirstName	VARCHAR(25)	Author's First name
MiddleName	VARCHAR(25)	Author's middle name
LastName	VARCHAR(25)	Author's last name
email	VARCHAR(35)	author's email address
phone	INT(15)	Author's phone number
Institution	VARCHAR(50)	Institutional affiliation of the author (e.g university or company)
PostalAddress	VARCHAR(100)	Postal address of the author

Table 7: Showing the column assignment and data type for the "Authors" table.

Publication/author table (author)

This is a relational table that links the "Publication" and "Author" tables together. From this table, we can match a single publication to multiple authors or vice versa. This is an important table, as it helps us avoid having repeat entries of the publications or authors.

FirstName	VARCHAR(25)	Author's First name
pub_id	INT	Publication ID from the "pub" Table serving as a Foreign Key
author_id	INT	Author ID from the "author" Table serving as a Foreign Key

Table 8: Showing the column assignment and data type for the "Journal Paper" table.
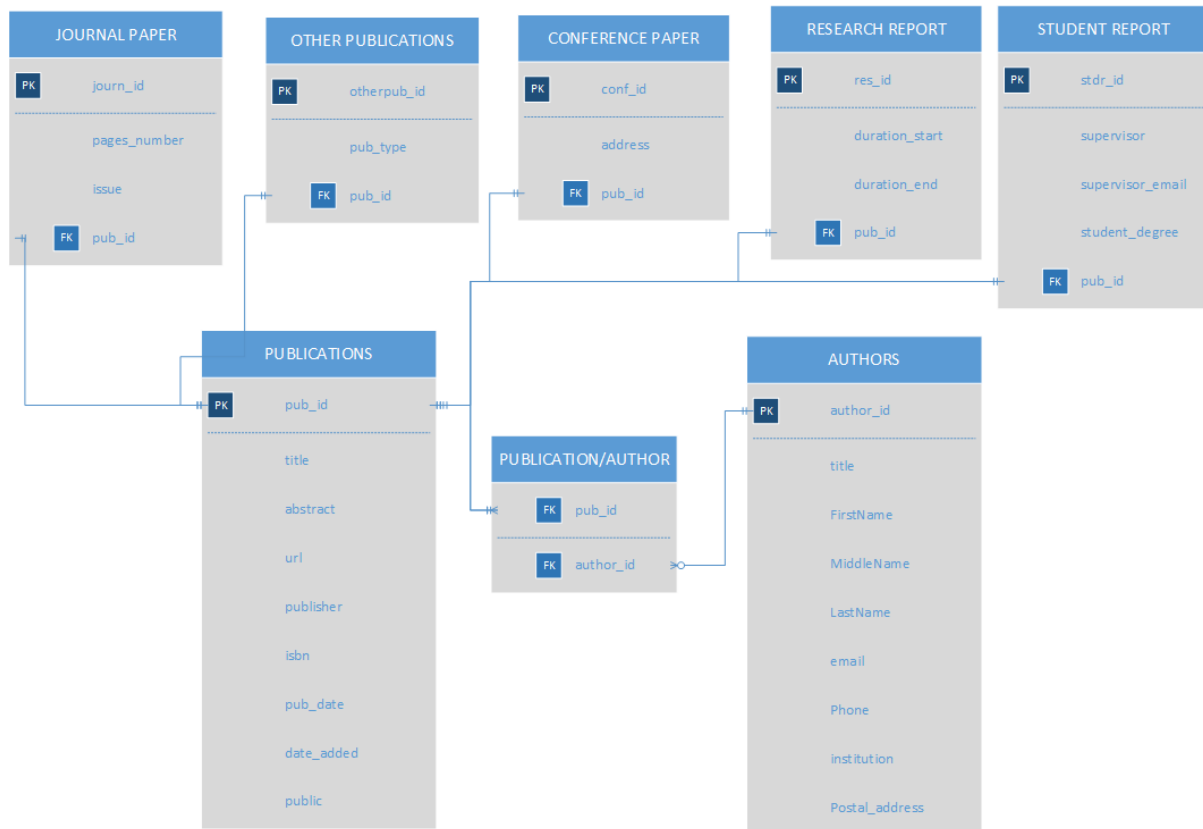
Figure 11: Database Entity Relationship diagram for the Publication Database

6.2.2 User data

The tables relating to user accounts are defined below. As with the EPL publication tables, these tables were designed to eliminate duplication of data and allow for easy access to relevant data in as little time as possible.

To achieve this, the user information is separated out into different tables. These are listed below.

Login table *(login)*

This table contains data required to authenticate the account of any user on the system. The data in this table is queried whenever any user tries to login to his/her account on the EPL. The data here is also used to determine the level of access the user will have, and download counter for the user. This data is separated out from the rest of the user information for security reasons, and also to reduce make it easier to query for the right user credentials when a login is attempted.

Column	Data Type	Description
login_id	INT	Primary Key for the login Table
username	VARCHAR(40)	username assigned to the user account
password	VARCHAR(80)	Hashed password for the user account. The password is stored in an encrypted hash, which is compared when a user attempts to login
downloads	INT(9)	Total Number of downloads per month. Incremented whenever a new download is made and reset at the end of the month
level_id	INT	Primary Key for the "levels" table serving as a Foreign Key for association with that table

Table 9: Showing the column assignment and data type for the "Login" table.

User info table (user)

Column	Data Type	Description
user_id	INT	Primary Key for "user" table
FirstName	VARCHAR(25)	The First name of the user
MiddleName	VARCHAR(25)	Middle name of the user
LastName	VARCHAR(25)	Last name of the user
dob	DATE	Date of birth of the user
login_id	INT	Primary Key for the "cred" table serving as the Foreign Key for association

Table 10: Showing the column assignment and data type for the "User info" table.

Address table (addr)

This table is used to store all the address information for each user. This is separated out in this way because the users might have more than one address attached to one user account. It also makes it easy to update the address information for a user without having to edit the rest of the information related to that user.

Column	Data Type	Description
addr_id	INT	Primary key for the address table
house_no	INT(5)	House number for the address
street	VARCHAR(20)	Street name for the address
town	VARCHAR(20)	Town name for the address
city	VARCHAR(20)	City name for the address
postcode	VARCHAR(10)	Postcode for the address
county	VARCHAR(25)	county/state/province for the address

country	VARCHAR(40)	Country name for the address
user_id	INT	Primary Key for the "user" table serving as the Foreign Key for association

Table 11: Showing the column assignment and data type for the "Address" table.

Class table (level)

This table contains all the defined user access levels in our system. Here, we can define the different types of users (students, staff, admin, etc.) and assign restrictions to them such as maximum number of downloads. The "class_id" attribute will then be appended to all users and this can be used to check the level of access that user has.

This table is necessary as it makes it easy to change the parameters regarding each user class without having to change all the users on that level manually. It insures against hardcoding user levels into the system, as this can make changes difficult to implement. It also makes it easy to assign a user to a particular class, elevate access or remove access without having to delete entries in other tables.

Column	Data Type	Description
level_id	INT	Primary key for the class table
class_name	VARCHAR(20)	Name of the user class
max_downloads	INT(5)	Maximum number of downloads assigned to that user class

Table 12: Showing the column assignment and data type for the "Class" table.

Phone number table (phone)

This table lists all the phone numbers for all the users in the system. The phone numbers are extracted into their own table to make it easier to manage. It is also necessary because multiple phone numbers could be assigned to a single user account.

Column	Data Type	Description
phone_id	INT	Primary key for phone table
phone_no	INT(15)	Phone number for a particular user
user_id	INT	Primary Key for the "user" table serving as the Foreign Key for associating a user with the phone number

Table 13: Showing the column assignment and data type for the "Phone number" table.

Email addresses *(email)*

The Email addresses are also separated for the same reason as the phone numbers.

Column	Data Type	Description
email_id	INT	Primary key for email address table
email_address	INT(15)	Email address for a particular user
user_id	INT	Primary Key for the "user" table serving as the Foreign Key for associating a user with the email address

Table 14: Showing the column assignment and data type for the "Email Address" table.
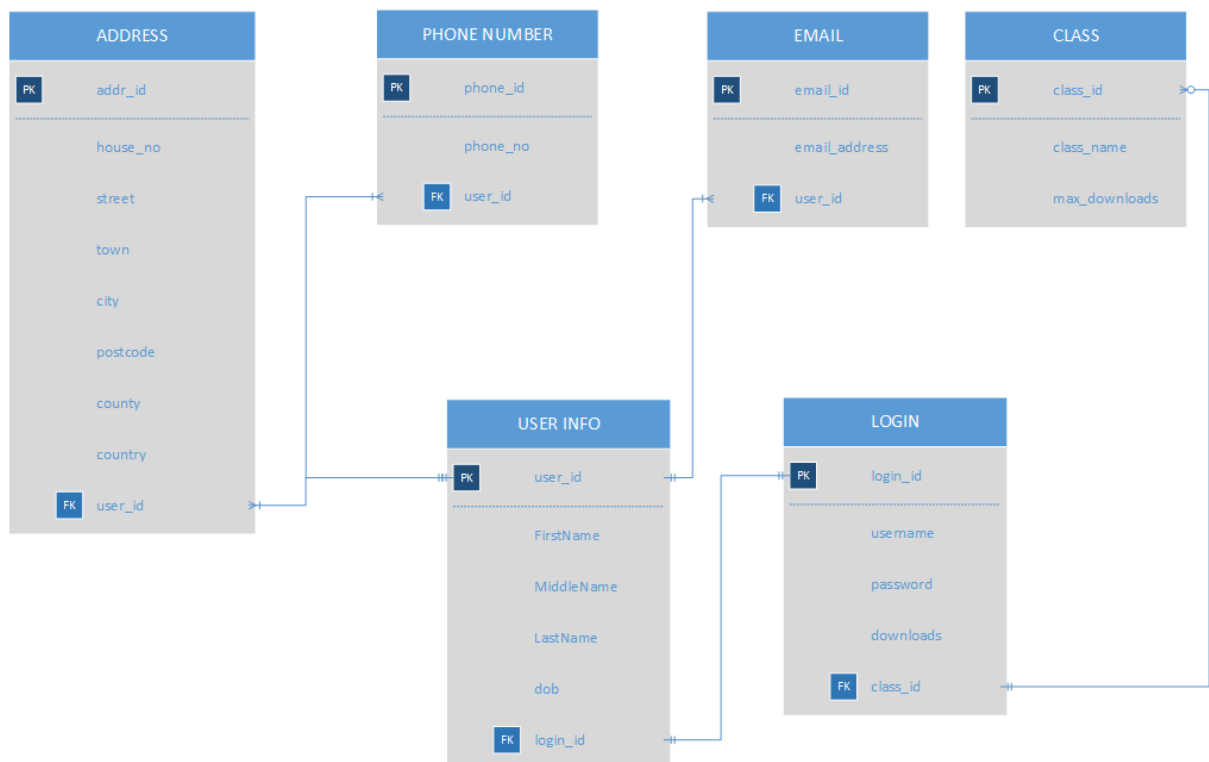

Figure 12: Database Entity Relationship diagram for the User Database

7. Conclusion

In this report, we have explored all the factors necessary for designing and building the Engineering Publication Library for the School of Engineering. The Literature review considered other similar systems in use, and compared their features. We also discussed the advantages of a web based Publication management system over the alternative.

Before beginning the design, we looked into our design objectives and the considerations that should be made to ensure that the website meets all the goals it needs to.

The design was then discussed in detail. First, we discussed the overall system functions which detailed everything the system will be able to do. The different classes of users in the system were discussed with the features of the system each class of user can access.

The Database design for both User and Publication databases were discussed in great detail, explaining the justification for the decisions made.

The Engineering Publication Library (EPL) Publication management system design has been fully outlined above, and it should be possible to implement this web project using the specifications from this report.

References

Eden, Bradford Lee (March 2006). *Content Management Systems*. Emerald Group Publishing Ltd. p5.

Kresh, D 2007, Whole Digital Library Handbook, ALA Editions, Chicago, IL, USA.

eprints. (2015). *eprints.* Available: http://www.eprints.org/uk/. Last accessed 16th november 2015.

Wikipedia, (2015). *EPrints.* Available: https://en.wikipedia.org/wiki/EPrints. Last accessed 16th november 2015.

parade@portsmouth. (2015). *Welcome to Parade@Portsmouth.* Available: http://eprints.port.ac.uk/. Last accessed 18/11/2015.

Benz D., et-al. (December 2010). The social bookmark and publication management system bibsonomy. *The VLDB Journal — The International Journal on Very Large Data Bases*. 19 (6), 849-875.

Yadav, SC, & Singh, SK 2009, Introduction to Client Server Computing, New Age International, Daryaganj, Delhi, IND. Available from: ProQuest ebrary. [16 November 2015].

W3SCHOOLS. (1999). *Introduction to HTML.* Available: http://www.w3schools.com/html/html_intro.asp. Last accessed 30th Oct 2015.

Wikipedia. (2015). *Cascading Style Sheets.* Available: https://en.wikipedia.org/wiki/Cascading_Style_Sheets. Last accessed 30th Oct 2015.

MDN. (2015). *About JavaScript.* Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript. Last accessed 30th Oct 2015.

Apache Software Foundation. (1997). *Apache HTTP Server Project.* Available: https://httpd.apache.org/. Last accessed 30th Oct 2015.

Nixon, R 2014, *Learning PHP, Mysql, Javascript, CSS & HTML5: A Step-By-Step Guide To Creating Dynamic Websites*, Sebastopol, Calif: O'Reilly Media, eBook Collection (EBSCOhost), EBSCO*host*, viewed 30 October 2015.