



UMS
UNIVERSITI MALAYSIA SABAH

FACULTY OF COMPUTING AND INFORMATICS

FINAL YEAR PROJECT I

SEMESTER 2 2020/2021

**SECURE WEB-BASED SYSTEM FOR COURSE FILE
STORAGE USING HYBRID ENCRYPTION**

STUDENT NAME : HERRY LIM

MATRIC NO. : BI 18110212

PROGRAM : HC05 NETWORK ENGINEERING

SUPERVISOR NAME : DR. LEAU YU BE

ABSTRACT

Every end of semester every lecturer required to have to prepare course file for the purpose of audit. The conventional method to store course file is not efficient and may lead to some issues like required large space to store it, document will get damaged, confidentiality issue, and hard to arranged in structured way. In this project, the objectives are to create web system for course file storage, implement hybrid encryption in the web system, and test the function of the web system using user acceptance test by getting user feedback. The methodology used in this project is agile method. The outcome of this project is web-based system to store course file.

Contents

ABSTRACT	i
CHAPTER 1	1
INTRODUCTION	1
1.1 Introduction	1
1.2 Project Background	2
1.3 Problem Statement	3
1.4 Project Goal	5
1.5 Objectives	6
1.6 Project Scope	6
1.7 Project Timeline	11
1.8 Organization of Report	12
1.9 Conclusion	14
CHAPTER 2	15
LITERATURE REVIEW	15
2.1 Introduction	15
2.2 Overview of Course File System	16
2.3 Overview of Hybrid Encryption	21
2.3.1 Comparison of SSL Certificate and SSL Certificate Experiment	28
2.4 Review of Existing System	29
2.4.1 e-Course File System in the Department of Engineering, Faculty of Engineering and Life Sciences, Universiti Selangor (UNISEL)	30
2.4.2 OpenKM	32
2.4.3 Papermerge	38
2.4.4 Existing System Comparison and Existing Systems' Features Adoption to Proposed System	43
2.6 Conclusion	44
CHAPTER 3	45
METHODOLOGY	45
3.1 Introduction	45
3.2 Selection of Development Methodology	46
3.3 Software and Hardware Requirement	47

3.4 Research Methodology	48
3.5 Conclusion	48
CHAPTER 4.....	49
SYSTEM ANALYSIS AND DESIGN.....	49
4.1 Introduction.....	49
4.2 System Analysis and Design	49
4.2.1 Data Flow Diagram (DFD).....	50
4.2.2 Entity Relationship Diagram (ERD)	56
4.2.3 Data Dictionary	57
4.3 User Interface (UI) Design.....	59
4.4 Conclusion	82
CHAPTER 5	84
RESEARCH IMPLEMENTATION / EXPERIMENT	84
5.1 Introduction.....	84
5.2 Research Background	84
5.3 Hypothesis.....	85
5.4 Experiment.....	85
5.4 Test Hypothesis	88
5.5 Analyse Results	88
5.6 Conclusion	90
CHAPTER 6	91
CONCLUSION	91
6.1 Introduction.....	91
6.2 Conclusion	91
6.3 Future Work.....	94
REFERENCES	95
APPENDIX	99
Appendix I: Interview with Dr Ervin Gubin Muong	99
Appendix II: Interview with Dr Suraya Alias.....	101

CHAPTER 1

INTRODUCTION

1.1 Introduction

Chapter 1 focuses on project history, problem description, aim, project scope, project schedule, report format, and summary. It uses hybrid encryption to build a secure web-based system.

The issue background focuses on project motivation. Inspiration is gained through watching the user's problem, interviewing consumers to fully grasp the challenges of users. Further study is done by reading books on managing the issue.

The problem statement covers typical methods of storing course files. The problems must be addressed, the significance of a web-based file storage system.

The project aims to create web-based system for course file storage and hybrid encryption to preserve privacy and privacy of online system data.

The goal describes the three major goals of this project. The project's goals are anticipated to be achieved by project conclusion. The project will be developed to the primary goals.

The project's schedule addresses what should be accomplished in that particular timeframe. The project schedule is designed to organise project due dates for various tasks from project start to conclusion for two semesters.

Project organisation describes how the chapter is organised. In project organisation, each summary chapter is addressed to explain the chapter topic.

Project summary summarises project report. The summary attempts to help the reader to better grasp the report's primary point.

1.2 Project Background

To ensure the quality of education in the Institute of Higher Learning, every Institute of Higher learning must undergo the Malaysian Qualification Agency's accreditation process (MQA). To guarantee the assessment's consistency, MQA develops the Malaysian Qualifications Framework (MQF). The development of MQF will describe, systematise, unify and harmonise all qualifications in Malaysia. The code used by MQF to achieve this goal is the Code of Practice for Program Accreditation (COPPA), issued in 2008.

All Institute of Higher Learning is required to represent their course file during the audit time. Course file is needed to see whether the Institute of Higher Learning is following the standard MQA. The course file is compulsory to store for five years for each batch. The conventional method to store course files is storing them using manual record or storing it in google drive.

The conventional method of storing course files is inefficient and can lead to some issues. The issues of storing course files using a manual record are lack of storage space, security issues, prone damage, and document transportation issues (Melo, 2019). This fact is also supported by LinkedIn. LinkedIn (Breitmeyer, 2015) states that storing manual records will give more disadvantage than the advantage to the organization. Storing paper documents will be timely to keep up, especially for abundant documents. Storing manual records requires much effort when searching for specific documents, which may cause inefficient Customer Service because Customer Service takes time to respond to Customer queries.

During the COVID – 19 pandemics, lecturers start using Google Drive to store course files. Every lecturer will create a specific file in Google Drive for each of their related course files. The issue of storing course files in Google Drive is a security issue. Another lecturer is able to access another lecturer course file. Storing course files in Google Drive can be so messy and unstructured that it makes it hard to analyse.

After realizing that issue faced by the Institute of Higher Learning on the conventional way of storing course files, the decision of creating a web-system for course file storage had been made. To understand the web-based system for course file storage, a web-based course file system from another university had been searched. However, the result from the research shows that only a few universities make their web-based system for course file storage and most universities are still using conventional methods to store course files.

Based on the fact that only a few Universities used web-based systems to store course files. The web-based system for course file system will be developed by taking inspiration from other organization web-based systems for file storage that will be customized according to Faculty of Computing and Informatics (FCI) UMS needs.

The system is developed for FCI to see the effectiveness of the system to store course files. If the solution that provides for FCI to develop a web-based system for course file storage, is effective and efficient to replace the conventional method of storing course file, the solution will be expanded to university scale. On the contrary, if there is any issue with the web-based system for course file storage, it is much easier to debug and improve, compared to starting the web-based system from the university scale.

1.3 Problem Statement

The major problem with the technique of keeping course files is that there is no customized system that has been developed, in other words, no web system that has been built to save FCI course files in an efficient and effective manner. Furthermore,

the traditional way of keeping course files, such as hard copies or soft copies on Google Drive, would result in the following issues:

i. The issues when the course file is store in hard copy is:

- Required large space to store it (Breitmeyer, 2015)

Every lecturer is required to keep every document related to the course file for five years.

During this time, the record will pile up. Thus, it needed a large space to store it. In the case of FCI UMS, the course file will be stored at the faculty's quality room for around five to six year for each batch. This situation will cause the room to become crowded with an abundance of documents (Alias, 2021).

- Environmental issue due to high paper consumption (Shah et al., 2019)

A study from the Department of Business Administration, College of Applied Sciences, Salalah, Sultanate of Oman shows that storing documents offline or manual recording has a bad impact on the environment due to high consumption of paper. This fact is supported by The World Counts (The World Counts, 2021) that the world will consume almost 120 million tons of paper in 2021, which will contribute to greenhouse gas emission.

- The course file original copy prone to lost (Moung, 2021)

Course files need to be stored for five to six year, during this time when the hardcopy of the document is lost, the lecturer is required to provide another copy of the course file. However, the lecturer may lose the document due to laptop issues, accidentally deleting the document, and forgetting where the file is stored.

ii. The issues when the course file is store in Google Drive:

- Hard to arranged in a structured way (Harkous and Aberer, 2017)

Google Drive has no specific format to store it. Thus, every lecturer can upload every

document in their format.

- Confidentiality issue (Harkous and Aberer, 2017)

Confidentiality issue happens because another lecturer that not related to the subject

can access the subject course file. Due to this issue the implementation of hybrid encryption is done to protect the access to the proposed system for course file storage.

- Version control system issue (Alias, 2021)

The version control system happened due to a myriad of documents from the last batch, which made the lecturer confused whether the course file is the latest or it was from the old version of course file.

iii. The issues when the course file is store in Google Drive and hard copy:

- No auto-alert system (Alias, 2021)

The conventional way of storing course files has no alert system to alert the lecturer if they have not completed the uploading of course files.

This issue can be solved by creating a web system to store course files and increase data confidentiality in the web system (Harkous and Aberer, 2017). The confidentiality criteria and techniques will be applied to the system. Static and dynamic testing also will be performed to ensure the web system confidentiality is used correctly.

1.4 Project Goal

The goal of the project is to develop a web-based system for course file storage. To secure the access of web-based systems the hybrid encryption will be implemented.

1.5 Objectives

- i. To investigate and implement hybrid encryption to improve confidentiality of web systems for course file storage.
- ii. To design and develop a web system for course file storage which can store course files securely.
- iii. To evaluate the proposed web system in terms of its function's performance by testing the user acceptance from user feedback.

1.6 Project Scope

The project's scope is to create a web system for course file storage for FCI and apply a cybersecurity method based on Cybersecurity Malaysia guidelines (Cybersecurity Malaysia, 2019) focusing on confidentiality on the web system by implementing hybrid encryption. The primary user of this web system is lecturer, head of the program, the quality panel, dean or deputy dean, and assistant registrar as the admin. In this system admin was held by the assistant registrar because the assistant registrar has the data about the list of lecturers, subject and position that they hold. The lecturer will be assigned as a lecturer. If the lecturer holds more than one position, they assign to both positions that they hold. Take an example. If the lecturer is also head of the program, the lecturer will have two accounts as lecturer and director of the program. Every semester, every lecturer must prepare a course file for each subject they are taught at the end of the semester. The course file will be reviewed by the head of the program, and if there is a problem with the document, submit it if the lecturer requests an update. Then, the document will be forwarded to the quality panel for further review, and lastly to the dean/deputy dean for approval. Each module of the web system is explained in Table 1.6.

Table 1.6 : Web System Module and Function

Module	Description	User
Account Management	<ul style="list-style-type: none">• Manage the users account according to their position, and subject teach.• Update, Delete, Add user.• Assign email and password that need to be changed later for the new user.	Admin (Assistant Registrar)
Folder Management	<ul style="list-style-type: none">• Create a submission folder for each batch of course files.• Assign each subject submission folder to the specific lecturer who teaches that subject.	Admin
Login, Logout, Authentication	<ul style="list-style-type: none">• The login process required users to enter their email using their official UMS email, password, and position. For the first time user, the user is required to use email and password that they have already assigned to them.• All the lecturers will be assigned as a lecturer. If the lecturer holds more than one position, the lecturer has more than one separated account according to their position. Take an example if the lecturer also holds the position as head of the program. Then, the lecturer will be assigned to two	

	<p>different accounts, lecturer and head of program.</p> <ul style="list-style-type: none"> • All the users must log in using UMS email. Email from another domain will be rejected. • The login authentication will be authenticating if the user enters the email in the correct format. The correct format of email will allow the user to login. After the login is submitted, the login will be authenticated if the user enters an UMS email. 	Lecturer, Head of Program, The Quality Panels, Dean / Deputy Dean, and Admin.
<p>User Dashboard: Lecturer Dashboard, Head of Program Dashboard, The Quality Panel Dashboard, Dean/Deputy Dean Dashboard, and Admin Dashboard</p>	<ul style="list-style-type: none"> • After login each user will go to the specific dashboard according to their position. • In the lecturer dashboard, it contains a specific folder about the subject that is taught by the lecturer. In this folder the lecturer is required to upload every document related to the course file. After the lecturer finishes uploading all the documents, they can submit it to the Head of Program. • In the Head of Program dashboard, it contains all the documents that are submitted by the lecturer. The Head of Program is required to check it before sending it to the Quality Panel. If there is any issue with the document the Head of program can reject the 	Lecturer, Head of Program, The Quality Panels, Dean/ Deputy Dean, and Admin.

	<p>document and request a new document upload from the lecturer.</p> <ul style="list-style-type: none"> • The Quality Panel dashboard contains the documents that have been checked by the Head of Program. In this dashboard the Quality Panel is required to review the document that is sent by the Head of Program before they send it to the Dean/Deputy dean for approval. • In the Dean / Deputy Dean dashboard, it contains the documents that have been checked by the Quality Panel. In this dashboard Dean / Deputy Dean is required to check documents that have been sent and do the document approval. • In the Admin dashboard, it contains account management, folder management , and progress monitoring the Lecturer, Head of Program, Quality Panels, and Dean / Deputy work. The progress monitor will monitor if all the documents have been uploaded or checked. Progress monitoring will be able to show which users have not completed their work by showing it in which part the document is pending. When the due date is under the corner and Admin finds out that some user has not completed their work, Admin can notify 	
--	--	--

	<p>the user through the Notification feature. When the uploading and document checking is finished, the progress monitor will give an alert "Complete". The finished document will be stored in this Admin dashboard.</p>	
File Management	<ul style="list-style-type: none"> • In every folder, it contains a file that requires the lecturer to upload a specific file. The file is arranged by the contain requirement of the file, and colour tags. • In this file management, it has a file viewer feature that allows the user to view files directly from the web. • Each file required the lecturer to upload is related, any file that has not been filled with a document will have an alert "Not Complete", and if the document has been completed the alert will be changed to "Complete". • Lecturers only can submit the folder when all the document has been uploaded, if any of the incomplete lecturers are not allowed to send the folder. 	Lecturer, Head of Program, The Quality Panels, Dean / Deputy Dean, and Admin
Hybrid encryption	<p>This module runs in the backend, to protect the web system access from unauthorised users.</p>	Lecturer, Head of Program, The Quality Panels, Dean / Deputy Dean, and Admin

1.7 Project Timeline

The project is divided into two phases FYP I and FYP II, in FYP I the project is focusing on research of web systems for course file storage and research of hybrid encryption. FYP II is focusing on the development of web-based systems and implementation of hybrid encryption in the web system. The web system testing to see the web functionality, static and dynamic testing also will be done in this project. The project flaws and bugs also will be fixed in this project. The project timeline of the project is explained in Figure 1.7.

Activity/ Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Final Year Project I (FYP I)															
Introduction															
Literature Review															
Methodology															
Progress Presentation															
Experiment															
System Analysis and Design (SAD)															
Conclusion															
Submission FYP 1															
Viva session															
Final Year Project II (FYP II)															
Web development															
Hybrid Encryption Implementation															

In this chapter, the discussion is focusing on the process from the starting of the project until the project is finished. The methodology approach to finish the project will be discussed in a detailed way. This chapter also includes the strategies and methods that would be used in the project's design and execution. Aside from that, this chapter also discusses justification for the methods/approaches used, as well as software and hardware requirements.

iv. Chapter 4 : System Analysis and Design

In this chapter focusing on the system analysis that will be implemented in the project. The system analysis that will be implemented in the project is Use Case Diagram, Entity Relationship Diagram (ERD), Context Diagram, and Data Flow Diagram. This chapter also discusses User Interface Design.

v. Chapter 5 : Conclusion

In this chapter discussing the summary of the project. Each of the chapter work will be summarized in this chapter. The finding of the project is also discussed, and the main point of what the reader should get in this project will be summed up in this chapter and future work.

1.9 Conclusion

In conclusion, the course file is compulsory for all Institute of Higher Learning to prepare for the purpose of audit, to ensure that the Institute of Higher Learning is following the standard of MCQ. However, the conventional method of storing course file using hard copy and Google Drive caused a lot issue to the user such as required large space to store course file, environmental issue due to high paper consumption, the course file original copy prone to lost, hard to arranged in a structured way, confidential issue, version control system issue, and no alert system. Thus, the web-based system for course file storage needs to be developed. The objectives are to design and develop a web system for course file storage, to investigate and implement hybrid encryption to improve confidentiality of web systems for course file storage and to evaluate the proposed web system function's performance and perform static and dynamic testing to test the confidentiality of the web system. The scope of the project is focusing on FCI UMS, to develop functioning web-based course file storage to solve the issue that has been faced by the user when storing the course file in conventional ways.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter focuses on the previous research and the existing system that is similar to web-based system for course file storage that will be done in this project. The existing system of course file storage will be studied to find out the way the system works, the system features, and system design. The findings from this study will be used as an example for the system in this project. The useful features and design will be adopted and adjusted to the project according to the project need.

In addition, this chapter is focusing on hybrid encryption to protect the privacy and data confidentiality of the web-based system. The preliminary study of hybrid encryption will be discussed in detail in this chapter by referring to the previous research. The research of hybrid encryption will discuss how further this encryption technique can secure the web-based system to ensure privacy and data confidentiality. The method of applying hybrid encryption, and the difference between hybrid encryption with other encryption techniques. The hybrid encryption will be compared with symmetric encryption, and asymmetric encryption. The best encryption technique will be chosen.

2.2 Overview of Course File System

Course file system can be defined as a folder that contains all of the relevant information about the batch, evaluation, and overall results of the course taught in every program. Course file include course synopsis (Table 4 MQA), list of students registered, teaching plan, lab sheets, students attendance record, teaching materials, list of assessment, mark distribution, grade analysis, student feedback (PK07), reflection, and online class evidence during Malaysian Government Movement Control Order (MCO).

The course file is compulsory for every faculty to prepare for the reason of an audit by MQA. The course file preparation will follow the specific format that has been setup by the faculty. In the case of FCI UMS, the course file will follow the FCI Course File Checklist. The course file will be prepared by all according to their teaching subject every end of semester. The prepared course file will be checked by the head of the program and will be verified. Then, will be passed to the to be further checked and verified. Lastly, the course file that has been checked by the quality panel will be sent to the dean or deputy dean for checking and verification. After that, the course file will be stored in the special room, the faculty's quality room. The course file will be stored for around five to six years (Alias, 2021).

The process of preparing a course file during the Malaysian Government Movement Control Order (MCO) is almost the same as the process of preparing a course file during the normal day. The only difference is during MCO the course file is prepared and stored online in Google Drive.

The process of storing course file using conventional way is not efficient and practical because it gives some issues such as required large space to store it, environmental issue due to high paper consumption, the course file original copy prone to lost, hard to arranged in a structured way, confidentiality issue, version control system issue, and no alert system.


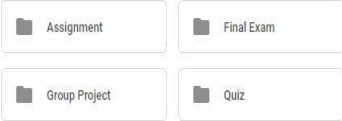
The new web-based system for course files needs to be developed due to the fact that conventional way of storing course files is not effective and practical. The new web-based system will be developed with an interactive user interface to make it easier to use. The new system will add some new features to solve the issue that

have been faced by the user when storing course files using conventional methods. The features of the new web-based system are developed according to the user requirement.

The study of existing systems will be done to understand the way the file system should work. The paramount feature from each of the existing systems will be adapting to the new system and modified according to the needs of the system. The new system also will be developed based on the work flow of conventional methods but will be added some features and modifications to make it more interactive and user-friendly. The content format of the course file based on FCI Course File Checklist will be as follows. The content of FCI Course File Checklist is shown in Table 2.21.

Table 2.21 : FCI Course File Checklist (FCI, 2020)

N O	TYPES OF FOLDER	CONTENTS	FILE NAME	EXAMPLE
1 .	Course Specific Information	Excel file that content: <ul style="list-style-type: none"> • Timetable • Lesson/Teaching Plan • Table 4 • CAP • Simplify JSU for all assessment (Except Final Exam) 	Course Code Course Information	IE32603 Course Information
		Printed page of Smart2/Smartv3 UMS	Course Code Smart2/Smartv3	IE3263 Smartv3
2 .	Students List &	Students List registered on SMP (Please download the list in week 4)	Course Code Students List	IE32603 Students List

	Student Attendance Analysis			
		Student Attendance Analysis	Course Code Attendance	IE32603 Attendance
		Please create 1 folder to store students' absent letters. Example: 	Date of Absent Student's Matric Number/ Name	12 September 2018_ NurSafiqah
3.	List of Assessments	Create one folder for each assessment as stated in Table 4. Example: 	Name the folder based on what stated in Table	<ul style="list-style-type: none"> • Assignment • Final Exam • Group Project • Quiz
		For each assessment folder: Descriptions/ Questions of assessments	Course code Name the file based on what it represents. **If the documents are confidential, protect the documents with passwords, however, for audit purposes, please provide the	<ul style="list-style-type: none"> • IE32603 Final Exam Question OR • IE32603 Individual Assignment Description
		Marking Scheme/Answer Scheme/Marking Rubric.		<ul style="list-style-type: none"> • IE32603 Final Exam Answer Scheme OR • IE32603 Group

			password to FKI upon request.	Assignment Marking Rubric
		<ul style="list-style-type: none"> • 9 samples answers for each Final Exam, Mid-Term Exam, Quiz, and Test. (Which are applicable) <ul style="list-style-type: none"> - 3 poor sample answers - 3 average sample answers - 3 good sample answers • 3 samples answers from for each assignment (e.g.: Individual Assignment, Group Assignment, Lab Assignment, Group Project)(Which are applicable) <ul style="list-style-type: none"> - 1 poor sample answers 		<ul style="list-style-type: none"> • IE32603 Final Exam_Poor1 • IE32603 Final Exam_Average1 • IE32603 Final Exam_Good 1
		<ul style="list-style-type: none"> - 1 average sample answers - 1 good sample answers 		

		If the document can't be uploaded, the lecturer must keep a hard copy. Please submit the hardcopy of the documents to FKI upon request for audit purpose.		
		Any related documents.		IE32603 Group Project Peer review
4 .	Students Performance – Mark & Grade	Mark distribution(include OBE) - (Excel)	Course Code Mark Distribution	IE32603 Mark Distribution
		Final Student Mark and Grade downloaded from SMP. The document must be in PDF format and verified by the Dean	Course Code Marks and Grade	IE32603 Marks and Grade
		Summary of Grade Analysis downloaded from SMP and must be verified by the Dean	Course Code Grade Analysis	IE32603 Grade Analysis
		OBE system report (print after fill in the UMS OBE system) – Starting Batch 2018/2020.	Course Code OBE	IE32603 OBE
5 .	Students Feedback	PK07. The PK07 must be verified by the dean.	Course Code PK07	IE32603 PK07

		Another Student Feedback (Other than PK07)lecturer initiative	Course Code PK07	IE32603 Feedback
6 .	Reflection	Lecturer's Reflection *Please refer Summary of Course Implementation Reflection (in 1_Rujukan folder Course File Checklist Form)	Course Code Reflection Lecturer's Name	IE 32603 Reflection Suaini Sura
7 .	Other Information	Any information related to the course. Example: Teaching material	Course code Name the file based on what it represents.	IE23603 Teaching Material

2.3 Overview of Hybrid Encryption

Hybrid encryption (Techopedia, 2021) can be defined as a combination of symmetric and asymmetric cryptography. Hybrid encryption is accomplished by transferring data using both identical session keys and symmetrical encryption. Random symmetric key encryption is achieved using public key encryption. The receiver then decrypts the symmetric key using the public key encryption process. The symmetric key is then used to decode the message until it has been retrieved.

The definition from Sharifian, S., & Safavi-Naini, R (2021) supports the definition from Technopedia by stating that hybrid encryption scheme is made up of two components: a public-key component and a (symmetric) secret-key component. The public-key component is referred to as the key encapsulation mechanism (KEM),

and it produces a pair of a randomly generated symmetric key K and a ciphertext c . The symmetric key component encrypts the actual data with the produced key K and generates the corresponding ciphertext c through an effective data encapsulation mechanism (DEM) (e.g. that can be constructed as a counter mode of AES). The pair (c, K) enables the decryptor to recover K from c before decrypting c and obtaining the info. Cramer and Shoup formalised the KEM/DEM model, which has been commonly used in Internet protocols to introduce public-key encryption in protocols such as TLS and SSH, as well as in specifications such as.

The research done by Safi (2017) shows the application of hybrid encryption in IOT. The researcher used hybrid encryption to improve data integrity, confidentiality, and non-repudiation during the data exchange in IOT. The hybrid encryption in this research is called HAN by the researcher. The proposed algorithm has unique characteristics in terms of encryption and decryption speed, also when creating keys, and it can also enhance internet protection by the use of multiple structures during algorithm execution and digital signatures.

The hybrid encryption that is used in this study is a combination of AES and NTRU encryption. To implement the hybrid encryption some step is done in this study. The first step is creating a 4 x 4 matrix of AES encryption. From this matrix the public key is produced. Then, the first message will be encrypted using NTRU encryption and XOR will have a public key that has been encrypted by AES encryption. For the decryption part the message will be decrypted using NTRU and XOR with a private key. The process will be explained in the HAN encryption in Figure 2.31.

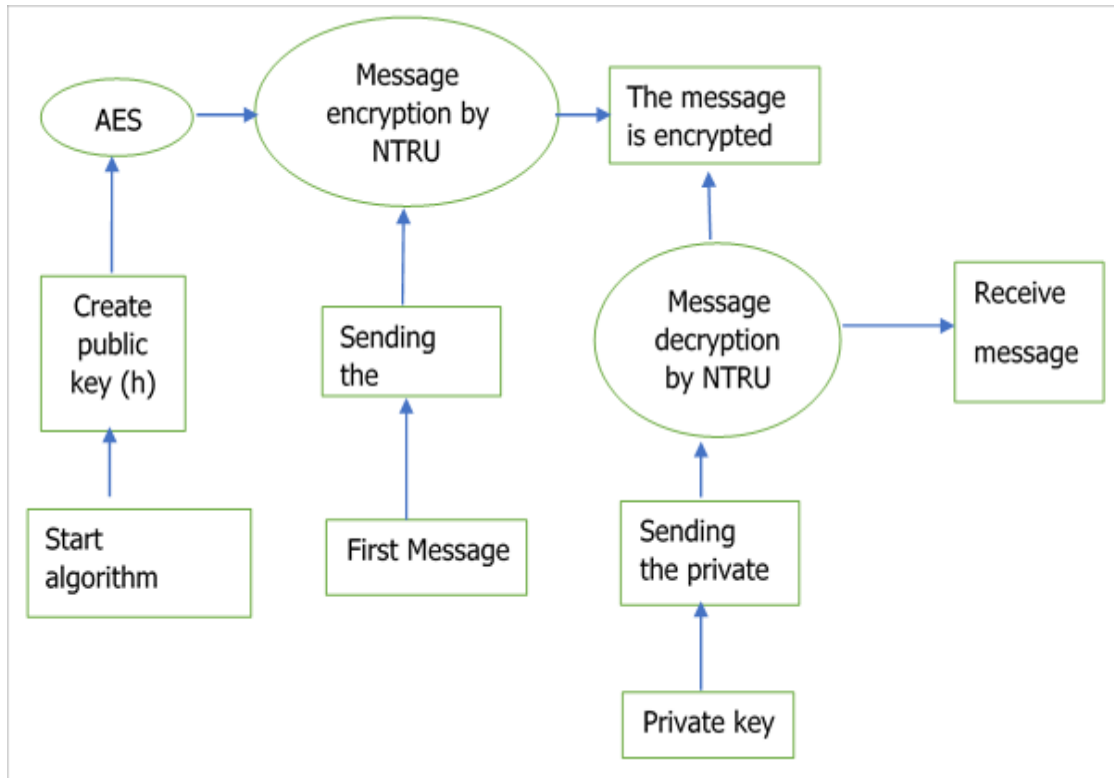


Figure 2.31: HAN encryption algorithm steps sending public key (Safi, 2017)

To prove the effectiveness and speed of HAN encryption. The HAN encryption is compared with AES and RSA encryption. The simulation for comparing the speed of encryption is done in MATLAB, and the result shows that HAN encryption is much faster than AES and RSA algorithms. Even when the researcher compares the HAN encryption combined with digital signature, compared with RSA and AES encryption without combined digital signature. The result is still the same, HAN encryption is still faster than AES and RSA encryption.

The research done by Agrawal and Patankar (2016) from Computer Engineering, Institute of Engineering & Technology Devi Ahilya University Indore, Madhya Pradesh, India developed hybrid encryption using combination of cryptography algorithms such as Diffie-Hellman Key Exchange Algorithm, RSA algorithm, Private Key Encryption, SHA-1, and RC5.

The researchers claim that the suggested solutions would not only support safe communication, but also enhance encryption by reducing overhead protection.

The architecture of the device is not dependent on the existence of any external system interface. The customer would be able to communicate with the application using a graphical user interface (GUI). From the user view, the user will enter data through text boxes and file forms and forward it to the receiver end to display performance. Between the sender and receiver computers, connectivity is accomplished through Bluetooth or a wired network. Connection between sender and receiver will be established using socket programming.

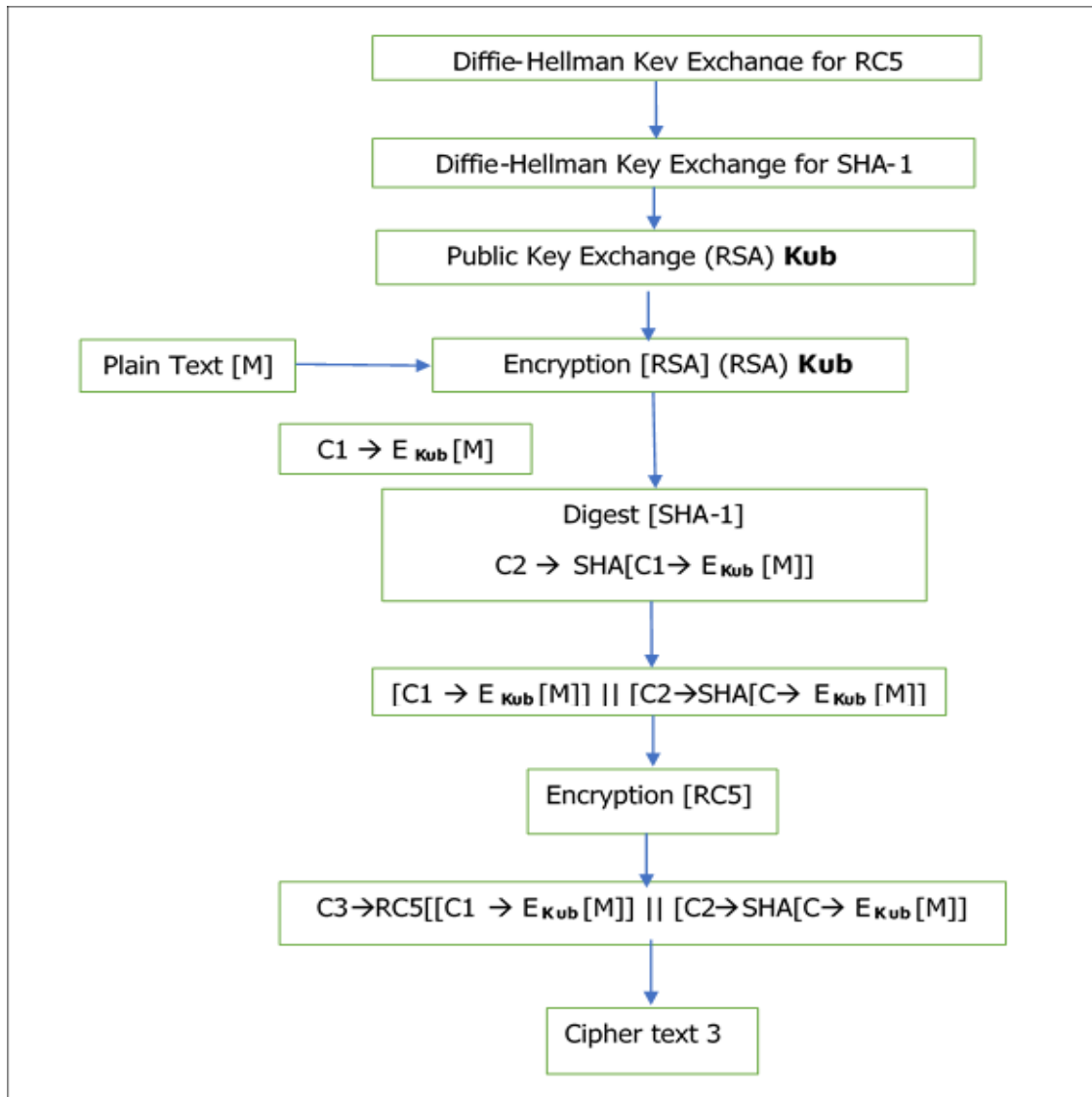


Figure 2.32: Agrawal and Patankar Proposed Encryption Algorithm's Architecture (Agrawal and Patankar, 2016)

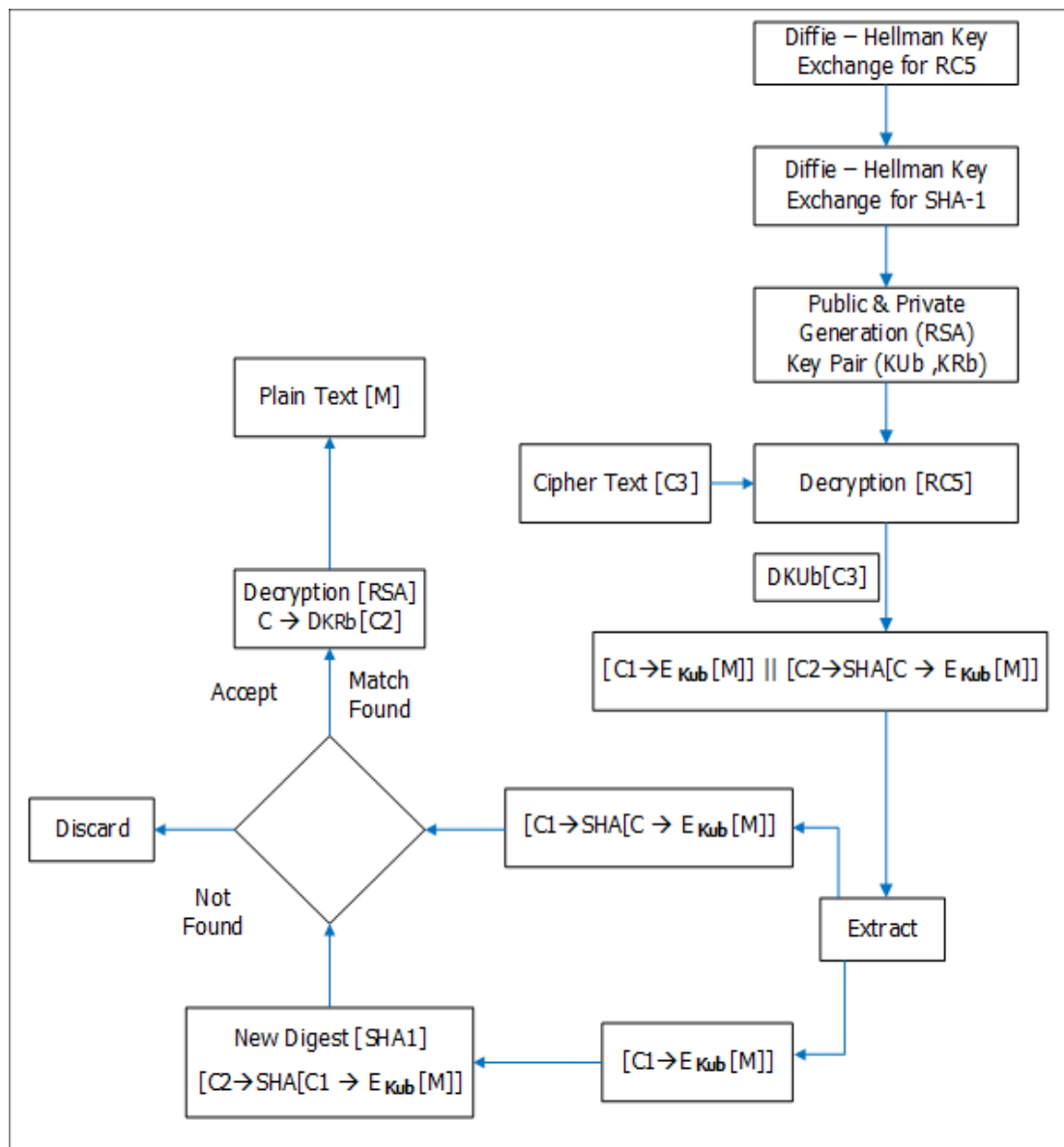


Figure 2.33: Agrawal and Patankar Proposed Decryption Algorithm's Architecture (Agrawal and Patankar, 2016)

The researcher's approach would include an alternate security paradigm to SSL and digital envelope in order to preserve intranet confidentiality. SSL needs HTTPs protocol where no protocol interference in applications is required for the proposed solution. In addition, integrating protection policies with local network systems is the fundamental application of the suggested approach.

Another way to implement hybrid encryption is by implementing SSL certificate (X.509 certificate). SSL certificates are small data files which connect an encryption key digitally to the information of an organisation. When mounted on a web server, it enables the padlock and the https protocol, allowing for safe communications between the web server and the browser (GlobalSign, 2021).

SSL certificates protect data during SSL Handshakes. As a browser tries to reach an SSL-secured website, the browser and the web server create an SSL link via a mechanism known as an SSL Handshake. To establish an SSL link, three keys are used: the public, internal, and session keys. Anything encrypted with the public key can only be decrypted with the private key, and the reverse is true. Since encrypting and decrypting with private and public keys requires a significant amount of computing power, they are only used to generate a symmetric session key during the SSL Handshake. After establishing a safe link, the session key is used to encrypt all transmitted data (DigiCert, 2021).

There are several benefits of applying for an SSL certificate in the web system. Study done by Forbes (Nikolov, 2018), applying an SSL certificate in a web system will improve the website security, provide safety for subdomain, ensure credibility and trust with the customer, and SEO advantages. The implementation of SSL certificates will improve website security due to SSL certificates encrypting confidential data sent to and from the website. Login data, signups, emails, and payment or personal information are examples of that information. SSL certificates encrypt the link and secure the visitors' information from being abused by hackers. Some SSL certificates allow users to secure its main website and all the website subdomain.

The application of an SSL certificate will also reduce the risk of phishing and gain visitor trust. In the address bar of the tab, a protection padlock will appear if the website uses an SSL licence. This indicates that the link is safe and demonstrates to the website's users how seriously the website takes their privacy. This indicator can assist the customer in distinguishing between a legitimate website and a phishing website.

The implementation of SSL certificates gives a lot of advantages to the web system. However, the implementation of authorised SSL certificate (CA certificate) will be costly and impractical when used to test websites during development. By the reason of that open source SSL certificate will be consider. There are some open source SSL Certificate that available such as OpenSSL, EasyRSA, and CFSSL.

OpenSSL (OpenSSL, 2018) is a feature-rich, stable, commercial-grade toolkit for the Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols. It's also a cryptographic library that can be used with something. OpenSSL is released under an Apache-style licence, which ensures it is free to use and use for commercial and non-commercial uses under some license conditions.

The research done by StackShare (2021) shows that OpenSSL has been used by 4240 companies like Alibaba Travel, GoDaddy, Finema, etc. OpenSSL is also integrated with Linux, Windows, Android OS, Mac OS X, and KintoHub.

Easy-RSA is a program that facilitates the management of X.509 PKI, or Public Key Infrastructure. A public key infrastructure is built on the concept of relying on a particular authority to authenticate a remote peer. The features of Easy-RSA are able to manage multiple PKIs, support Multiple Subject Name (X.509 DN field) formatting, support multiple platform, CRL, CDP, keyUsage/eKu attributes, and other functionality are included in Easy-X.509 RSA's support. As an advanced feature, the included support can be modified or expanded, interactive and can be automated, and Easy-RSA can be used without having to update a configuration file because of built-in defaults.

CloudFlare's PKI/TLS swiss army knife is CFSSL. It's a command-line interface for signing, checking, and bundling TLS credentials, as well as an HTTP API server. The CFSSL features are a collection of packages useful for building custom PKI tools, the cfssl command line utility that utilises these packages, the mroot (CFSSL) tool that builds certificate authorities, and its companion (multi-key signing), and the mkbundle (a multi-key certificate authority that generates bundles, keys, and CRLs, and bundles certificates and CAs) that uses the multiroot tool to generate CRL, multi-root certificates, have been implemented into the program.

2.3.1 Comparison of SSL Certificate and SSL Certificate Experiment

The SSL Certificate will be compared to understand the features of each SSL Certificate. Understanding the features that each SSL Certificate provides will help to choose the suitable SSL Certificate to implement in this project. Experiment technique from the previous research will be discussed to implement the experiment technique to find out which of the SSL Certificates provide more secure service and suitable to implement in this project.

Table 2.31 : Comparison of SSL Certificate

SSL Certificate	Features
OpenSSL	Feature-rich, stable, commercial-grade toolkit for the Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols, and contain cryptographic library.
Easy-RSA	Manage multiple PKIs, support Multiple Subject Name (X.509 DN field) formatting, support multiple platforms, CRL, CDP, keyUsage/eKu attributes, and other functionality are included in Easy-X.509 RSA's support. As an advanced feature, the included support can be modified or expanded, interactive and can be automated, and Easy-RSA can be used without having to update a configuration file because of built-in defaults.
CFSSL	Collection of packages useful for building custom PKI tools, the cfssl command line utility that utilises these packages, the mroot (CFSSL) tool that builds certificate authorities, and its companion (multi-key signing), and the mkbundle (a multi-key certificate authority that generates bundles, keys, and CRLs, and bundles certificates and CAs) that uses the multiroot tool to generate CRL, multi-root certificates.

There is some method that was suggested by previous researchers to test the security of SSL Certificates. The researcher, Chen et al. (2018) from Shandong University, Jinan, China propose DRLgencert that develops using deep reinforcement learning. Another researcher, Pathak et al. (2018) from Jaypee Institute of Information Technology India, using different approaches to test SSL Certificate using online lab. The result from the online lab will be graded according to the security level of the SSL Certificate, the most secure will be graded as A+ and less secure as F.

In this research, the experiment method from Pathak et al. (2018) will be adopted into this project. The reason for choosing this method is to save time for this experiment. The online lab also provides detailed assessment to test SSL Certificates which produce accurate results that help to distinguish the difference between each of the SSL Certificates. The online lab that is used in this project is ImmuniWeb SSL Security Test, because this lab provides detailed assessment that is required for this project's experiment.

2.4 Review of Existing System

The study of existing systems is done to understand how the system works and to get inspiration to develop a new web-system for course file storage. After some searching, only a few examples can be found because most of the Institute of Higher Learning still use conventional methods to store course files. The example of Higher Institution that develop web-system for course file storage is Universiti Selangor (UNISEL) that documented in Journal Online Jaringan Pengajian Seni Bina (JOJAPS), title "Development of e-Course File System in the Department of Engineering, Faculty of Engineering and Life Sciences, Universiti Selangor" (Muslim et al., 2019). Thus, another example of file storage system from different sources will be taken and adjusted according to the need of course.

2.4.1 e-Course File System in the Department of Engineering, Faculty of Engineering and Life Sciences, Universiti Selangor (UNISEL)

The UNISEL e-Course File System is an example of course file digitalization in the Institute of Higher Learning in Malaysia. This e-Course File System was developed in 2017 by UNISEL to reduce the usage of paper and stationary.

The UNISEL e-Course File System idea will be referring to creating a web-based system of course file storage. The method that is used by UNISEL to develop a web-based system of course file is the development graphical user interface (GUI) according to the content needed: course information, teaching timetable, lecture materials, final examination, and course analysis. The system is developed using PHP, MySQL database, and Apache as a web server.

The user in UNISEL e-Course File System is divided into two categories, content provider and administrator. Content provider is the place for the lecturer to upload, view, and download documents. Administrator is the place for the administrator to add a new user, semester, and course according to the need and request from the faculty department.

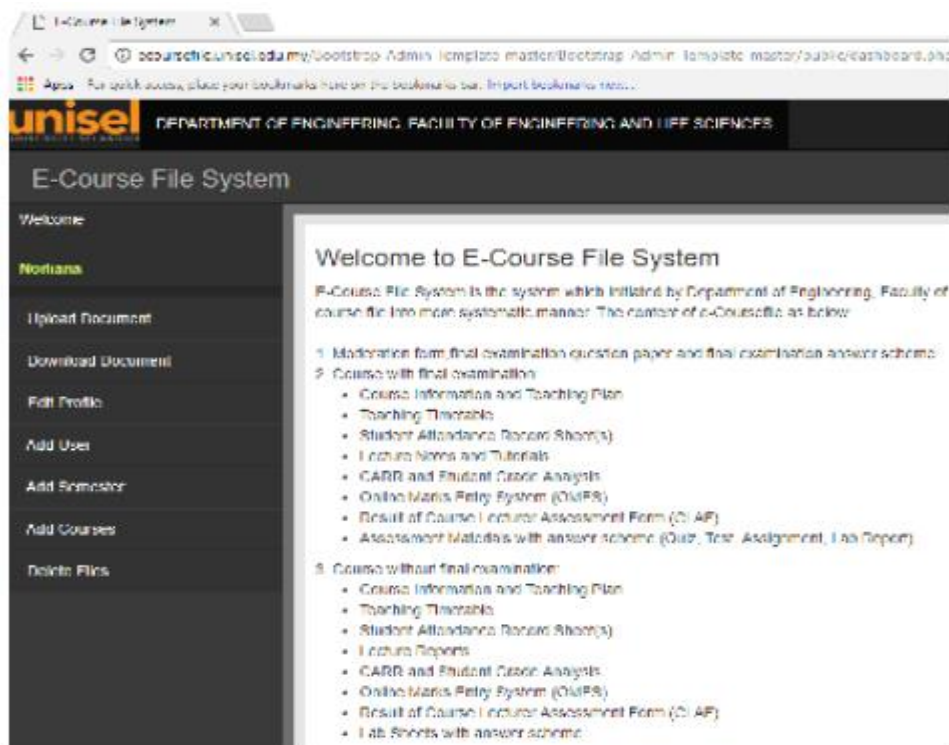


Figure 2.41 : Screenshot of Main Page of UNISEL e-Course File System

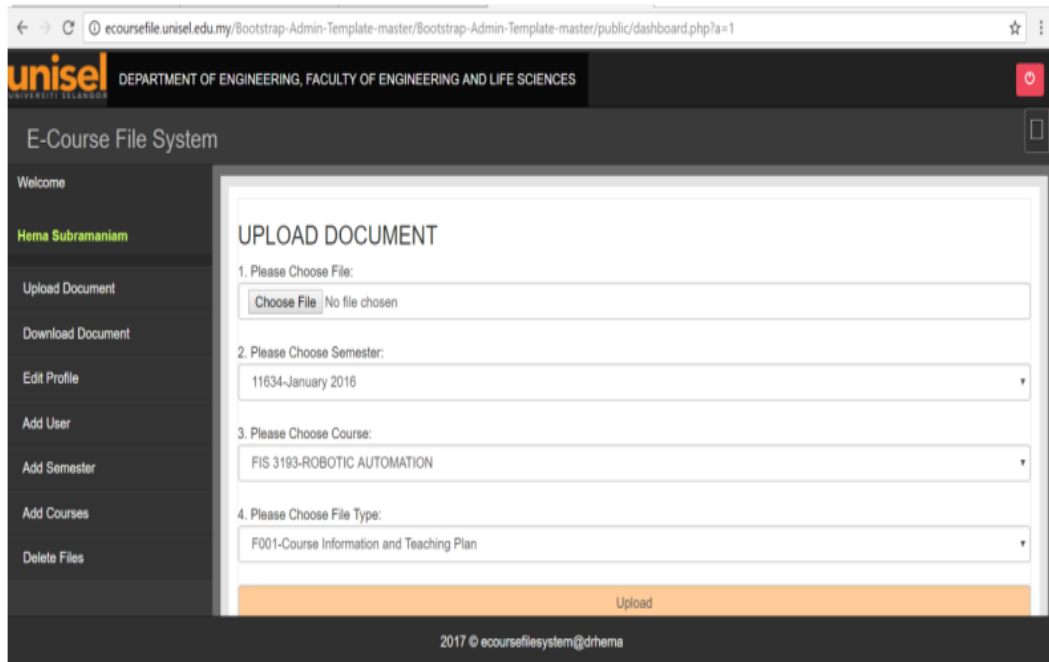


Figure 2.42 : Content Provider User Interface of UNISEL e-Course File System.

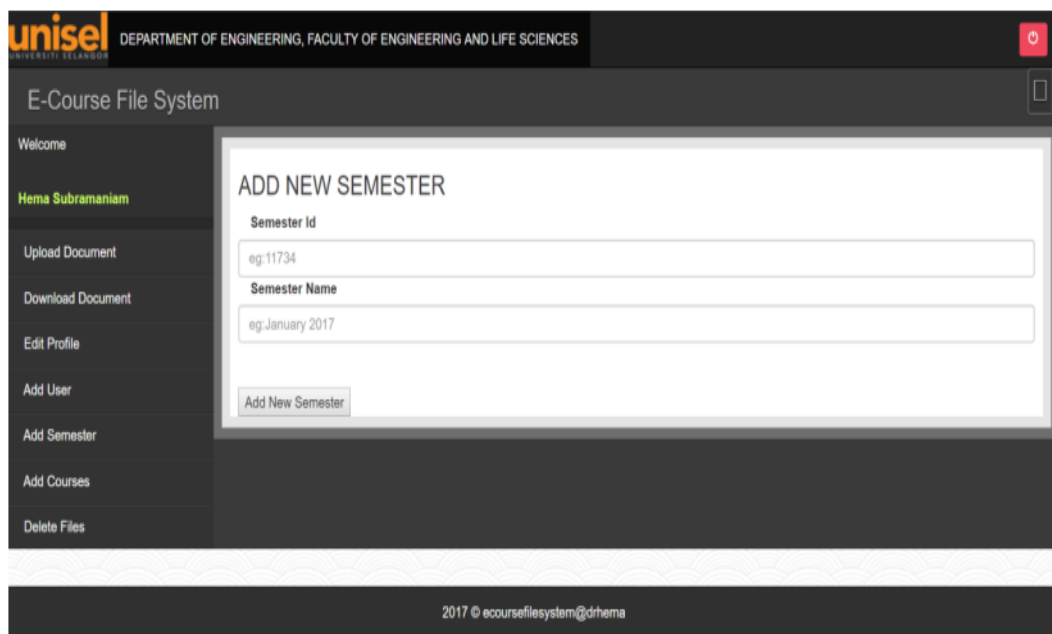


Figure 2.43 : Administrator user interface of UNISEL e-Course File System.

The system of UNISEL e-Course File System is very simple, a lot of new features should be added to make it more user friendly and interactive. The UNISEL

e-Course File System needs to add new features and some modification in the development of the new system.

2.4.2 OpenKM

OpenKM is an open-source document management system (DMS) that is used to organize any types of organizations' documents. OpenKM provides useful features of document management, team collaboration, and search functionality. OpenKM system also provides administrative tools for defining user functions, access management, user quotas, document protection levels, comprehensive operation reports, and automation configuration (Avila, 2021).

OpenKM helps users to control their organisation content, gathering organisation's data from digital sources, cooperating with another co-worker, making it easy for the organisation to analyse the data, features for managing corporate content, and organizing and managing multimedia material and documentation (OpenKM, 2021).

OpenKM provides some useful features for the user to make the process of document management more effective and efficient (OpenKM, 2021). The features provided are automatic cataloguing, automatic metadata capture, Optical Character Recognition (OCR), modules, preview, version control, tracking, barcode reader, web service, networking, integrations, and Optical Mark Recognition (OMR).

Automatic cataloguing is the OpenKM features that help users to set the format of the document event or the way of the document arrangement and management. With this feature users can customize the arrangement of documents such as automatically move the specific document to the specific folder in OpenKM. Automatic cataloguing also allows users to set up document security, and change document format such as changing document to PDF format.

Automatic metadata capture is an OpenKM feature that uses OpenKM Zone OCR technology to process data capture and document. The aim of automated metadata capture is to convert streams of documents of any structure into business-ready records.

Optical Character Recognition (OCR) is an OpenKM feature that allows you to transform different types of documents into editable and searchable records, such as scanned paper documents, PDF files, or digital camera photographs.

OpenKM provides some modules that are multitenant, mail archiver, digital signature, cryptography, integrated BPM engine, stamp, reporting engine, task manager and calendar. OpenKM also has the feature to preview AutoCAD files.

Version control feature is the administration of record updates. Integer or letter code called the revision number is commonly used to identify them. Each revision has a timestamp as well as the consumer who made the update. Changes can be compared graphically, and old versions can be restored.

Tracking is an OpenKM feature that provides a configurable audit trail. Thus, the customer would provide documentary records of the chain of events that impacted anything in the device during the execution of a particular activity, process, or occurrence.

OpenKM have bar code reading feature that support the reading of the following barcode format: Codebar, Code 39, Code 93, Code 128, EAN-8 and EAN-13, ITF-14, UPC-A and UPC-A, RSS-14, RSS Expanded, Data Matrix, PDF417, QR Code, and Aztec. OpenKM also integrates easily with third-party applications. There are integrations available with Bonitasoft BPM, Microsoft Office, and Vtiger.

Another feature provided by OpenKM is Optical Mark Recognition (OMR). OMR means interpreting styles and records with ovals and checkboxes. The Optical Mark Reader is a scanning system that is used to interpret documents that have been set up with OMR areas or regions. OpenKM also provides web services such as REST, SOAP and CMIS. It also supports CIFS, FTP and WebDAV protocols.

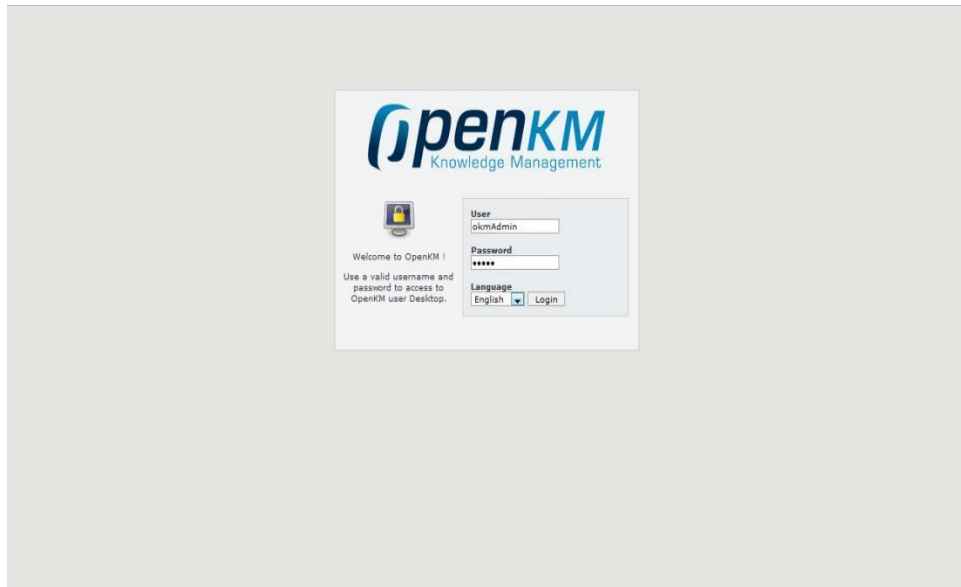


Figure 2.51 : Screenshot of OpenKM Login Page.

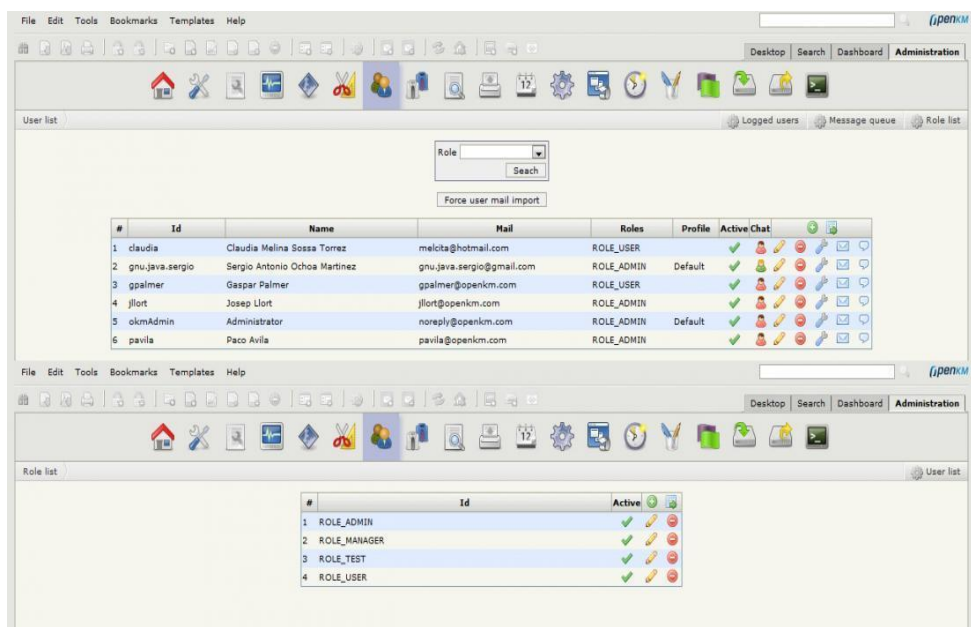


Figure 2.52 : Screenshot of OpenKM Users and Role Setup.

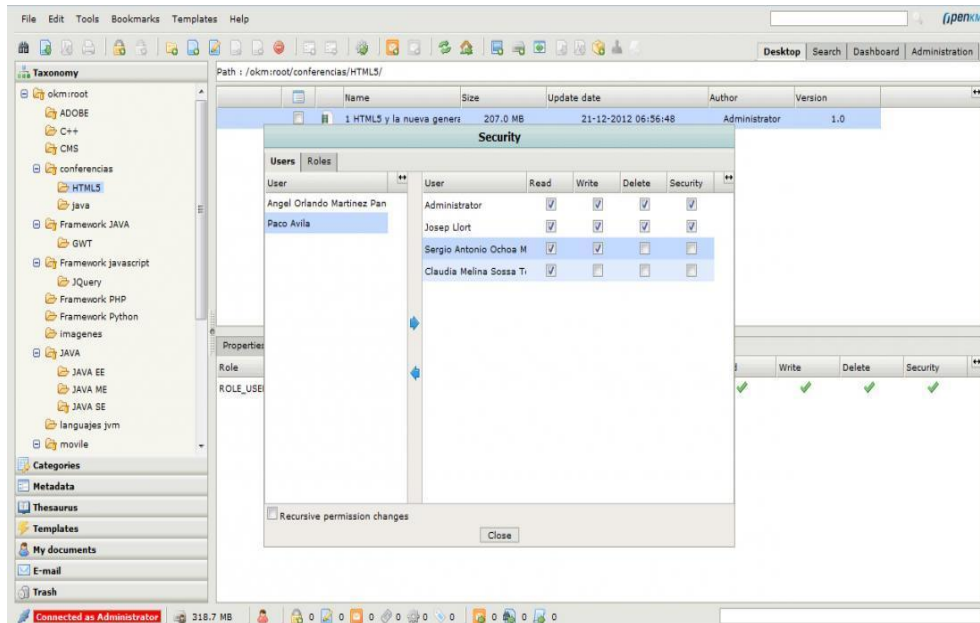


Figure 2.53 : Screenshot of OpenKM Security: Granular Access Control List

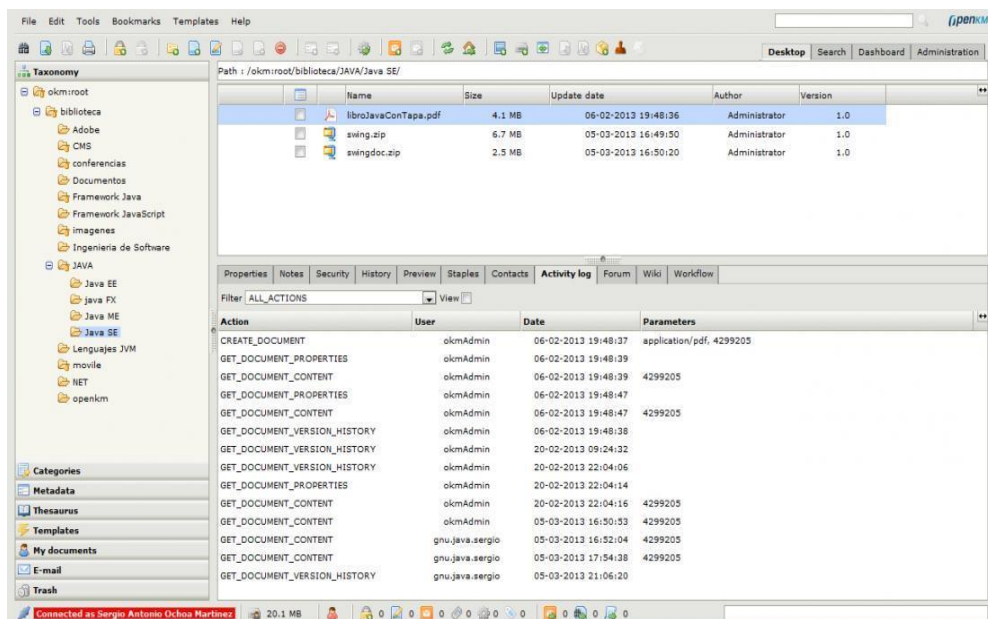


Figure 2.54 : Screenshot of OpenKM Log Per Document Detail

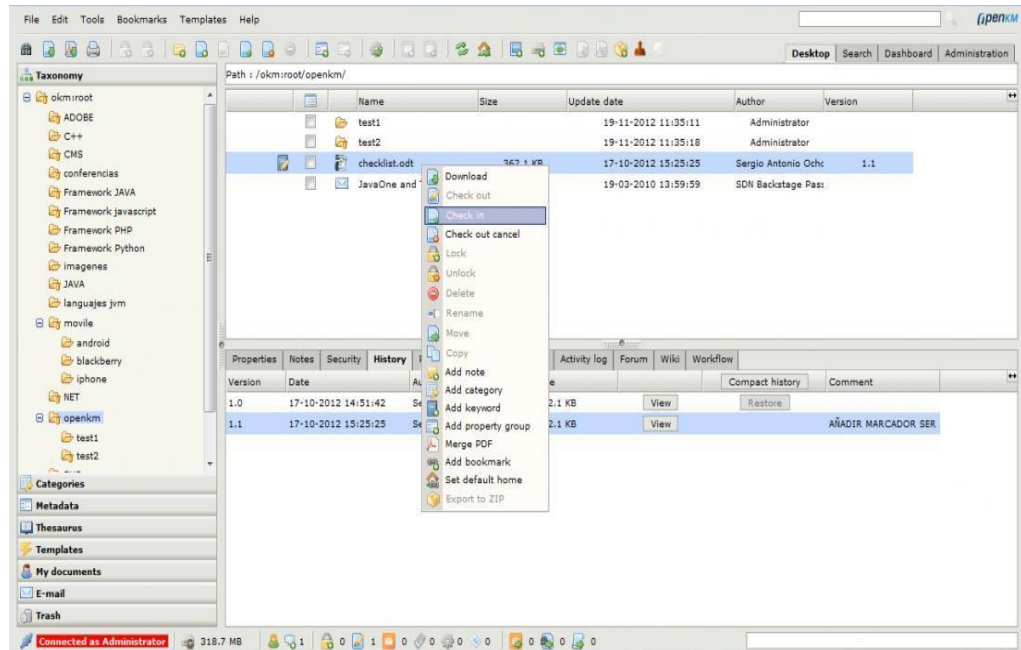


Figure 2.55 : Screenshot of OpenKM Version Control

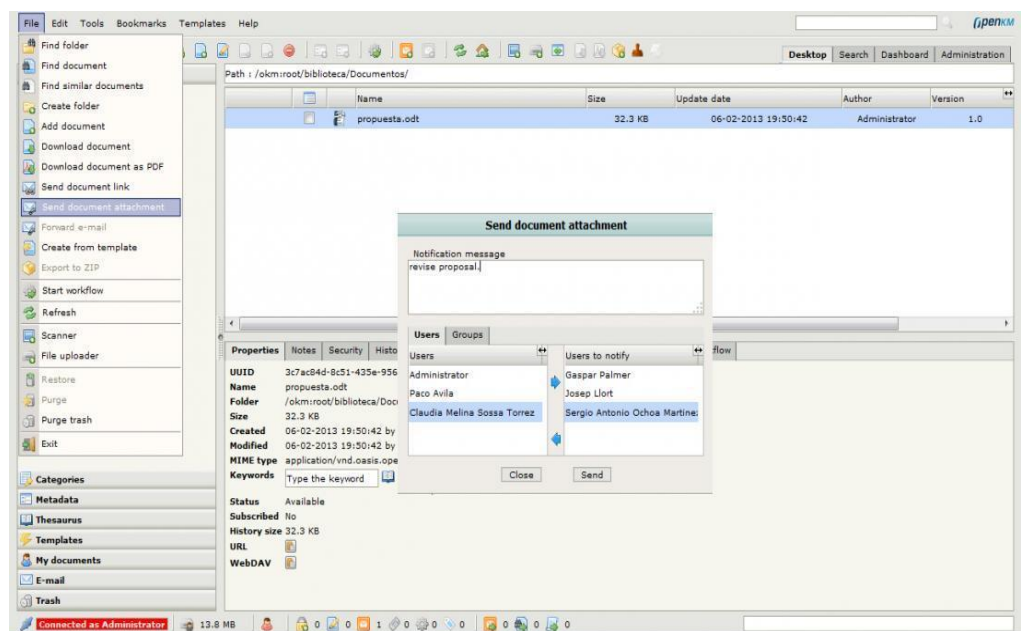


Figure 2.56 : Screenshot of OpenKM Notifications

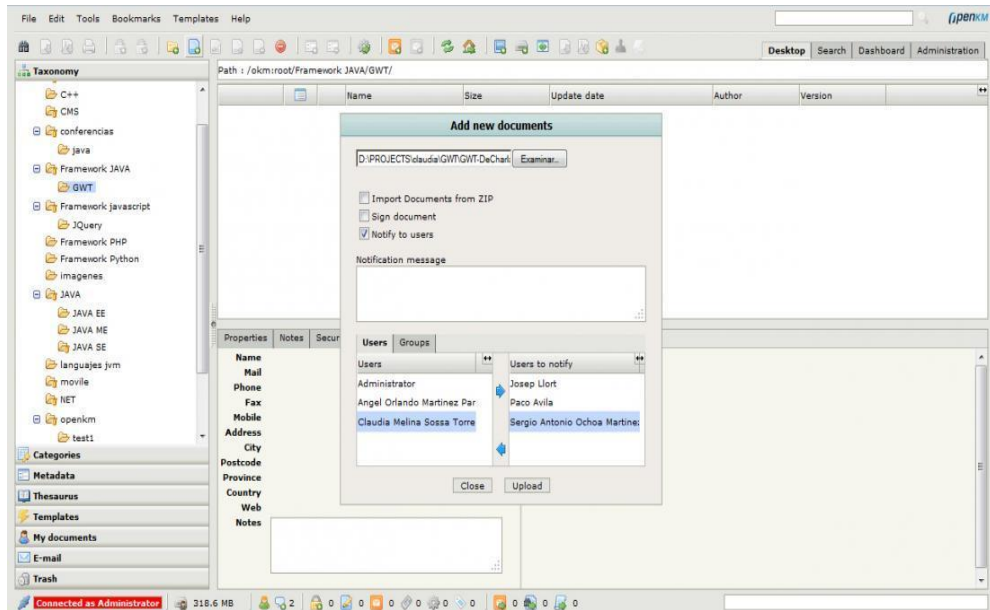


Figure 2.57 : Screenshot of OpenKM File Upload.

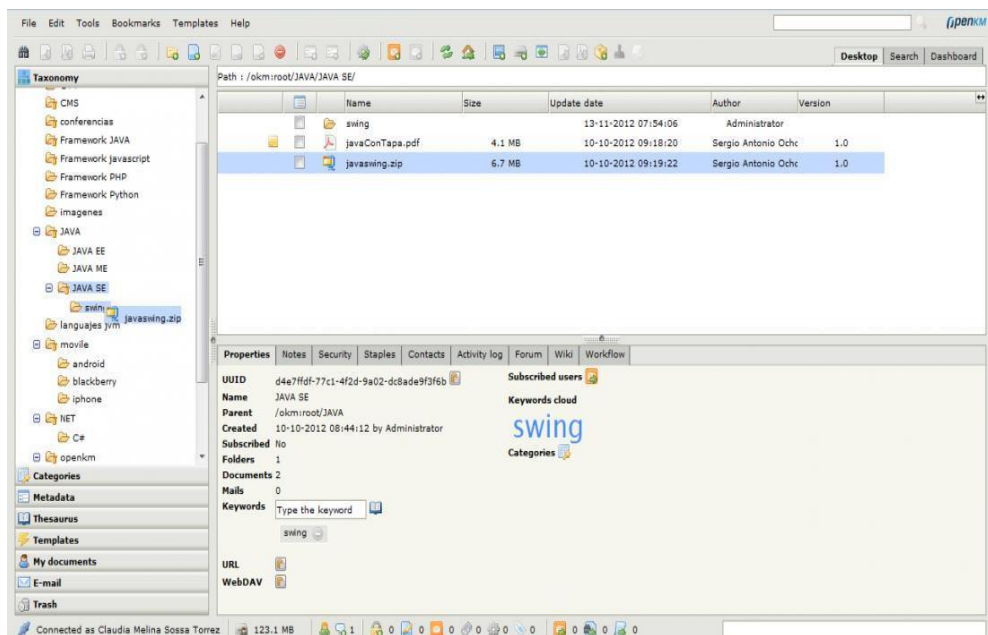


Figure 2.58: Screenshot of OpenKM Drag & Drop: Files or Documents from Desktop.

2.4.3 Papermerge

Papermerge is an open source document management framework that was created specifically for the purpose of working with scanned papers. It extracts text from your scans (which may be PDF, TIFF, JPEG, or PNG), indexes it, and makes it ready for complete text search. Papermerge is a file explorer that has the look and sound of a classic desktop file browser. Names, hierarchical directories, and automations are among the features that help the users manage the documents more effectively (Papermerge, 2021).

Papermerge features (Papermerge, 2021) are Optical Character Recognition (OCR), folder structure, coloured tags, full text search, give value with to the scan document with metadata, automate task, multi-user collaboration, page management, REST API, receipt handling, and support most of scanner output format.

Papermerge OCR technology is used to extract text from the scanned document, it will help users to convert hard copy documents to soft copy documents. Papermerge also gives metadata to documents for arrangement purposes.

Papermerge provides folder organisation to make the user files neater and more structured. To make document management interactive and effective, coloured tags feature are provided. The colour tag will help users to easily differentiate different folders and documents by just looking at it. The tags also can be pinned to make it easier to find specific documents. Another feature to make it easier for the user to find specific documents is full text search. Users can search specific documents just by typing some keywords.

Additionally, some features are provided by Papermerge to make effective document management systems like automation. Users can do task automation such as transferring records to the correct archive, metadata and extracted page. Papermerge also provides a paper management feature. Users can cut or paste pages between documents and rearrange the pages, and also users can delete the pages they want.

Papermerge also supports multiple-user collaboration. The owner of the document can set some permission for the document modification during the

collaboration. Papermerge also provides REST API which enables users to import documents from other sources such as SFTP account and email attachment. Papermerge also supports most scanner output formats including PDF, JPEG, PNG, and TIFF.

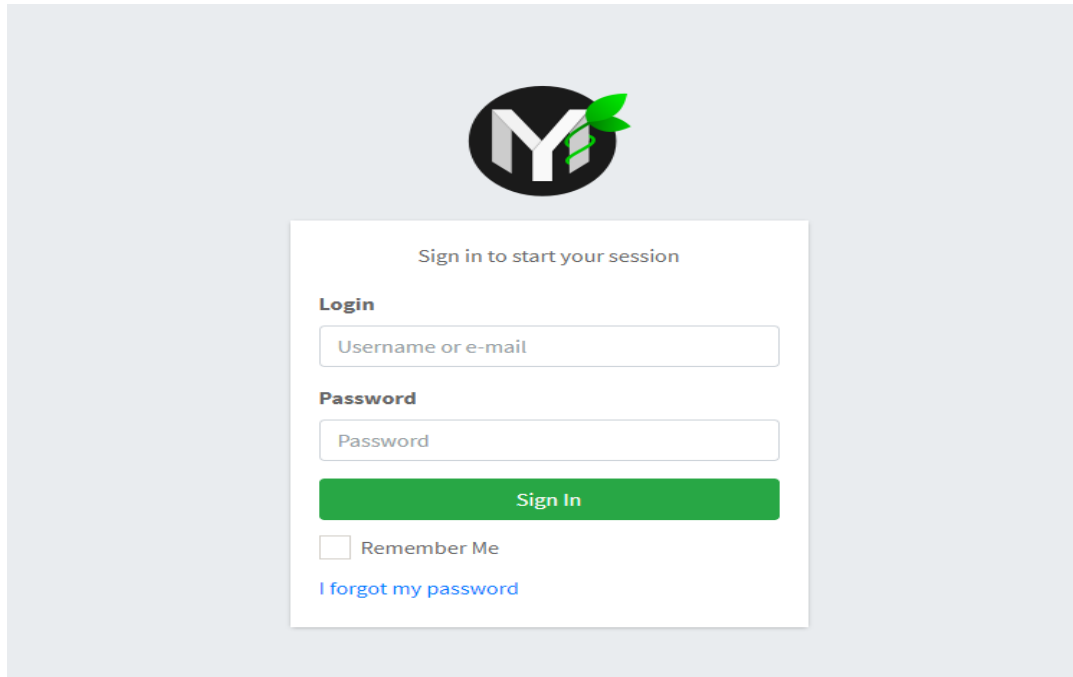


Figure 2.61 : Screenshot of Papermerge Login Page

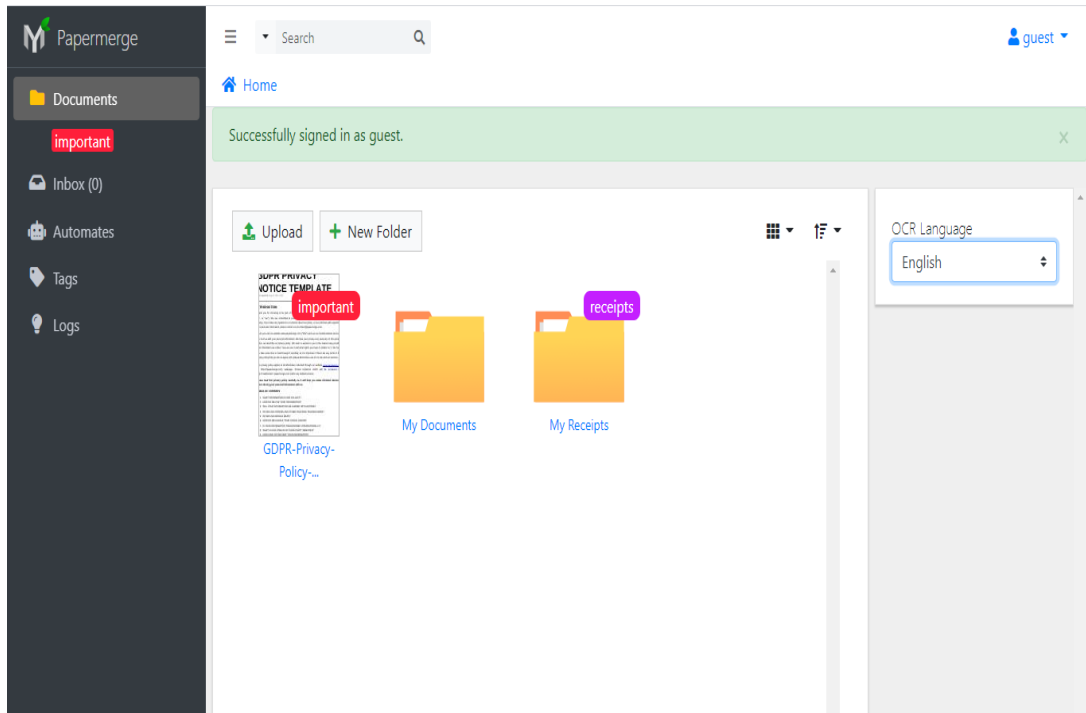


Figure 2.62 : Screenshot of Papermerge Home Page

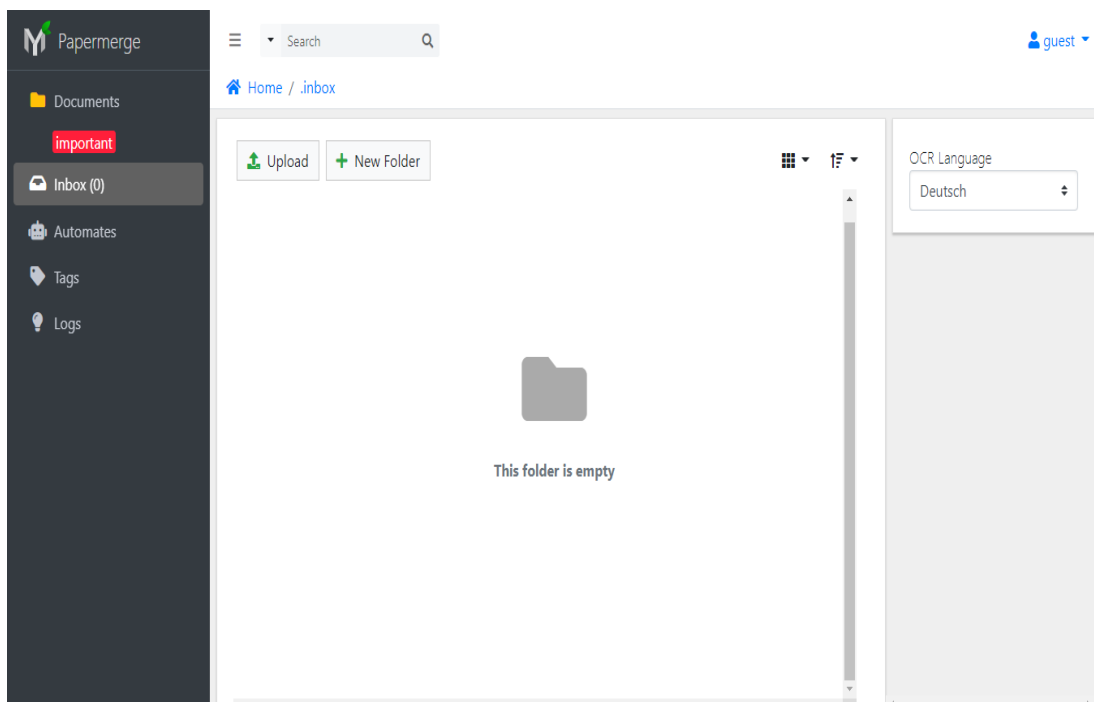


Figure 2.62 : Screenshot of Papermerge Inbox Page

Papermerge

Automates / New

Name:

Match:

Matching Algorithm:

Any

Is case sensitive: ☒

Tags:

These tags will be assigned to matched document.

Destination Folder:

Submit

Cancel

Figure 2.63 : Screenshot of Papermerge Automates Page

Papermerge

Tags / New

Name

Name

☐ Pinned

Pinned tag will be displayed under Documents menu. It serves as shortcut to quickly filter folders/documents associated with this tag

Foreground Color

Background Color

Description (optional)

Description (optional)

Submit

Cancel

Figure 2.64 : Screenshot of Paper Merge Tags Page.

Papermerge

Documents

important

Inbox (0)

Automates

Tags

Logs

Search

Q

guest

Logs

1234>

TIME	MESSAGE	USER	LEVEL
<input type="checkbox"/> 02.03.2021 08:32:51	Node(s) test were deleted. Node ids=[12]	guest	Info
<input type="checkbox"/> 02.03.2021 08:32:42	Node/Folder test created. Folder id=12.	guest	Info
<input type="checkbox"/> 02.03.2021 08:09:39	COMPLETE OCR for document GDPR-Privacy-Policy-Papermerge.pdf, p...	guest	Info
<input type="checkbox"/> 02.03.2021 08:09:20	STARTED OCR for document GDPR-Privacy-Policy-Papermerge.pdf, pa...	guest	Info
<input type="checkbox"/> 02.03.2021 08:09:20	COMPLETE OCR for document GDPR-Privacy-Policy-Papermerge.pdf, p...	guest	Info
<input type="checkbox"/> 02.03.2021 08:09:18	COMPLETE OCR for document GDPR-Privacy-Policy-Papermerge.pdf, p...	guest	Info

.....

Apply

Figure 2.65 : Screenshot of Papermerge Logs Page.

42

2.4.4 Existing System Comparison and Existing Systems' Features

Adoption to Proposed System

The existing systems' features in this study will be compared, and the useful features from each of the systems will be adapted to the proposed system according to the needs of the proposed system. The comparison is done in table 2.44.

Table 2.44 : Comparison of Existing System

System/ Features	E-course File UNISEL	OpenKM	Papermerge	Proposed System
Admin	✓	✓	✓	✓
Content Provider	✓	✓	✓	✓
Folder Structure	✓	✓	✓	✓
Colour Tags	x	✓	✓	✓
Preview	x	✓	✓	✓
Multiple User Collaboration	x	✓	✓	✓
Version Control System (VCS)	x	✓	✓	x
Logs	x	✓	✓	✓
Hybrid Encryption	x	✓	✓	✓

2.6 Conclusion

In conclusion, this chapter discusses the course file requirement to develop web-based systems for course file storage, examples of existing systems, and hybrid encryption. The course file requirement is obtained from FCI Course File Checklist. The checklist contains a specific folder that is required to upload a specific file. In the overview of the existing system, discuss the example of an existing system that used to store course files. Since, there is less example of system for course file system, another suitable file storage system is taken as example for this project. Examples of existing systems that are taken are UNISEL e-Course File System, OpenKM, and Papermerge. Hybrid encryption discusses the hybrid encryption that can be applied in the system to secure access to the system. The hybrid encryption discussed in this chapter is HAN encryption (Safi, 2017), Agrawal and Patankar proposed encryption algorithm's architecture, and SSL Certificate (X.509 Certificate). Based on the comparison that has been made, choosing SSL Certificate is the best choice due to time effectiveness and suitable for this project. SSL Certificate examples are OpenSSL, EasyRSA, and CFSSL.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter discusses the development methodology, SDLC for software development. This chapter also discusses software and hardware requirements that are needed for the development. Lastly, this chapter will discuss the testing methodology to find the web system vulnerability.

3.2 Selection of Development Methodology



Figure 3.1 Agile methodology diagram.

The methodology used in this project is agile. Agile methodology is chosen because it is very flexible to change. Thus, this methodology is suitable because it requires a change to follow the user requirement before the final launch.

The requirement phase was obtained from the FCI course file checklist. They from the course file checklist will be used to develop the web system. Then, to understand how the system should function, the user story will be taken from the interview with the head of the program and the lecturer.

After the requirement phase, the UI will be designed according to the requirement. Every design function in the web system will be based on the FCI course file checklist. The design will be following basic design principles and Norman's design principles to create interactive and user – friendly design.

The development phase will be developed using MVC architecture to connect between frontend, middle end, and backend. In this development phase, the chosen hybrid encryption will be implemented.

The testing phase is done to see the web system functional, and in this phase, the system will be tested for user acceptance by getting feedback after testing the system.

Then, the deployment phase will be releasing the first web system. The web system will be reviewed by the user, and any bugs and new requirements will be used for the web system improvement.

3.3 Software and Hardware Requirement

In this project, some software and hardware are required for the web-based system development.

Table 3.1 : Software Requirement

Software Requirement	Details
Xampp	Database, web server, PHP support
Figma	Software designing tools for User Interface (UI) design
Laravel	PHP Framework
Visual Studio Code	Text editor

In this project, there is no special hardware requirement, it just requires a personal computer (laptop). The hardware that will be used in this project is a laptop with processor I5 – 9 generation, 8 GB RAM, GPU NVIDIA GTX 1660, and Operating System Windows 10.

3.4 Research Methodology

The research methodology in this project is to find out the most secure SSL Certificate. The three types of SSL Certificate OpenSSL, EasyRSA, and CFSSL will be tested using an online lab to find out the details of the secure level for each SSL Certificate.

The SSL Certificate will be tested using an online lab called ImmuneWeb. The online lab will show the details of the security implementation of the SSL Certificate. Each of the certificates will be graded. Grade A+ will be more secure, and for less secure will be grade F.

3.5 Conclusion

Finally, the Agile Development Process would be the project's approach. This method involves six phases, including necessity, planning, production, testing, implementation, and evaluation, and each step has been clarified. Having a methodology in place for a project allows a seamless transfer between stages which gives visibility into the duration of each process. The software and hardware specifications detail the components used to build a web-based system for storing course files.

CHAPTER 4

SYSTEM ANALYSIS AND DESIGN

4.1 Introduction

This chapter will go through the system design. The web-based system for course file storage, as well as the prototype, will be comprehensively discussed. Data Flow Diagram (DFD), Entity Relationship Diagram (ERD), and Data Dictionary will be used to demonstrate the system design. The user interface (UI) section will demonstrate and explain the design of the system's UI. Finally, the purpose of the summary section is to recap the discussions in this chapter.

4.2 System Analysis and Design

System analysis and design (Tutorialspoint, 2021) is the research process for a system or its components in order to establish its goals. It is a problem-solving strategy that enhances the system and guarantees that all of the system's components work together to achieve the system's goal. Then, design a new business system or replace an old system by describing its components or modules to meet

the new system's particular needs. The system analysis and design will be created according to the user requirement that had been obtained from an interview with the potential user. The system analysis and design will help visualize the way the system should function and to ensure the fulfilment of user requirements.

4.2.1 Data Flow Diagram (DFD)

In this project, a Data Flow Diagram (DFD) is used to depict the data process. Data Flow Diagram (Lynch, 2019) is a method of illustrating information flows inside a system that is based on systematic analysis and design. It presents logic models and represents data transformation in a system, gives a framework for modelling information flow, and offers deconstruction to demonstrate particular data flows and operations. The Data Flow Diagram in the project will be represented in three levels: context diagram, DFD level 1, and DFD level 2.

i. Context Diagram

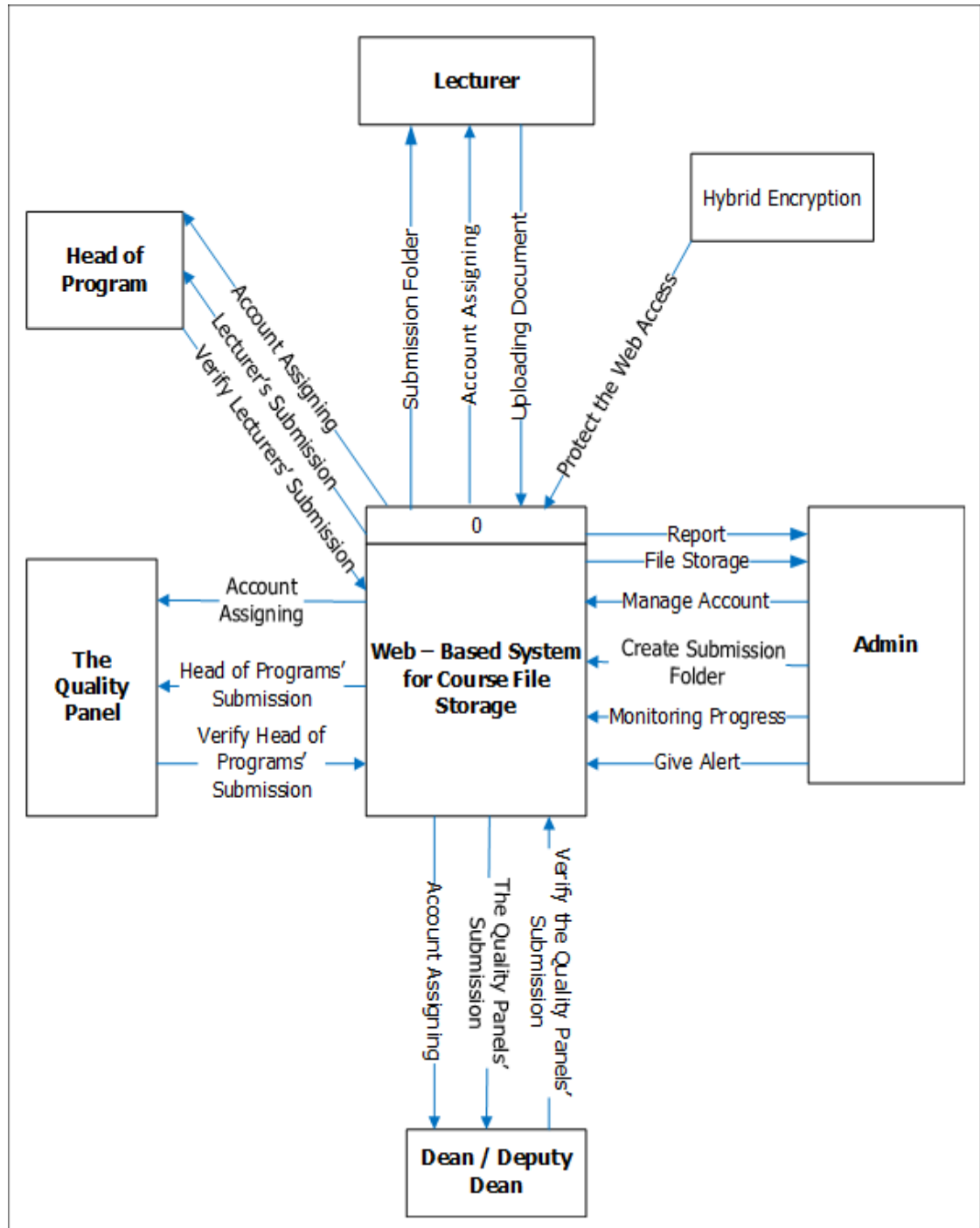


Figure 4.1: Context Diagram

The Context Diagram represents how the users will interact with the system. In this system the users are lecturers, head of programs, the quality panels, dean / deputy dean, and admin. Each of the users has different tasks that are assigned to them in this system. Admin task is to manage user accounts, create submission folders,

monitor progress, and give alerts to the user if the submission documents due date is under the corner. The system will assign the admin with a report about the other users' progress and file storage, after the file has been verified by the dean / deputy dean. Lecturer, head of program, the quality panel, and dean / deputy dean is assigned with an account according to their position. The users also assign the specific task that they need to do. Lecturer will assign a submission folder that they have to upload related documents. The head of the program will be assigned with lecturers' submission documents that need to be verified. The quality panel will be assigned with the head of programs' submission documents that need to be verified. Lastly, the dean / deputy dean will assign the quality panel submission that needs to be verified and finally stored.

ii. DFD Level 1

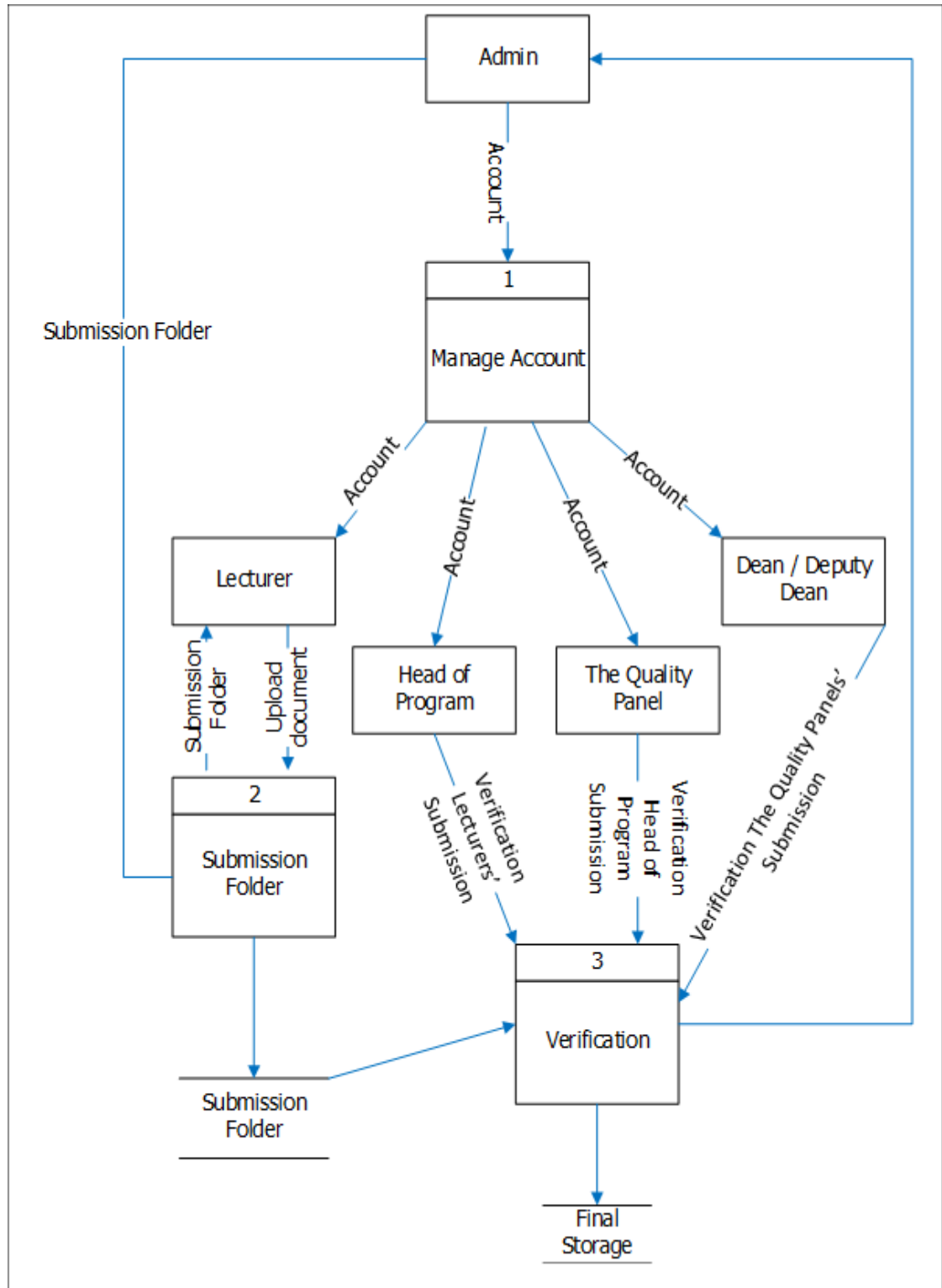


Figure 4.2: DFD Level 1

Figure 4.2 depicts the basic procedure of preparing a course file in this system. The first step is for the admin to create accounts for each of the following users: lecturer,

head of program, quality panel, and dean / deputy dean. The admin will establish a submission folder, which must be filled with course-related documents and uploaded by the lecturer. The submission folder will then be sent to the verification procedure, where it will be reviewed by the program's head of program. Then, the quality panel will verify the procedure for verifying the head of program submission. The procedure will then be reviewed by the dean / deputy dean to ensure that the quality panel submission is acceptable for final storage of the course file.

iii. DFD Level 2

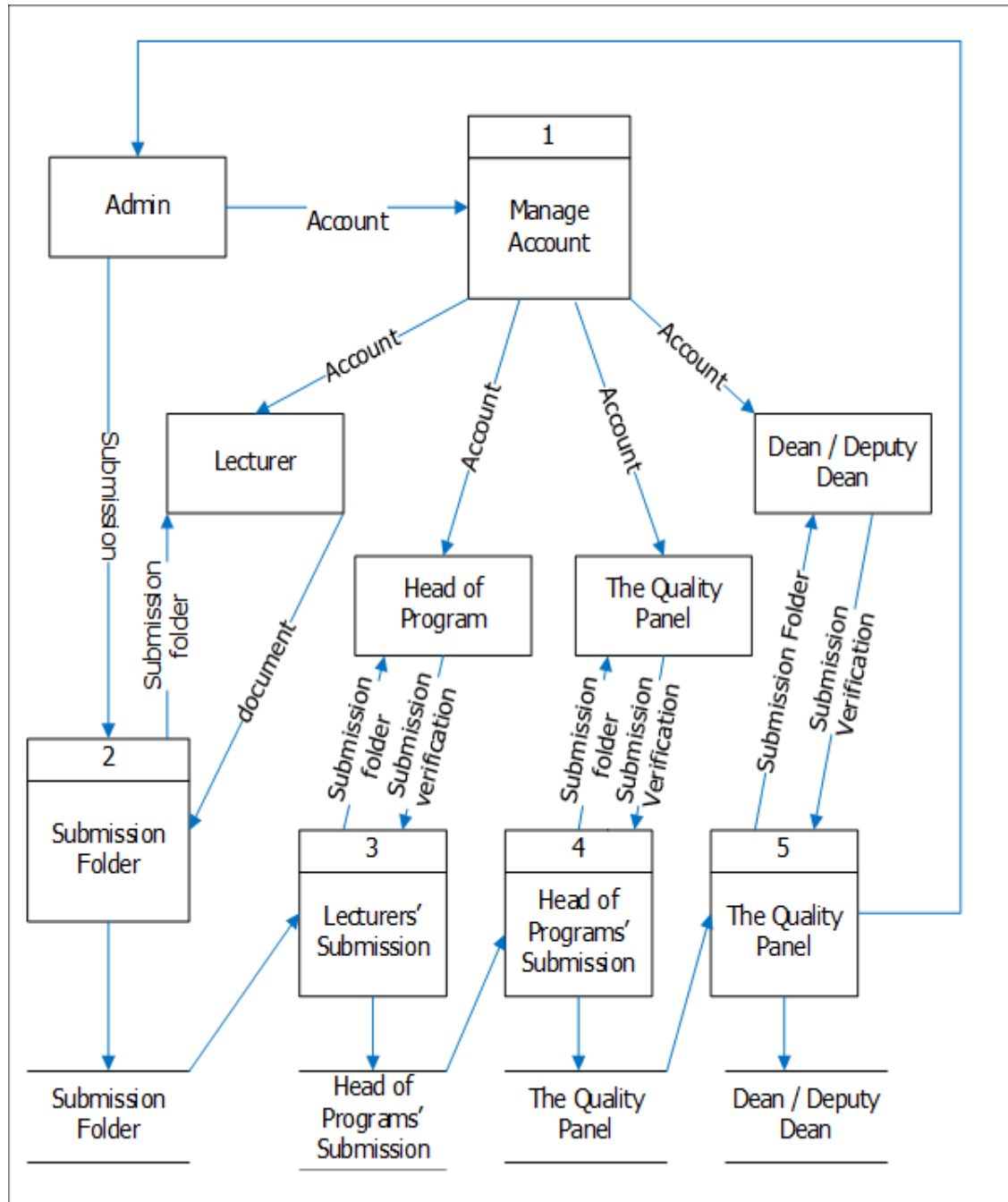


Figure 4.3: DFD Level 2

Figure 4.3 illustrates the data flow for course file preparation. The process is same like DFD level 1, the different is the process of submission course file related document that need to be upload by lecturer, then, will be verify by head of program, the quality panel, and dean / deputy dean. The process of submission verification is explained in more detail in this DFD level 2.

4.2.2 Entity Relationship Diagram (ERD)

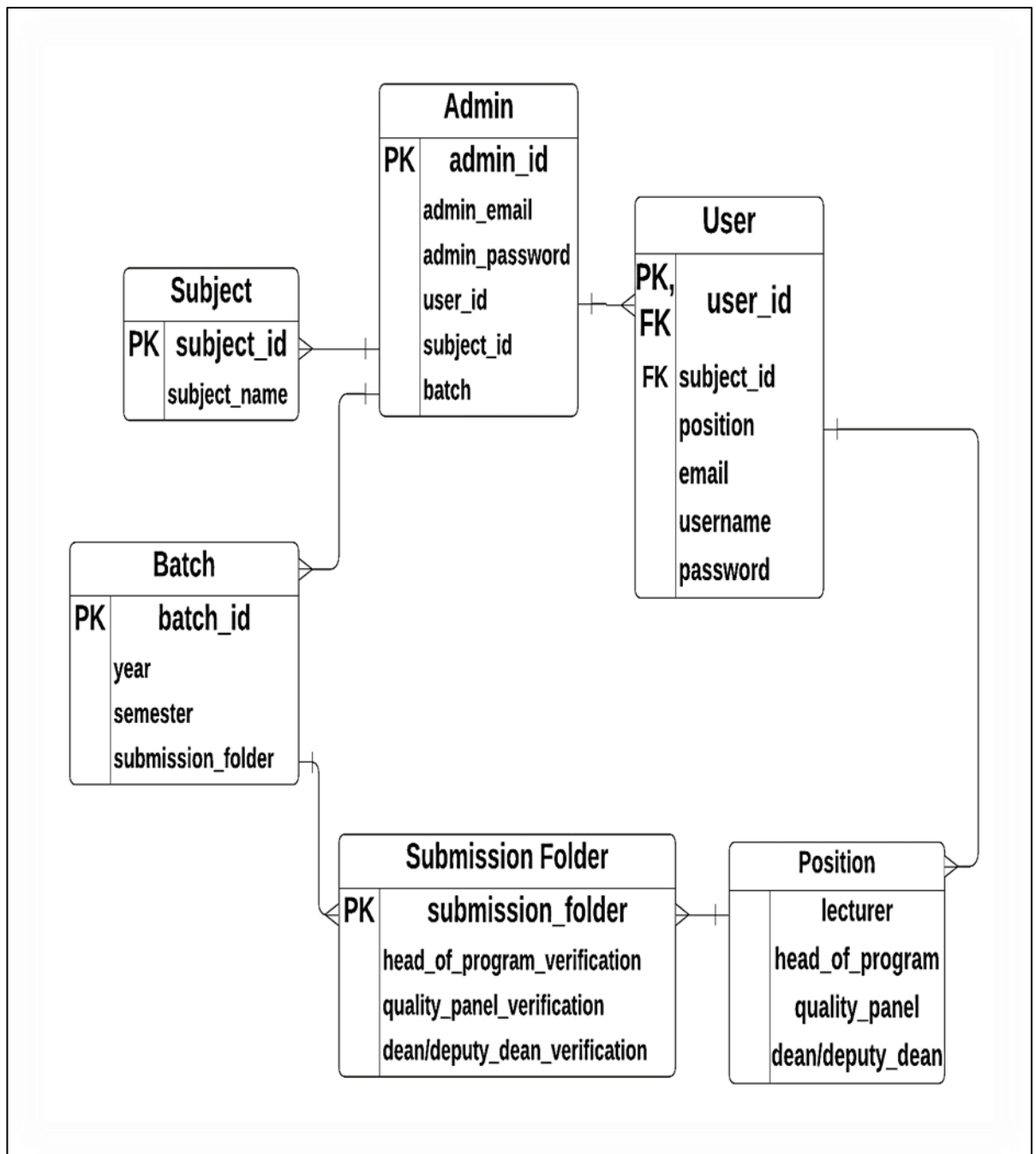


Figure 4.4: Entity Relationship Diagram (ERD)

Figure 4.4 illustrates the entity relationship diagram between the attributes on the system. The system contains admin, subject, user, batch, submission folder, and position.

4.2.3 Data Dictionary

Data dictionary (Techopedia, 2021) is a file or a collection of metadata files containing a database. The data dictionary stores information about other items in the database, such as data ownership, data linkages to other objects, and other data. The data dictionary included in this chapter will provide information on the data that will be used in this project and will be explained in detail. The data dictionary is will be described in the terms of field name, data type, field size, description and example.

The data dictionary for admin information will be described in Table 4.1:

Table 4.1 : Data Dictionary of Admin

Field Name	Data Type	Field Size	Description	Example
admin_id	Varchar	20	Unique ID for admin	BI1811XXXX
admin_email	Varchar	50	Email for admin	admin1@ums.edu.my
admin_password	Varchar	30	password for admin	Password123 Test\$\$#
user_id	Varchar	20	Unique ID for the user	BI180XXXXXX
subject_id	Varchar	10	Unique ID for each subject	KP00XXX
batch_id	Varchar	20	Unique ID for current batch (year and semester)	2021SEM2

The data dictionary about batch details will be explained in the Table 4.2:

Table 4.2 : Data Dictionary of Batch

Field Name	Data Type	Field Size	Description	Example
batch_id	Varchar	20	Unique ID for batch creation	2021SEM2
year	YEAR	4	Year of current batch	2021

semester	int	1	Current semester, describe as semester 1 or semester 2	1
submission_folder	BLOB	1073741824	The place to store file	.pdf .docx

The data dictionary about subject details will be explained in the Table 4.3:

Table 4.3 : Data Dictionary of Subject

Field Name	Data Type	Field Size	Description	Example
subject_id	Varchar	10	Unique ID for each subject	KP00XXX
subject_name	text	30	subject name	Network Security

The data dictionary about user details will be explained in the Table 4.5:

Table 4.5 : Data Dictionary of User

Field Name	Data Type	Field Size	Description	Example
subject_id	Varchar	10	Unique ID for each subject	KP00XXX
subject_name	Text	30	subject name	Network Security
position	Enum	4	Type of position that is held by the FCI staff: lecturer, head of program, the quality panel, and dean / deputy dean.	Lecturer
email	Varchar	50	email for user	user@ums.edu.my
username	Text	30	username for user	Dr User
password	Varchar	30	password for user	Password123***###

The data dictionary about submission folder details will be explained in the Table 4.6:

Table 4.6 : Data Dictionary of Submission Folder

Field Name	Data Type	Field Size	Description	Example
submission_folder	BLOB	1073741824	The place to store file	.pdf .docx
head_of_pogram_verification	BLOB	1073741824	The place to store file	.pdf .docx
quality_panel_verification	BLOB	1073741824	The place to store file	.pdf .docx
dean/deputy_dean_verification	BLOB	1073741824	The place to store file	.pdf .docx

4.3 User Interface (UI) Design

This system will have a user interface to make the system simpler to use. The user interface is built on Norman's design principles. Visibility, feedback, constraints, mapping, consistency, and affordance are Norman's design principles (Rekhi, 2017). The visibility principle is used in this UI design by selecting a colour that provides sufficient contrast between the backdrop and the foreground element. Use a bright colour for the backdrop and a darker colour for the foreground area. Glassmorphism UI is also used in the design idea. Glass Morphism (Malewicz, 2020) is a kind of user interface that employs a frosted-glass look achieved by creating a translucent and hazy backdrop. The feedback principle is shown by the system when a user completes a certain activity and the system sends feedback such as submission success. The constraints concept is shown in the system by employing an icon as a signal or hint of the activity that the user must do. The mapping idea in this system is shown by arranging each piece in a structured manner. The use of the same icon, colour, and element in the design demonstrates the consistency concept in this system. Finally, the idea of affordance is shown by utilising a dropdown menu for the selection

information that the user must pick and an icon as a sign and hint for the user to comprehend what action should be made by the user.

The user interface (UI) in this system is separated into categories depending on the user's role. The login page, admin page, lecturer page, head of program page, quality panel, and dean/deputy dean are all part of the UI category.

i. Login Page

In this system, the login page is the initial UI that the user encounters on entering the system. For the first time user, admin will provide a user name and password that must be changed afterwards. Sign up is not possible in this system since the system is only accessible to FCI professionals involved in course file preparation. As a result, the admin will allocate accounts to the staff members who are involved in the course file storage based on their position.

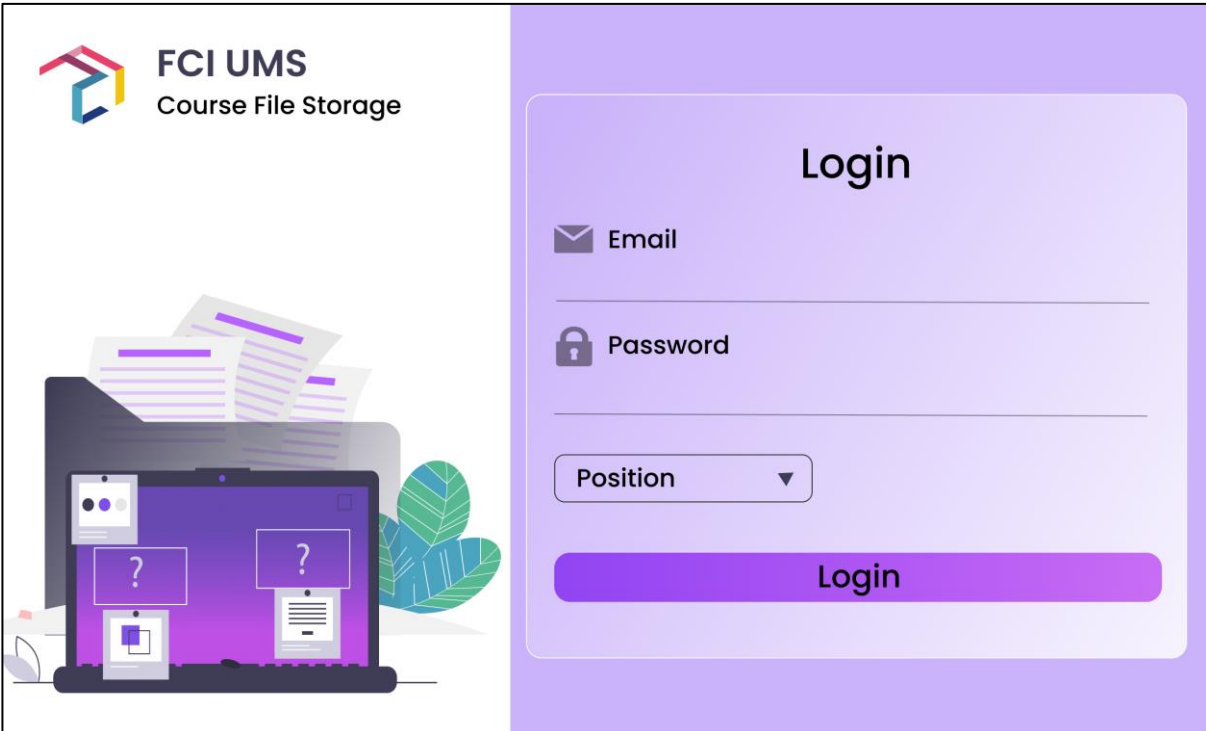


Figure 4.4: Login Page

This system's login page includes an email textbox, a password field, and position information. This system will only receive UMS email from users; any other email will

be rejected by the system. In this system, the positions available are admin, lecturer, head of program, the quality panel, and dean / deputy dean.

ii. Admin Page

The admin page in this system contains an admin dashboard, user accounts, folder, and events.

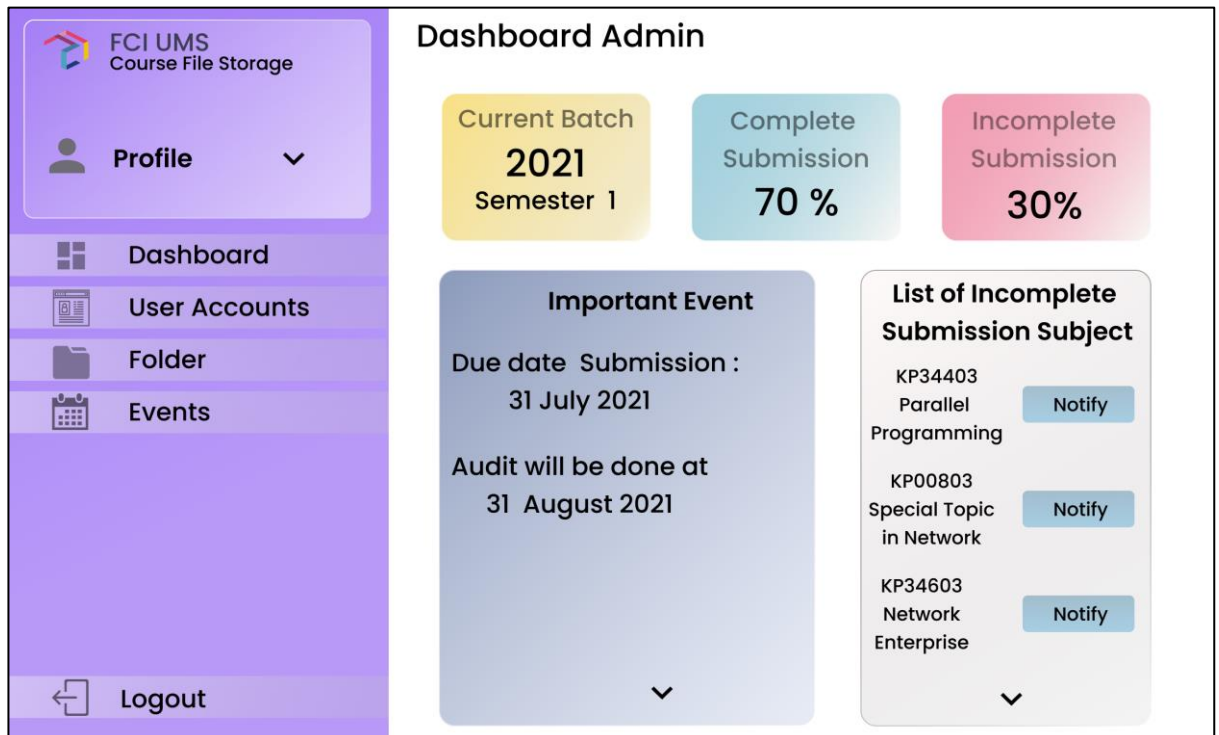






Figure 4.5: Admin Dashboard


The admin dashboard displays the current batch of the year and semester, complete submission of course file, incomplete submission of course file, list of incomplete submission subjects that need to be notified by the admin when the due date approaches, and the important event as a reminder to the admin about the important event related to course file preparation.



FCI UMS
 Course File Storage



Profile

 **Dashboard**

 **User Accounts**

 **Folder**

 **Events**

 **Logout**

User Accounts

ID	Name	Email	Password	Position
1	User 1	user@ums.edu.my	*****	Lecturer <input checked="" type="checkbox"/> Head of Program <input checked="" type="checkbox"/> The Quality Program <input checked="" type="checkbox"/> Dean / Deputy Dean <input type="checkbox"/>
2	User 2	user2@ums.edu.my	*****	Lecturer <input checked="" type="checkbox"/>







 1
 


Figure 4.6: Admin's User Accounts


The admin's user accounts include information on the user accounts, such as the staff ID, name, email, password, and the user's position. Admins may create new users, edit existing users' information, and remove user's accounts.



FCI UMS
 Course File Storage



Profile

 **Dashboard**

 **User Accounts**







 **Folder**

 **Events**

 **Logout**

Folder

Add Batch
Add Subject

Year	2021	Semester	2		
Subject Code	Subject Name	Lecturer	Action		
KP34401	Network Security	Lecturer 3			
KP00703	Network Enterprise	Lecturer 4			










Year	2021	Semester	1		
Subject Code	Subject Name	Lecturer	Action		
KP34403	Parallel Programming	Lecturer 1			
KP00803	Special Topic in Network	Lecturer 2			


Figure 4.7: Admin's Folder


The admin's folder contains the submission of course files, which is used by the admin to create new submissions based on the current batch. The submission folder will be established based on the subject and assigned to the lecturer who teaches the subject. The administrator may also track the status of each subject's course file submission.


 FCI UMS
Course File Storage

 Profile ▾

 Dashboard




 Folder

 Notification

 Logout

← Subject Folder


2021Semester 2



Subject code	Subject Name	Folder	Completion Status
KPxxx03	Basic Programming		Pending at Lecturer
KPxxxxxx	Technology in Network		Pending at Head of Program
KPXXXX	Network Security		Completed


▼


Figure 4.7: Admin’s Subject Folder


The admin's topic folder contains the status of course file uploads for each subject. The pending status indicates where the uploading of a file is still awaiting. When a lecturer has a pending status, it signifies that the lecturer has not finished the course file uploading. When there is a pending status at the head of programs, the quality panel, or dean / deputy dean, it means that the professor's submission has not been verified by the FCI executives. When the submission is complete, the course file will be saved in this folder and accessible to the administrator.



FCI UMS
 Course File Storage



Profile


 **Dashboard**

 **User Accounts**

 **Folder**

 **Events**

 **Logout**

Events

Add Events













Date	Activity	Action
30/6/2021		  
1/7/2021		  
10/7/2021		  
30/7/2021		  

Figure 4.8: Admin's Events

The admin events are significant events linked with the preparation of course file storage. Administrators may use this page to keep track of all events related to course file storage and use it as a reminder when an important occasion is approaching.

iii. Lecturer Page

The lecturer page contains the lecturer dashboard, submission folder, subject folder, uploading page, and notification.

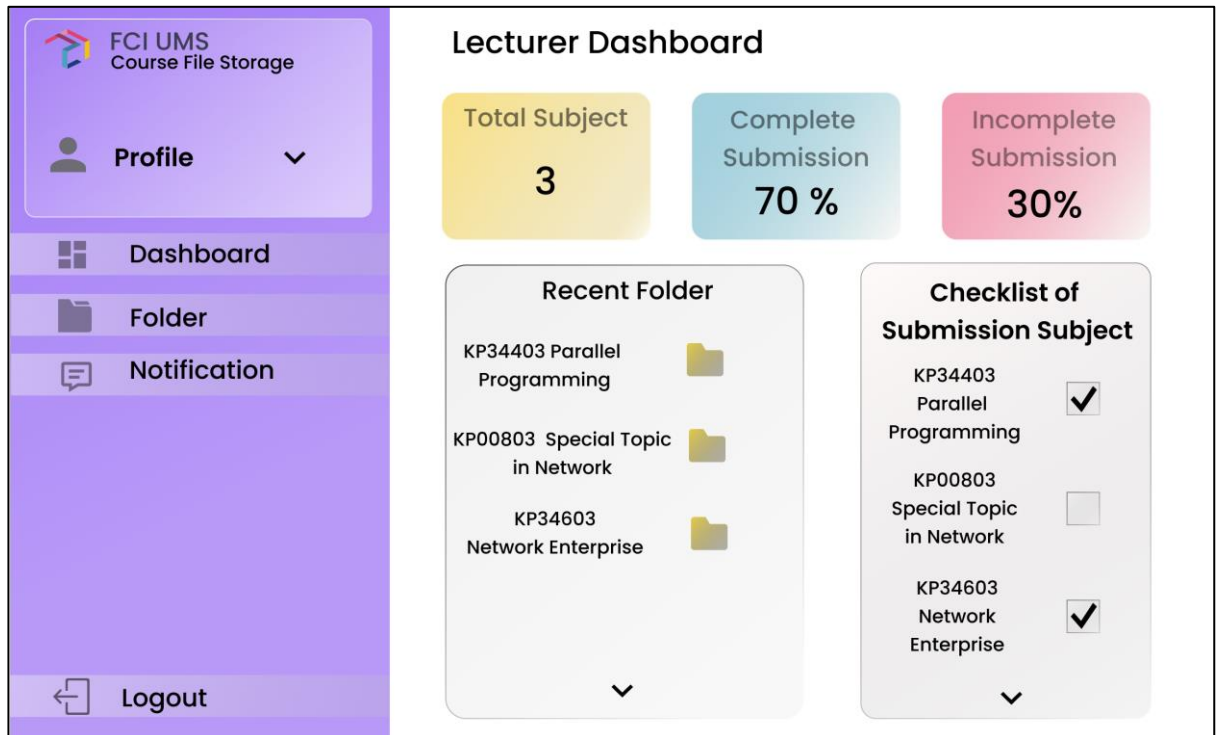


Figure 4.9: Lecturer Dashboard

The lecturer dashboard displays the overall number of subjects taught by the lecturer, as well as the overall proportion of full and incomplete course file submissions. This website also includes a checklist to help the lecturer determine which subjects have still to be completed for course file submission.

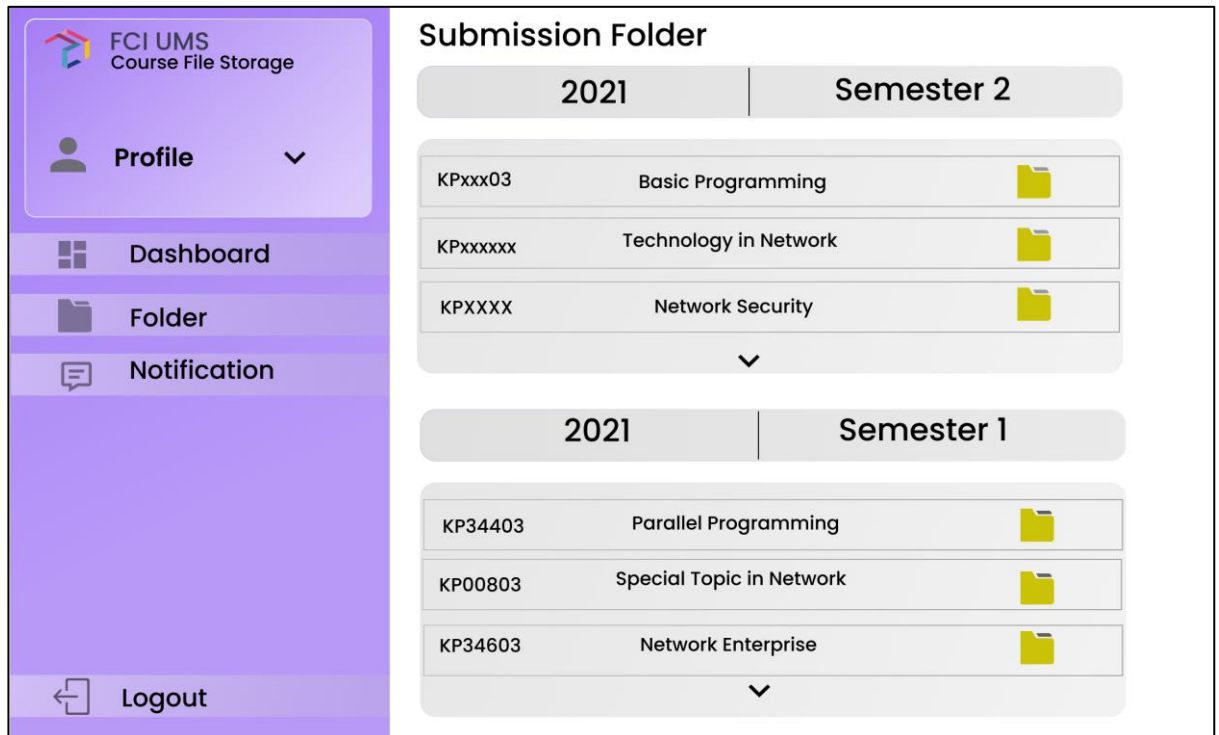


Figure 4.10: Lecturer's Submission Folder

The lecturer's submission folder includes the batch of the submission folder. Each of the subjects includes a special folder that needs to be uploaded with a course file associated with the document. The number of the topic folder is depending on the lecturer's teaching subject.

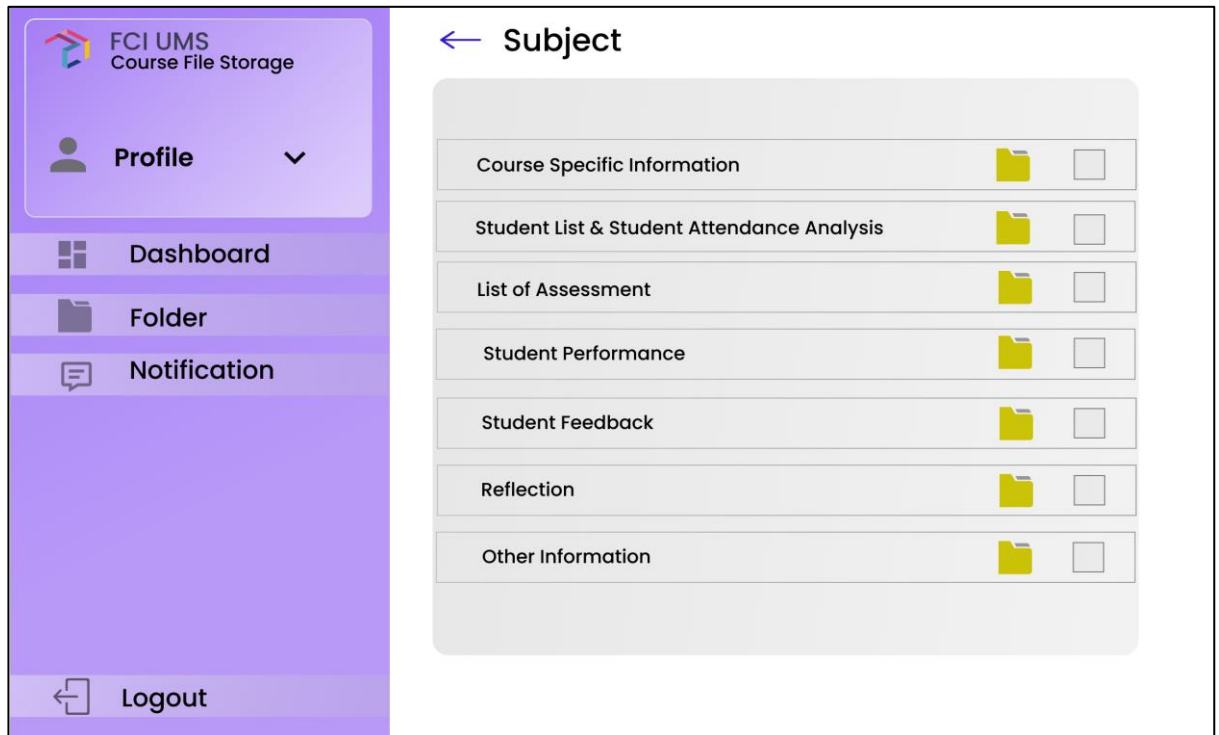


Figure 4.11: Lecturer's Subject Folder

The lecturer's subject folder contains folders for each kind of document associated with the course file, as defined by the FCI Course File Checklist. Each folder must be uploaded along with the associated document. When the instructor has completed uploading, the checklist box will be marked with a tick (✓).

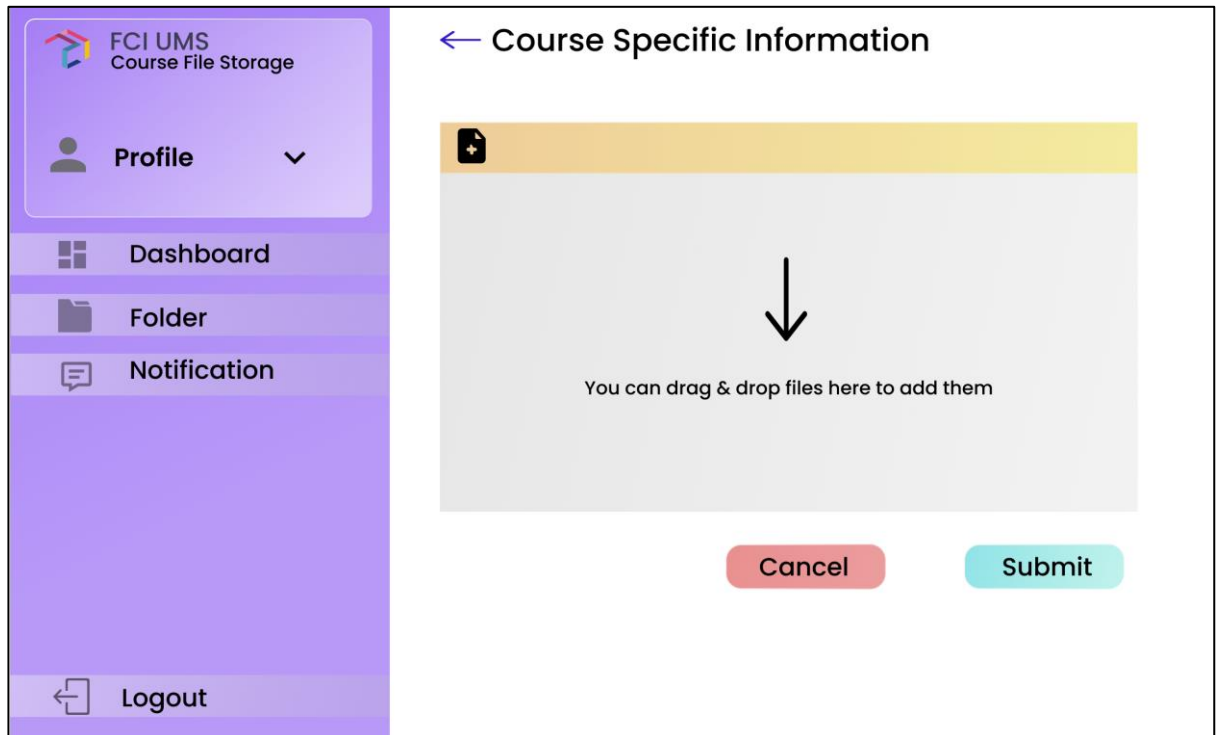


Figure 4.12: Lecturer's Uploading Page

The lecturer's uploading page is where lecturers upload course file materials. Lecturers may upload files to this page by selecting the file icon and searching for the file they want to upload, or by dragging and dropping the file into the uploading box.

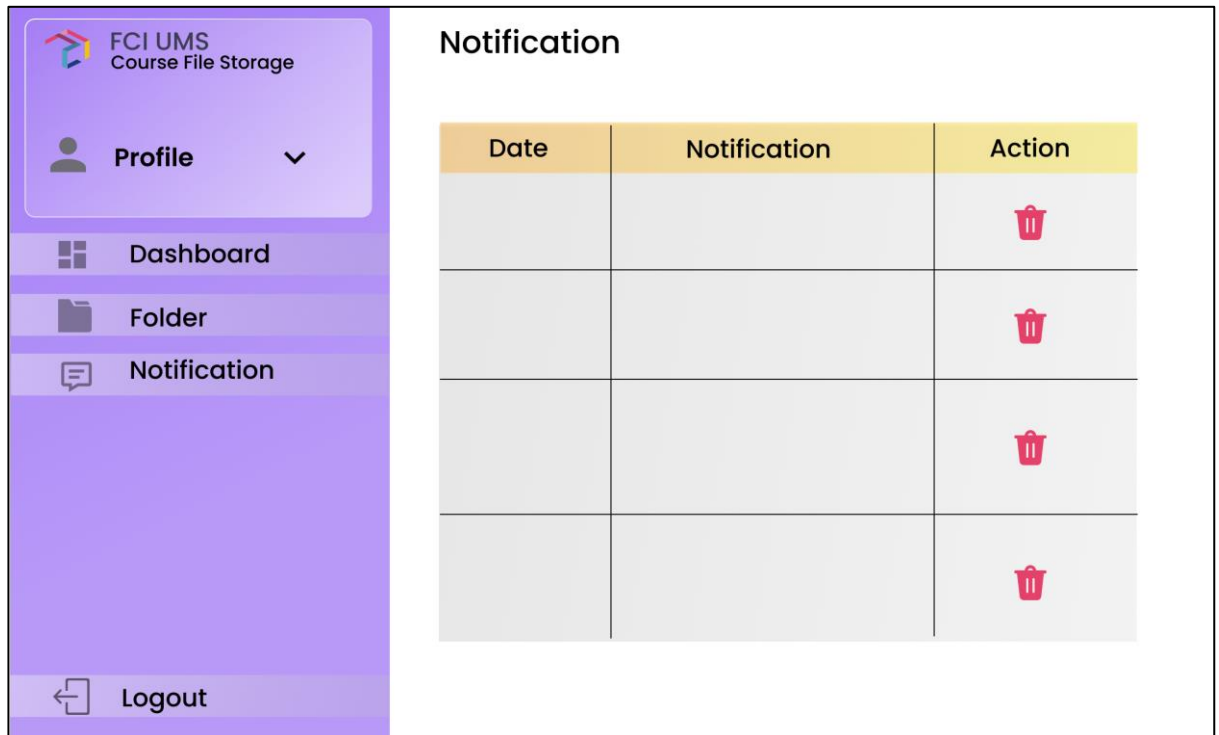


Figure 4.13: Lecturer's Notification Page

This page includes notice to the lecturer from the admin and other FCI leadership regarding whether the course file has any difficulty.

iv. Head of Program Page

The head of program page contains a dashboard for the head of program, a folder containing course-related documents submitted by lecturers that the head of program must check and validate. This page also includes notifications.

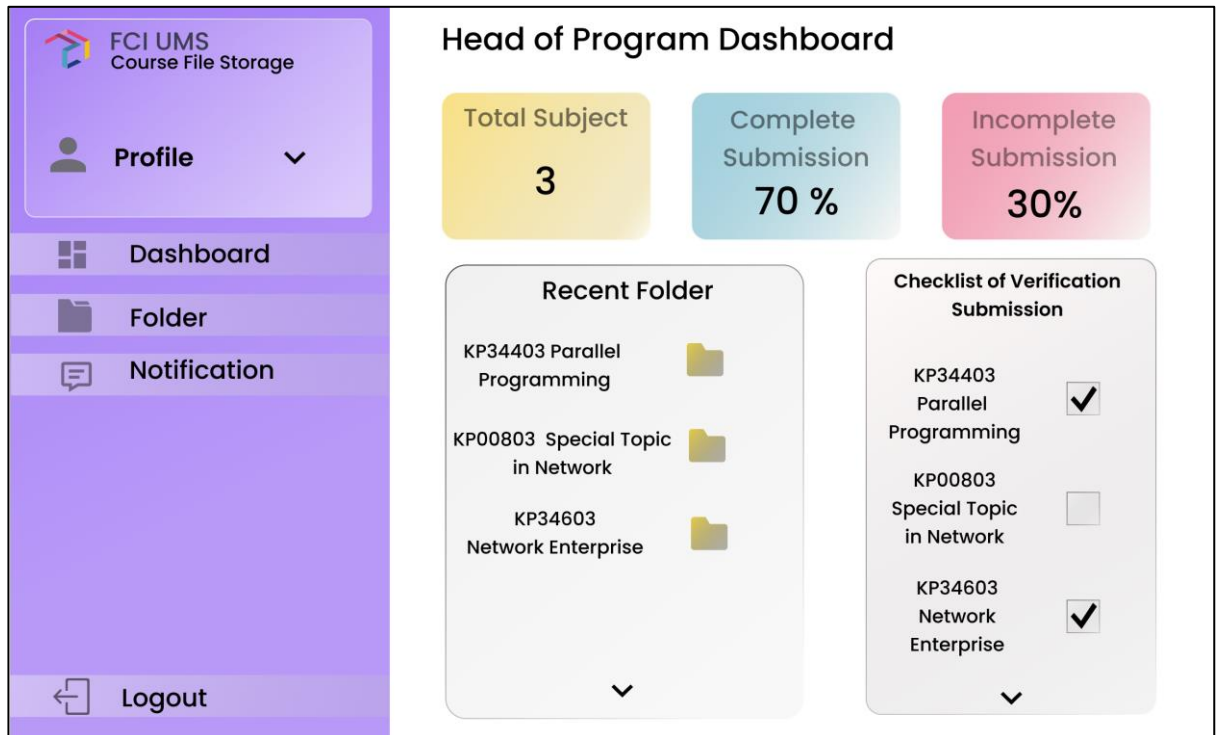


Figure 4.14: Head of Program Dashboard

The head of the program dashboard displays the total number of subjects that the head of the program needs to examine and verify, full and incomplete submission folder verification, and a checklist of the full verify folder to help the head of the program determine which submission folders have not been verified.

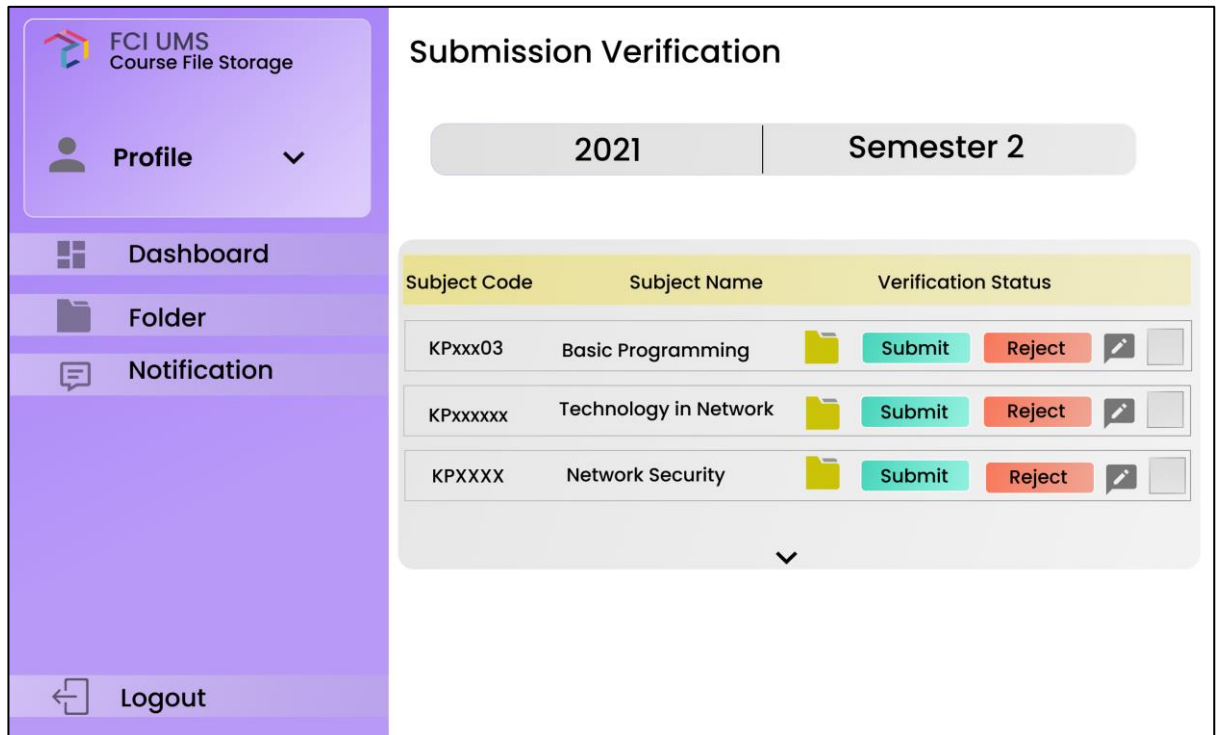


Figure 4.15: Head of Program’s Submission Verification Page.

The submission verification page includes a list of subjects that must be verified by the head of the program. If the submission folder is satisfactory to the head of the program, the document may be sent to the quality panel once it has been checked. However, if the submission folder has a problem, the head of the program can reject it and request that the lecturer submit a new submission. This folder also has a remark option, which the head of program may utilise to provide feedback to lecturers regarding course file submissions or to the quality panel. Additionally, the website has a checklist box; after the head of the program has completed the checking submission folder, the checklist will be marked with a tick.

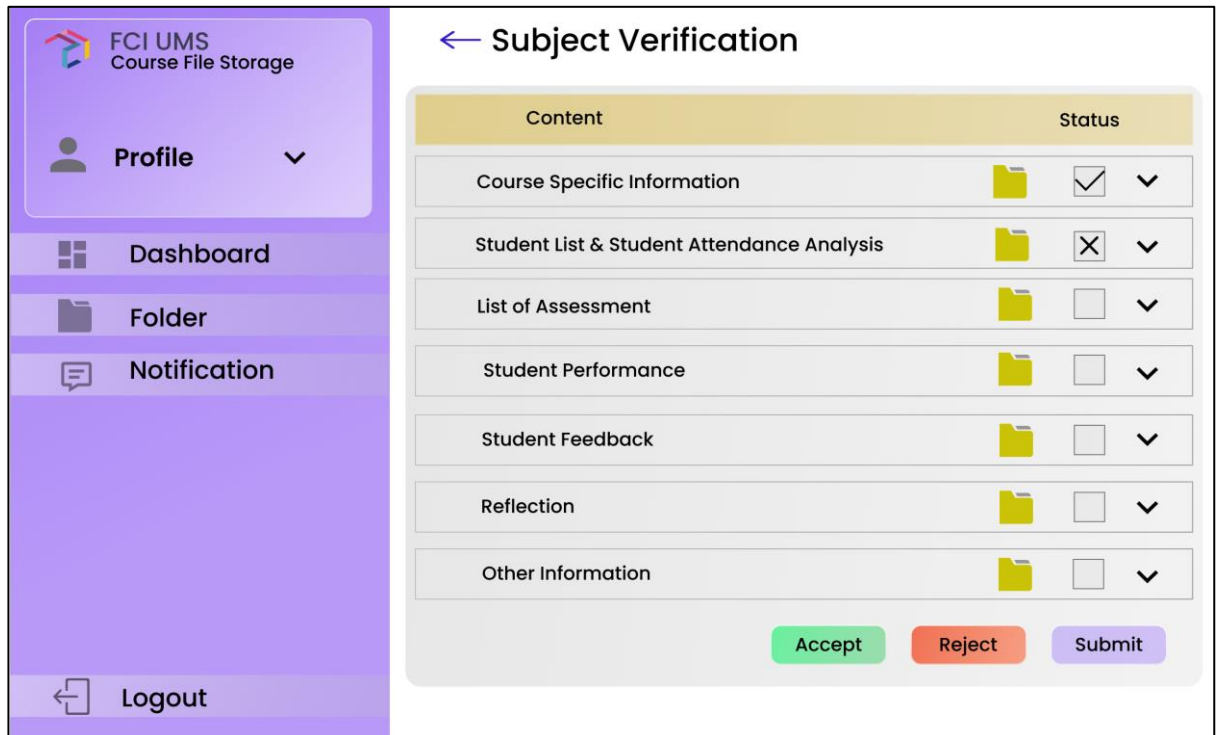


Figure 4.16: Head of Program's Subject Verification Page

The head of the program's subject verification page is the content of every subject folder that needs to be checked by the head of the program. When all the folders have been checked and show no issue then only the head of the program can accept the submission, otherwise the head of the program can reject it.

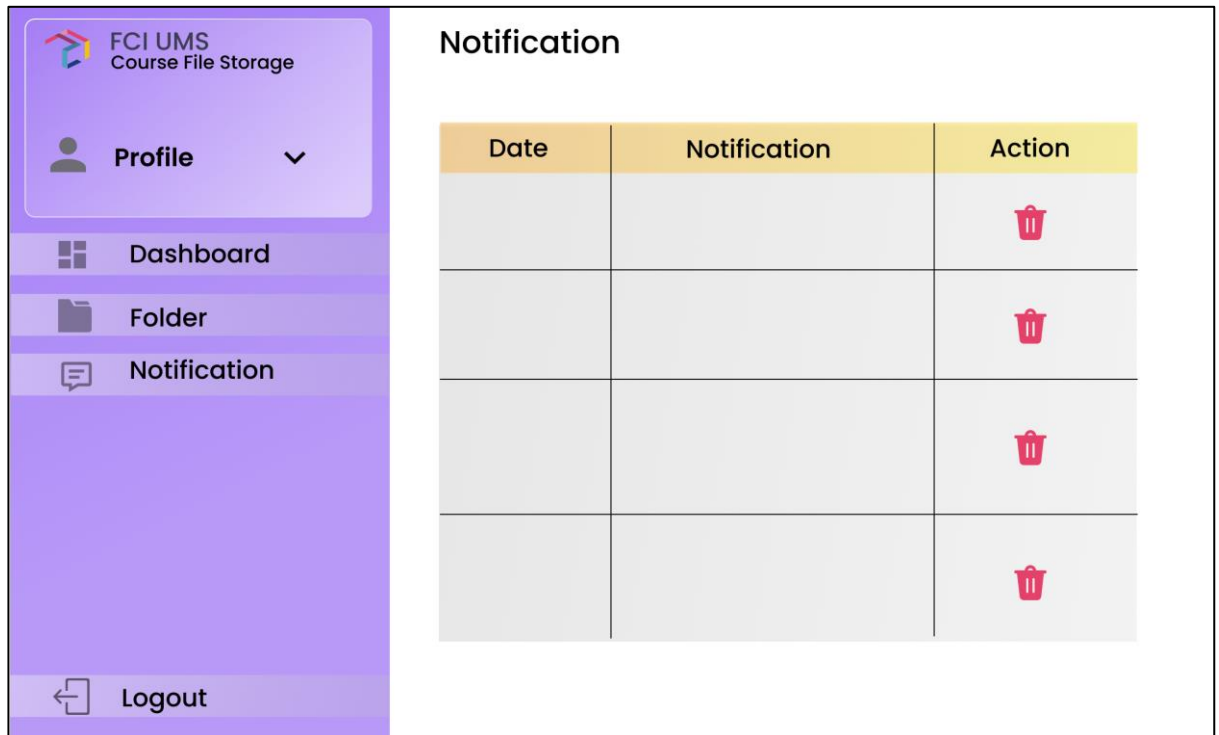


Figure 4.17: Head of Program’s Notification Page

This page includes notification to the head of the program from the lecturer, admin and other FCI leadership regarding whether the course file has any issue.

v. The Quality Panel Page

The quality panel page contains a quality panel dashboard, a folder containing course-related documents submitted by the head of the program that will be checked and validated. This page also includes notifications.

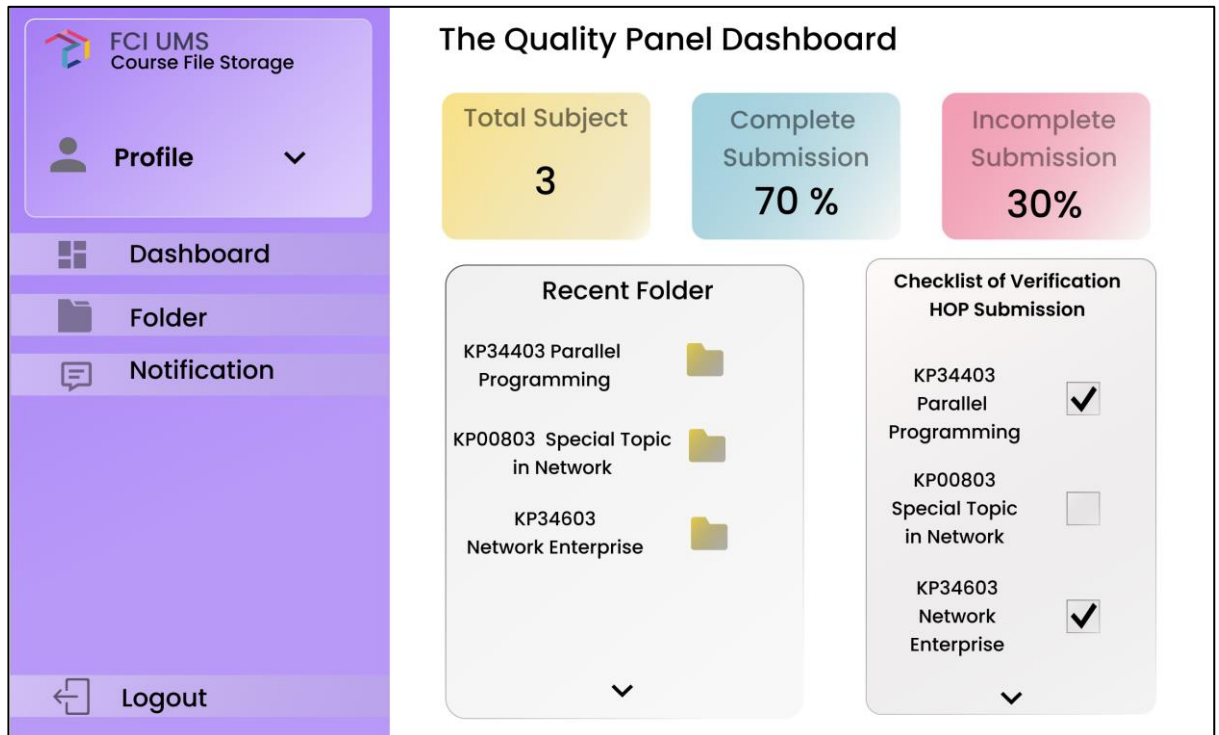


Figure 4.18: The Quality Panel Dashboard Page

The quality panel dashboard shows the total number of subjects that need to be examined and verified, the total number of complete and incomplete submission folder verifications, and a checklist of the complete verify folder to assist the quality panel in determining which submission folders have not been verified.

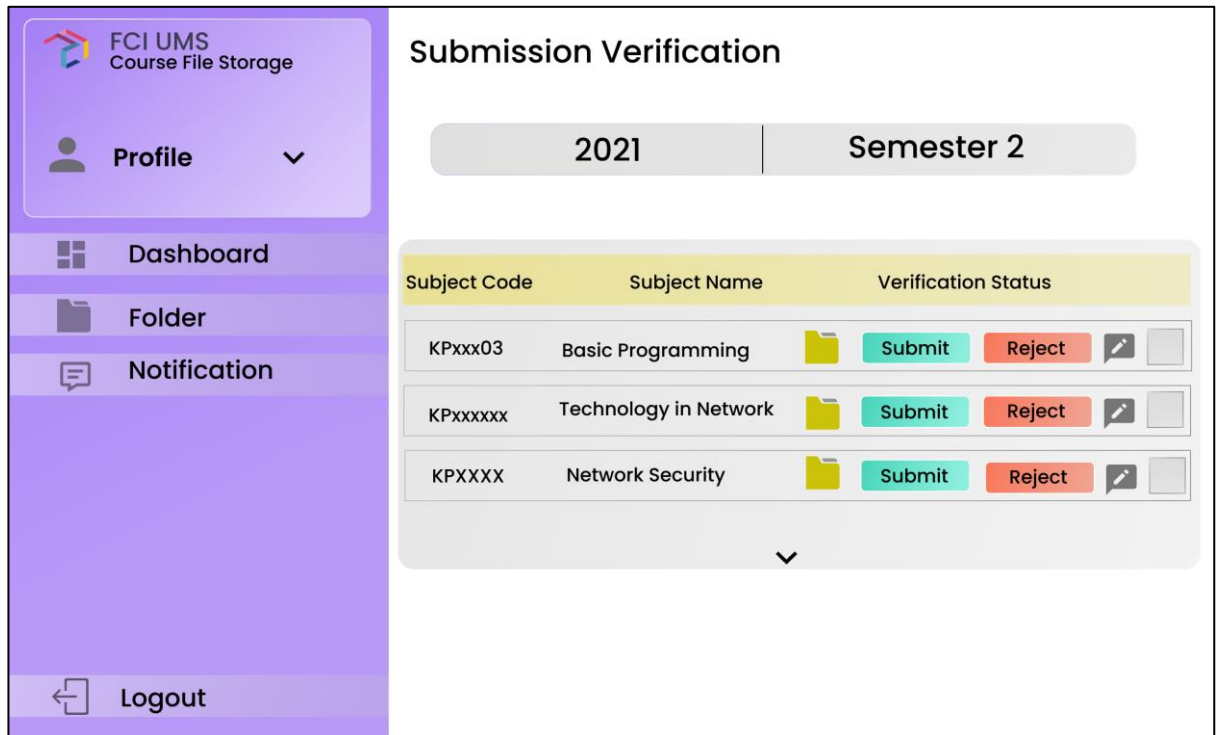


Figure 4.19: The Quality Panel’s Submission Verification Page

The submission verification page contains a list of subjects that the quality panel must verify. If the quality panel deems the submission folder suitable, the document may be delivered to the dean/deputy dean for review. However, if the submission folder has an error, the quality panel may reject it and ask the head of program to resubmit. This folder also has a comment option, which the quality panel may use to convey comments to the head of program or to the dean/deputy dean on course file submissions. Additionally, the website includes a checkbox; after the quality panel has finished reviewing the submission folder, the checkbox will be marked with a tick.

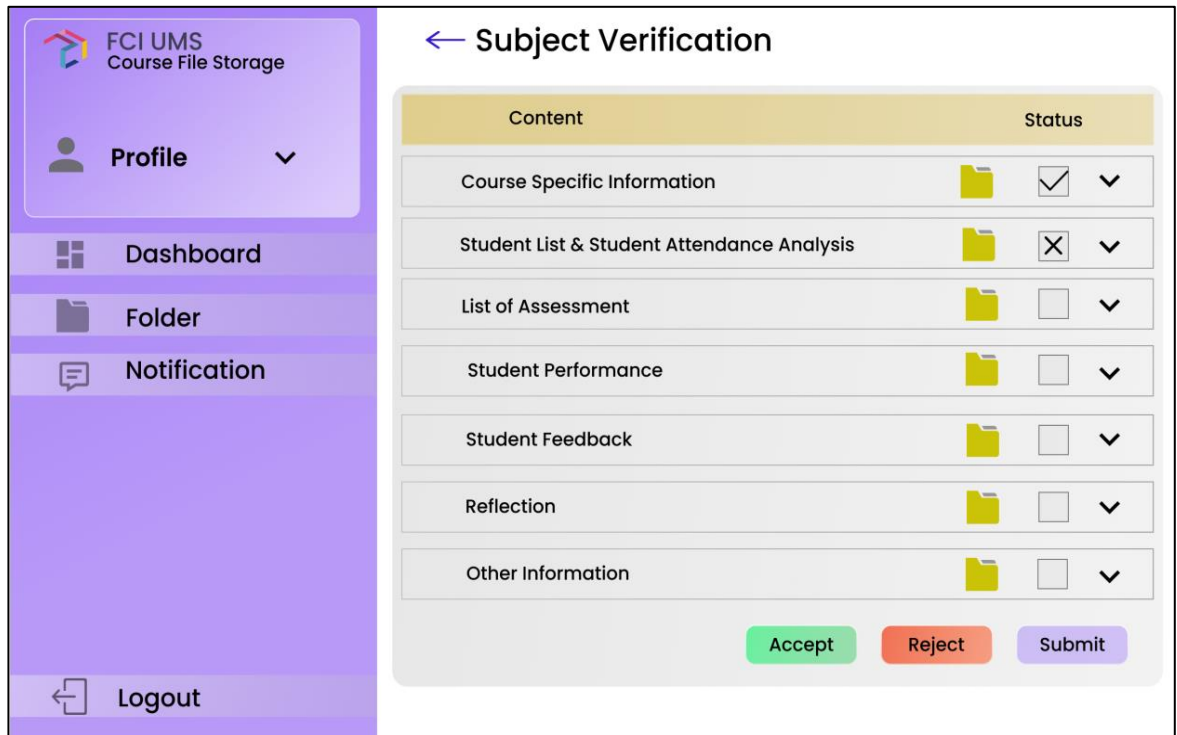


Figure 4.20: The Quality Panel's Subject Verification Page

The subject verification page of the quality panel contains the contents of each topic folder that the quality panel must review. The quality panel may approve the submission if all of the folders have been examined and indicate no issues; otherwise, the quality panel may reject it.

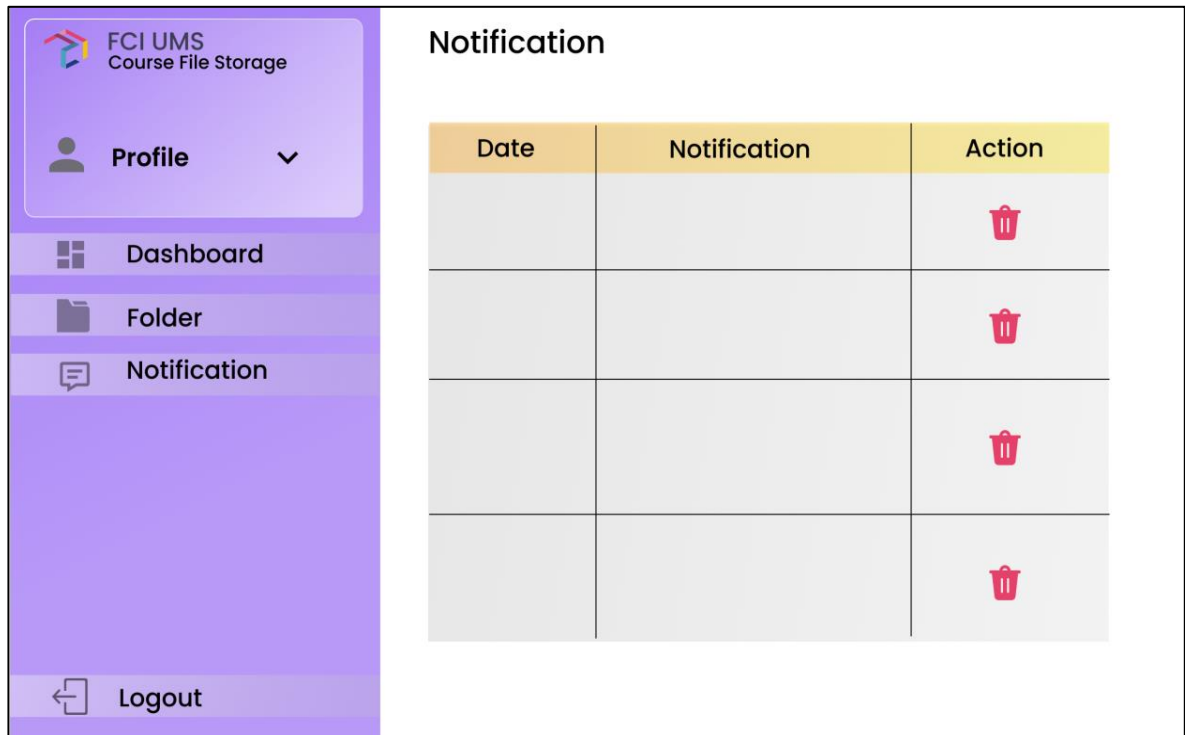


Figure 4.20: The Quality Panel’s Subject Verification Page

This page includes notification to the quality panel from the head of program, admin and other FCI leadership regarding whether the course file has any issue.

vi. Dean/deputy dean Page

The dean/deputy dean page contains a dashboard for the dean / deputy dean, a folder containing course-related documents submitted by the quality panel that the dean/deputy dean must check and validate. This page also includes notifications.

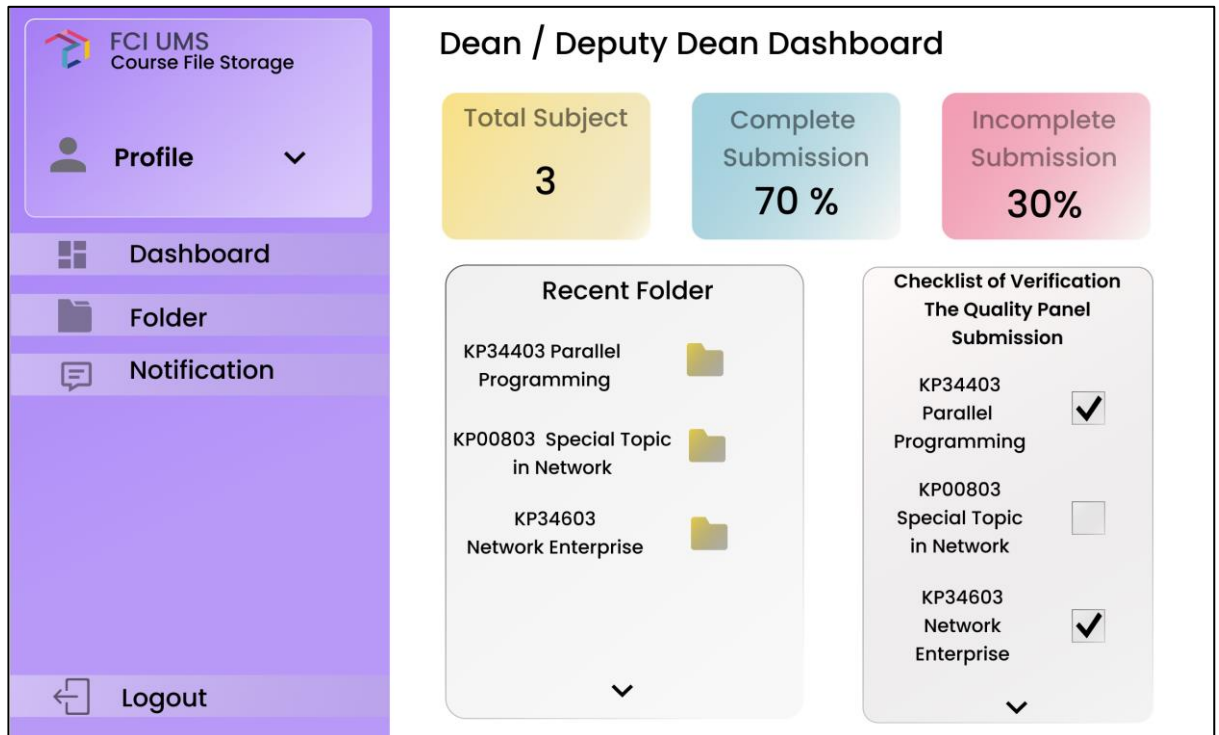







Figure 4.21: The Dean/deputy dean Dashboard.


The dean/deputy dean dashboard displays the overall number of subjects to be inspected and verified, the total number of full and incomplete submission folder verifications, and a comprehensive verify folder checklist to help the dean / deputy determine which submission folders have not been verified.


 FCI UMS
Course File Storage

 Profile 

 Dashboard
















 Folder

 Notification

 Logout

Submission Verification

2021Semester 2

Subject Code	Subject Name	Verification Status
KPxxx03	Basic Programming	    
KPxxxxxx	Technology in Network	    
KPXXXX	Network Security	    




Figure 4.22: The Dean/deputy dean submission verification page.

The submission verification page includes a list of subjects that must be verified by the dean / deputy dean. The document may be saved as a final course file if the dean / deputy dean thinks the submission folder is sufficient. If the submission folder has a mistake, the dean or deputy dean may reject it and request that the quality panel resubmit it. This folder also contains a remark option, which the dean/deputy dean may use to provide feedback on course file submissions to the quality panel or admin. Additionally, the website provides a checkbox that will be marked with a tick after the dean/deputy dean has completed examining the submission folder.

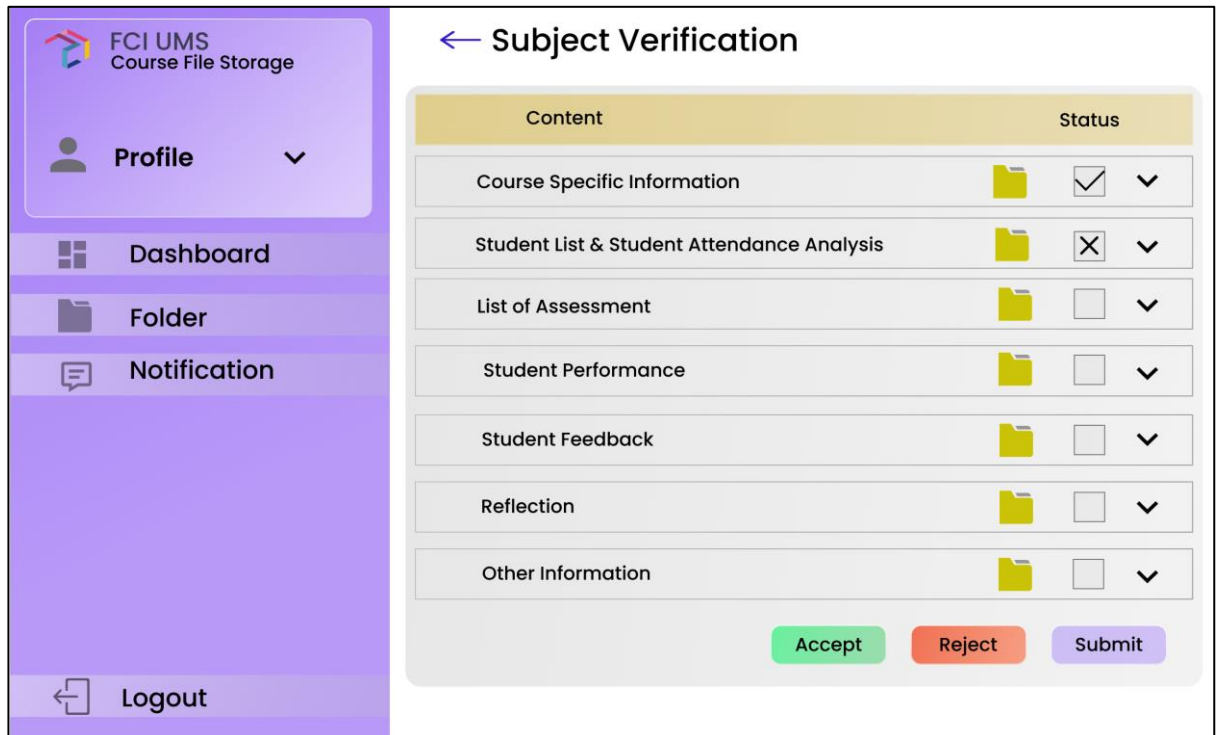


Figure 4.23: The Dean/deputy Dean Subject Verification Page

The dean's / deputy dean's subject verification page comprises the contents of each topic folder that the dean/deputy dean must evaluate. If all of the files have been checked and no concerns have been identified, the dean/deputy dean may accept the submission; otherwise, the dean/deputy dean may reject it.

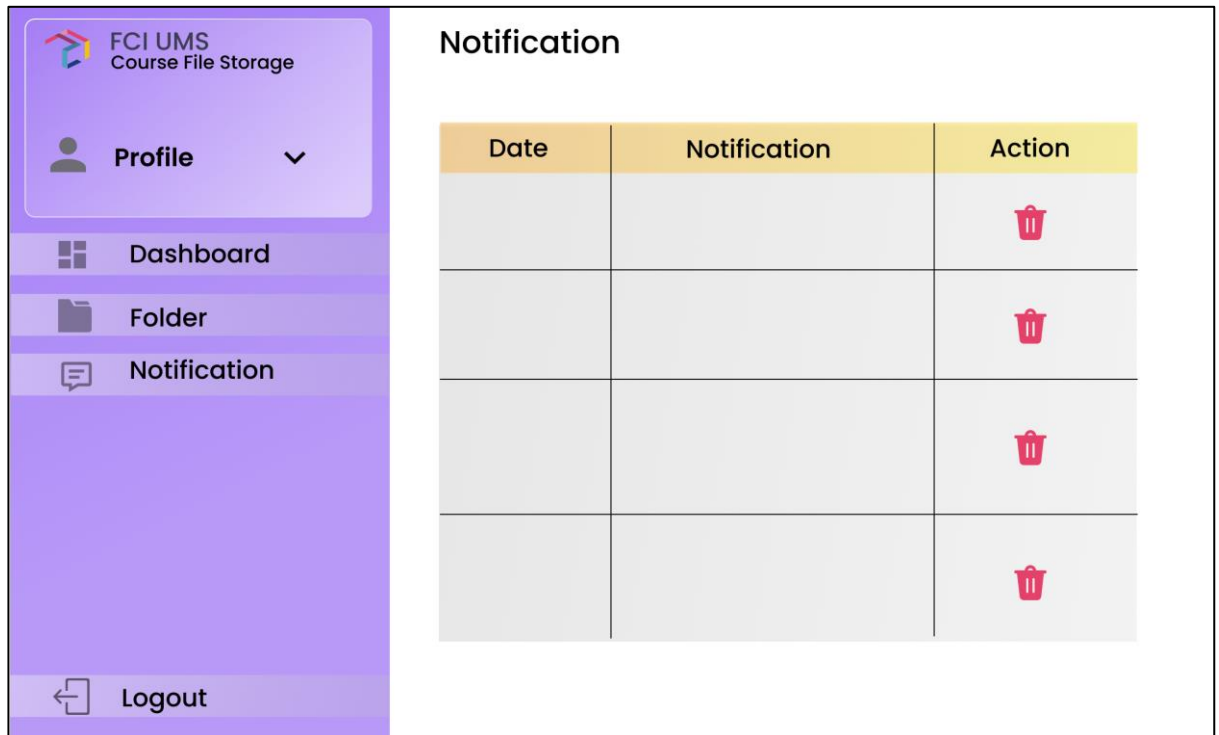


Figure 4.24: The Dean/deputy dean Notification Page.

This page contains notifications to the dean/deputy dean from the quality panel, administration, and other FCI leadership on the status of the course file.

4.4 Conclusion

In conclusion, the web-based system for course file storage, as well as the prototype, will be comprehensively discussed. Data Flow Diagram (DFD), Entity Relationship Diagram (ERD), and Data Dictionary will be used to demonstrate the system design. The user interface (UI) section will demonstrate and explain the design of the system's UI. The Data Flow Diagram (DFD) depicts the data process. A data flow diagram is a technique of depicting information flows inside a system that is based on systematic analysis and design. It displays logic models and depicts data transformation in a system, provides a framework for modelling information flow, and provides deconstruction to show specific data flows and activities. In this project, the Data Flow Diagram will be represented in three levels: context diagram, DFD level 1, and DFD level 2. The Entity Relationship Diagram (ERD) between the attributes on the system. The system contains admin, subject, user, batch,

submission folder, and position. Data dictionary contains information about other database items, such as data ownership, data links to other objects, and other data. The data dictionary provided in this chapter will give detailed information about the data utilised in this project. The data dictionary is given as field name, data type, field size, description, and example. This system will feature a user interface to make it user-friendly. Norman's design ideas build the user interface. Norman's design principles include visibility, feedback, constraints, mapping, consistency, and affordance. The design concept also uses Glassmorphism UI. Glass Morphism is a type of user interface that uses a frosted-glass appearance, producing a transparent and hazy background. UI design is split into user types: professor, program head, quality panel, dean/deputy dean.

CHAPTER 5

RESEARCH IMPLEMENTATION / EXPERIMENT

5.1 Introduction

This chapter will cover the research methods employed in this study. The research methodology will go through the research background, hypothesis, experiment technique, hypothesis testing, and findings analysis.

5.2 Research Background

The security of the website system is so important nowadays. There are several methods that are used to protect the web, one of them is by implementing an SSL Certificate. The implementation of SSL Certificate (Meralus, 2020) is used to secure user data, data transfer, and logs. Based on the fact that implementation of SSL Certificate is important to keep the website secure, in this project the research is

conducted to find out the suitable open source SSL Certificate to implement in this project.

The study will concentrate on the validity and quality of open source SSL certificates. The SSL Certificate will be tested to ensure that regulatory requirements and compliance, including some defined in the PCI DSS, GDPR, or NIST, are being followed correctly. Furthermore, SSL/TLS security testing ensures that users are properly protected against MITM attacks and other data interception methods (in the case of HTTPS encryption).

5.3 Hypothesis

The security level of open source SSL Certificates is different from each other and depends on the SSL provider.

5.4 Experiment

The experiment in this project uses an online lab called ImmuniWeb (ImmuniWeb, 2021) to determine the security level of various open source SSL certificates, including OpenSSL, EasyRSA, and CFSSL. The experiment approach is inspired by the experiment technique done by Pathak, A., Sharma, R. D., & Dey, D. (2018) from Bennett University, Greater Noida, India that utilized the same procedure to evaluate SSL Certificate.

This method is used because ImmuneWeb contains a lot of features that assess the SSL Certificate in detail. The way this online lab analyses the SSL Certificate according to PCI DSS requirements, HIPAA guidance or NIST guidelines. The result for the experiment will be graded according to the score obtained. The grading is based on scoring below.

Table 5.3 : Scoring of ImmuneWeb Tools (ImmuneWeb, 2021)

Grade	Score
A+	More than 99
A	90 – 99
A-	80 – 89
B+	70 – 79
B	60 – 69
B-	50 – 59
C+	35 – 49
C	20 – 34
F	Less than 20

The criteria of scoring in ImmuniWeb is based PCI DSS requirements, HIPAA guidance or NIST guidelines, the criteria and scoring point will be explained in Table 5.4:

Table 5.4 : Scoring Criteria of ImmuneWeb Tools (ImmuneWeb, 2021)

Criteria	Point
SSL Certificate is Extended Validation (EV) certificate	+10 points
HTTP website redirects to HTTPS	+10 points
The server prioritizes cipher suites that provide high Perfect Forward Secrecy (PFS)	+10 points
TLS Fallback SCSV extension is provided by the server.	+10 points
The server employs a long duration of HTTP Strict Transport Security (HSTS).	+10 points
TLSv1.3 is supported by the server.	+10 points
The certificate Server X509 is before version 3	-5 points
SSL Certificate has been used for more than three years period	-5 points
The SSL Certificate was not signed using the correct algorithm.	-5 points
The server is not capable of OCSP stapling.	-5 points
P-256 and P-384 curves are not supported by the server.	-5 points

The server does not support some encryption suites suggested by NIST or HIPAA.	-5 points
Support is provided for TLS cipher suites that have not been authorised by NIST or HIPAA.	-5 points
Elliptic Curves are supported by the server, but not the EC Point Format extension.	-5 points
There is no certificate chain offered.	-10 points
Vulnerable (HTTP) content is included in the website.	-10 points
Client-initiated secure renegotiation is accepted by the server.	-10 points
The server makes no reference to support for secure renegotiation.	-10 points
TLSv1.3 is not supported by the server.	-10 points
The certificate chain is dependent on an expired certificate.	-20 points
The certificate signature does not use SHA2.	-20 points
The certificate does not include information on revocation.	-20 points
Support for SSL but choose TLSv1.1 or TLSv1.2 or TLSv1.3	-20 points
SSL/TLS cipher suites that are not PCI DSS compliant are supported.	-40 points
The certificate key length or the DH parameter are insufficient (2048 bits or 256 bits for EC).	-40 points
At least one elliptic curve with a size less than 224 bits is supported by the server.	-40 points
SSL is supported, but not TLSv1.1, TLSv1.2, or TLSv1.3.	-40 points
The server utilises TLS compression, which makes it vulnerable to CRIME attacks.	-40 points
SSL/TLS cipher suites that are not PCI DSS-certified are preferable.	-50 points
The certificate is unreliable or invalid.	-60 points
CVE-2014-0224 vulnerability exists on the server (OpenSSL CCS flaw)	-60 points
CVE-2016-2107 vulnerability exists on the server (OpenSSL padding-oracle flaw)	-60 points
CVE-2021-3449 vulnerability may exist on the server. (OpenSSL renegotiation vulnerability created deliberately)	-60 points
POODLE over TLS is a vulnerability on the server.	-60 points
GOLDENDOODLE is a threat to the server.	-60 points

Zombie POODLE may infiltrate the server.	-60 points
Sleeping POODLE is a threat to the server.	-60 points
0-Length OpenSSL is a vulnerability on the server.	-60 points
Client-initiated insecure renegotiation is accepted by the server.	-60 points
ROBOT (Return of Bleichenbacher's Oracle Threat) is a threat to the server.	-60 points
Heartbleed is a vulnerability on the server.	-70 points

5.4 Test Hypothesis

The hypothesis will be tested by collecting the ImmuniWeb results. According to the ImmuniWeb results, all open source SSL Certificates: OpenSSL, EasyRSA, and CFSSL were achieved grade A.

Based on this finding, the hypothesis is rejected since all of the open source SSL Certificates have the same security level, implying that the open source SSL Certificate security level is the same and does not rely on the SSL Certificate provider.

5.5 Analyse Results

The result of the ImmuniWeb test shows that all the open source SSL Certificates show the same result, which indicates that all the security levels of SSL Certificates are the same and the only difference is the issues that are faced by each of the SSL Certificates. The full result of the ImmuniWeb will be shown in the Appendix.

The issue face by each of the SSL Certificate will be in the Table 5.5:

Table 5.5: SSL Certificate Issues (ImmuneWeb, 2021)

SSL Certificate	Issues
EasyRSA	<ul style="list-style-type: none">• Issue with supported cipher TLSV1.0, support TLS_RSA_WITH_3DES_EDE_CBC_SHA which does not meet the standard of PCI DSS, NIST and HIPAA requirements.• Supported TLSv1.0 protocol which does not meet the standard of PCI DSS NIST and HIPAA requirements.
OpenSSL	<ul style="list-style-type: none">• Supported TLSv1.0 protocol which does not meet the standard of PCI DSS NIST and HIPAA requirements.• For TLS versions 1.2, the server does not support the Extended Master Secret extension.• The server does not support TLSv1.3, which is the only TLS version with no known vulnerabilities or exploitable vulnerabilities.• The server allows client-initiated secure renegotiation, which may be dangerous due to a misconfiguration or vulnerability, and may allow Denial of Service attacks.
CFSSL	<ul style="list-style-type: none">• Issue with supported cipher TLSV1.0, support TLS_RSA_WITH_3DES_EDE_CBC_SHA which does not meet the standard of PCI DSS, NIST and HIPAA requirements.• Supported TLSv1.0 protocol which does not meet the standard of PCI DSS NIST and HIPAA requirements.

According to the aforementioned results, there are various differences between the open source SSL Certificate problems. As a result, the SSL Certificate

will be selected based on the amount of issues that it encounters. The SSL Certificate with the fewest problems will be chosen. Since EasyRSA and CFSSL both have the same problems, CFSSL will be chosen because it provides additional features for securing the online system.

5.6 Conclusion

In conclusion, the project background is about the research to secure a website via the use of an SSL Certificate. There are many open sources SSL Certificate providers, each of which offers a unique set of features and security levels. The project's hypothesis is that the security quality of open source SSL certificates varies and is dependent on the SSL provider. ImmuniWeb is used to conduct the experiment in this research. ImmuniWeb will assign a grade based on the degree of security, and the highest-grade SSL Certificate will be chosen for use in this project. However, the outcome for all SSL Certificates is the same, A, which rejects the hypothesis and indicates that all security levels are the same and independent of the SSL Certificate issuer. Because the security level is the same, the SSL Certificate will be chosen based on the problems that the SSL Certificate encounters. CFSSL was selected as the SSL Certificate for this project since it demonstrates more features with less issues.

CHAPTER 6

CONCLUSION

6.1 Introduction

This chapter summarizes the whole chapter of this project. Each chapter is summarized, beginning with Chapter 1, then Chapter 2, Chapter 3, Chapter 4, and Chapter 5. Additionally, in this chapter, we will explore the possibility of further work.

6.2 Conclusion

To ensure the uniformity of the evaluation, every Institute of Higher Learning must go through the Malaysian Qualification Agency's (MQA) certification procedure. The conventional way of storing course files is inefficient and may cause problems. This includes a lack of storage space, security concerns, the possibility of damage, and a problem with document transit. The idea for understanding the web-based approach for course file storage comes from how other institutions keep their course materials. Taking inspiration from other organizations, the web-based system for course file systems will be created.

The system is being developed for the Faculty of Computing and Informatics (FCI) UMS to test the system's ability to store course files. If a problem arises, debugging and improving the system is considerably simpler than beginning from the university scale.

The goal of the project is to develop a web-based system for course file storage and implement hybrid encryption. The objectives of this project is to investigate and implement hybrid encryption to improve confidentiality of web system for course file storage, to design and develop a web system for course file storage which can store course file securely, and to evaluate the proposed web system in term of its function's performance by testing the user acceptance from user feedback.

The project's scope is to create a web system for course file storage for FCI. The primary user of this web system is lecturer, head of the program, the quality panel, dean or deputy dean, and assistant registrar as the admin. Every semester, every lecturer must prepare a course file for each subject they are taught at the end of the semester. The course file will be reviewed by the head who will submit it if there is a problem with it. Then, the document will be forwarded to the quality board for further review, and lastly to the dean for approval.

The requirement is derived from the FCI Course File Checklist. The checklist includes particular folders to upload certain files. Discuss the example of an existing system used to store course files in the current system overview. Since there are fewer system examples for the course file system, another appropriate file storage system is chosen as an example for this project. Examples of existing systems include UNISEL e-Course File System, OpenKM, and Papermerge. The hybrid encryption is addressing the hybrid encryption that may be used to secure system access. The hybrid encryption discussed in this chapter is HAN encryption (Safi, 2017), Agrawal and Patankar developed encryption architecture and SSL Certificate (X.509 Certificate). Based on the analysis conducted, selecting SSL Certificate is the best option owing to the timing of this project. Examples include OpenSSL, EasyRSA, and CFSSL.

The Agile Development Process is the project methodology. This technique includes six stages, including necessity, planning, manufacturing, testing,

implementation, and assessment, and clarifying each step. Having a project methodology in place enables a smooth transition between phases, which offers insight to the length of each step.

The system analysis used in this project is Data Flow Diagram (DFD), Entity Relationship Diagram (ERD), and Data Dictionary to illustrate system architecture. The UI section will show and explain the system's UI design. The Data Flow Diagram (DFD) illustrates the process. A data flow diagram is a method of systematically analysing and designing information flows inside a system. It shows logic models, illustrates data transformation in a system, offers a framework for information flow modelling, and provides deconstruction to reveal particular data flows and activities. The data flow diagram will be shown in three levels: context diagram, DFD level 1 and DFD level 2. Entity Relationship Diagram (ERD) between system attributes. The system has admin, topic, user, batch, folder and position. Data dictionary provides information about other database items, including data ownership, data connections to other objects, and other data. The data dictionary given in this chapter will describe the data used in this project. Data dictionary is provided as a field name, kind of data, field size, description, example. This system has a user-friendly interface. Norman's design concepts create a user interface. Norman's design principles include visibility, feedback, limitations, mapping, consistency, and affordability. The design idea utilises Glass Morphism UI. Glass Morphism is a kind of user interface using a frosted-glass look, creating a translucent, hazy backdrop. UI design is divided into user types: professor, programme leader, quality panel, dean/deputy.

The research methodology is about research to secure a website using an SSL Certificate. There are numerous open SSL Certificate providers, each offering a distinct set of features and security levels. The premise of the project is that the security level of open source SSL certificates varies, depending on the supplier. ImmuniWeb is used to perform the experiment. ImmuniWeb will award a security-based grade and choose the highest-quality SSL Certificate for usage in this project. However, the result for all SSL Certificates is the same, A, which rejects the hypothesis and shows that all security levels are the same and independent of the issuer of SSL Certificates. Because the security level is the same, the SSL Certificate

is selected depending on the issues the SSL Certificate experiences. CFSSL has been chosen as the SSL Certificate for this project, showing greater features with less problems.

6.3 Future Work

In the future, this project can be improved into a large scale that can be used to store course files that can be used by all faculty in UMS. The web-based system also can be improving in term of design and add more features to make the web system more functional and user-friendly.

The project also should be improving in the implementation of hybrid encryption to implement certified authority (CA) SSL Certificate, to improve the security of the web system.

The web system also should be implemented with algorithms that can increase confidentiality, availability, data integrity, and non – repudiation. For example, encrypting passwords using a steganography algorithm.

REFERENCES

- Agrawal, A., & Patankar, G. (2016). Design of Hybrid Cryptography Algorithm for Secure Communication. *International Research Journal of Engineering and Technology (IRJET)*, 3(1), 1323 - 1326. <https://doi.org/e-ISSN: 2395 -0056>
- Alias, S. (2021). Interview of Course File Details [In person]. UMS.
- Avila, P. (2021). *openkm/document-management-system*. GitHub. Retrieved 5 April 2021, from <https://github.com/openkm/document-management-system>.
- Breitmeyer, R. (2015, July 21). *What are the 7 disadvantages to a manual system?* LinkedIn. Retrieved 5 February 2021, from <https://www.linkedin.com/pulse/what-7-disadvantages-manual-systemrichard-breitmeyer>.
- Chen, C., Diao, W., Zeng, Y., Guo, S., & Hu, C. (2018, September). DRLgencert: Deep learning-based automated testing of certificate verification in SSL/TLS implementations. In 2018 IEEE International Conference on Software Maintenance and Evolution (ICSME) (pp. 48-58). IEEE.
- CyberSecurity Malaysia. (2019). CYBER SECURITY GUIDELINE FOR 2 SECURE SOFTWARE 3 DEVELOPMENT LIFE CYCLE (SSDLC) (pp. 1 - 69). Cyberjaya: CYBERSECURITY MALAYSIA.
- Document Management System Software / OpenKM*. OpenKM. (2021). Retrieved 5 April 2021, from <https://www.openkm.com/#DocumentManagement>.
- Documentation / Papermerge documentation*. Papermerge.com. (2021). Retrieved 7 April 2021, from <https://papermerge.com/docs/>.
- Enlightn: Boost your Laravel App's Performance & Security*. Laravel-enlightn.com. (2021). Retrieved 13 April 2021, from <https://www.laravel-enlightn.com/>.
- FCI. (2020). *GUIDELINE FOR FKI COURSE FILE*

- Harkous, H., & Aberer, K. (2017, March). " If You Can't Beat them, Join them" A Usability Approach to Interdependent Privacy in Cloud Apps. In *Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy* (pp. 127-138).
- Lynch, W. (2019). Data Flow Diagram Comprehensive Guide with Examples. Medium. Retrieved 28 May 2021, from <https://warren2lynch.medium.com/data-flow-diagram-comprehensive-guide-with-examples-d9858387f25e>.
- Malewicz, M. (2020). Glass Morphism in user interfaces. UXcollective. Retrieved 12 June 2021, from <https://uxdesign.cc/glassmorphism-in-user-interfaces-1f39bb1308c9>.
- Melo, S. (2019). 8 Disadvantages of a paper document management system. Mydatascope.com. Retrieved 5 February 2021, from <https://mydatascope.com/blog/en/8-disadvantages-of-paper-document-management-system/>.
- Meralus, T. (2020). How to Secure Your Website with OpenSSL and SSL Certificates | Linux Journal. Linuxjournal.com. Retrieved 13 June 2021, from <https://www.linuxjournal.com/content/how-secure-your-website-openssl-and-ssl-certificates>.
- Moung, E. (2021). Interview of Course File Details [In person]. UMS.
- Muslim, N., Rahman, M. D. A., Subramaniam, H., Rosman, S. M., & Sukunam, M. (2019). Development of e-Course File System in the Department of Engineering, Faculty of Engineering and Life Sciences, Universiti Selangor. *Journal Online Jaringan Pengajian Seni Bina (JOJAPS)*, 14, 76–81. <https://doi.org/eISSN 2504-8457>.
- Nguyen-Duc, A., Do, M. V., Hong, Q. L., & Khac, K. N. (2021). On the combination of static analysis for software security assessment--a case study of an open-source e-government project. *arXiv preprint arXiv:2103.08010*.
- Nikolov, I. (2018). *Council Post: Why an SSL Certificate Is Important for Your Company Website*. Forbes. Retrieved 12 April 2021, from

<https://www.forbes.com/sites/forbestechcouncil/2018/05/18/why-an-ssl-certificate-is-important-for-your-company-website/?sh=646721ae1dc3>.

OpenSSL. Openssl.org. (2018). Retrieved 12 April 2021, from <https://www.openssl.org/>.

Pathak, A., Sharma, R. D., & Dey, D. (2018). How vulnerable are the Indian banks: A cryptographers' view. arXiv preprint arXiv:1804.03910.

Rekhi, S. (2017). Don Norman's Principles of Interaction Design. Medium. Retrieved 12 June 2021, from <https://medium.com/@sachinrekhi/don-normans-principles-of-interaction-design-51025a2c0f33>.

Shah, I. A., Amjed, S., & Alkathiri, N. A. (2019). The economics of paper consumption in offices. *Journal of Business Economics and Management*, 20(1), 43-62. <https://doi.org/10.3846/jbem.2019.6809>

Sharifian, S., & Safavi-Naini, R. (2021). Information-theoretic Key Encapsulation and its Applications. arXiv preprint arXiv:2102.02243.

SSL Certificate - SSL Information Center / GlobalSign. GlobalSign GMO Internet, Inc. (2021). Retrieved 11 April 2021, from <https://www.globalsign.com/en/ssl-information-center/what-is-an-ssl-certificate>.

The World Counts. Theworldcounts.com. (2021). Retrieved 5 April 2021, from <https://www.theworldcounts.com/challenges/consumption/other-products/environmental-impact-of-paper/story>.

Tutorialspoint.com. 2021. System Analysis and Design - Overview - Tutorialspoint. [online] Available at: <https://www.tutorialspoint.com/system_analysis_and_design/system_analysis_and_design_overview.htm> [Accessed 29 May 2021].

What is Hybrid Encryption? - Definition from Techopedia. Techopedia.com. (2021). Retrieved 10 April 2021, from <https://www.techopedia.com/definition/1779/hybrid-encryption#:~:text=A%20hybrid%20>

encryption%20scheme%20is,keys%20along%20 with%20symmetrical%20 encryption.

What Is SSL (Secure Sockets Layer)? / What is an SSL Certificate? / DigiCert.
Digicert.com. (2021). Retrieved 11 April 2021, from <https://www.digicert.com/what-is-an-ssl-certificate>

APPENDIX

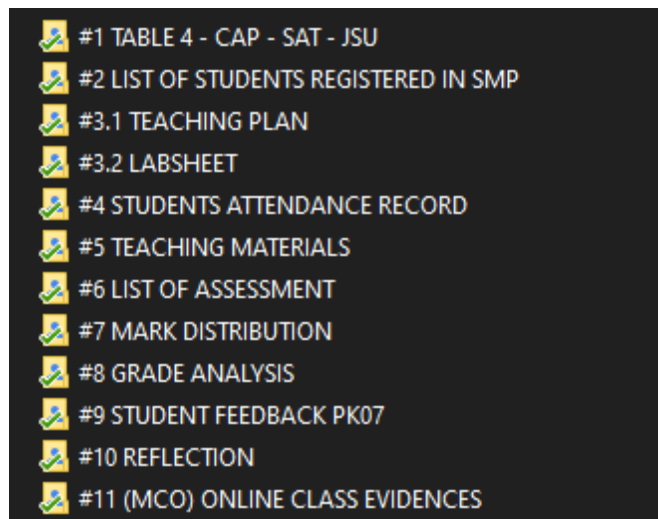
Appendix I: Interview with Dr Ervin Gubin Muong

Interview Script

1) What is course file?

In the academic setup, a course file is essentially a document that includes all the necessary details regarding the batch, assessment, and overall outcomes of the course. The university mandate the need to keep a course file by the faculties, and most are quite strict on following it too.

2) What is the document required to prepare course file?



3) Explain the process of preparing course file?

After a semester ends, all lecturers must upload the course files.

4) How the course file is store? Is there is any different of storing course file method before COVID 19 pandemic and during COVID 19 pandemic?

Pre-covid19=hardcopy. Covid19-era=softcopy

5) How long each batch of course file should be store?

Every semester must have its own set of course files.

- 6) What are the pros and cons of storing course file using conventional method?
Explain what is the problem that face by FCI lecturers when storing course file using conventional method?

Hardcopy = (pro) More secured as it was stored in the faculty's quality room.

(Con) If any of the files are needed during the online audit, it will be a hassle.

This is especially true for old course files, as lecturers are prone to deleting old files from their PC/Laptop, forgetting where they store the softcopy, or the PC/Laptop that stored those files has broken.

- 7) What is your opinion about digitalization of course file by creating web – based storage over storing course file using conventional method?

Please proceed. It's good.

- 8) In the web – based system for course file storage who is the user and what is their role?

All the executive staff in each faculty. To do what is necessary during audit.

- 9) Who should become admin in web – based system for course file storage ?

All the executive staff in each faculty.

- 10) Who should check the course file quality and approve it?

All the executive staff in each faculty.

- 11) How the report format that should be generated by the report generator of web – based system for course file storage?

Please consult with head of program as they know better. At least, the report should be able to identify missing file/insufficient file. The main objective is to make sure that the course files are complete for every semester.

- 12) What is the features or function that you are hoping in the web – based system for course file storage?

Upload/download and all that necessary user information.

Appendix II: Interview with Dr Suraya Alias

1) What is course file?

Course file is the file that contain information about subject that teach by a lecturer for every end of the semester for the purpose of audit.

2) What is the document required to prepare course file?

I will send you the FCI Course File Checklist it contain all the information need to prepare course file.

3) Explain the process of preparing course file?

Course file will be prepared by the lecturers according to the subject that they teach , then they submit to head of program for verification, the head of program will submit to the quality panel for checking, and lastly will be verify and finally store faculty's quality room.

4) How the course file is store? Is there is any different of storing course file method before COVID 19 pandemic and during COVID 19 pandemic?

Before COVID 19 offline, during pandemic online using Google Drive.

5) How long each batch of course file should be store?

Around 5 – 6 years.

6) What is the pros and cons of storing course file using conventional method? Explain what is the problem that face by FCI lecturers when storing course file using conventional method?

It takes time find the document when needed, no version control system so it will confuse lecturer whether the course file is old version or latest one.

7) What is your opinion about digitalization of course file by creating web – based storage over storing course file using conventional method?

I think it is good.

8) In the web – based system for course file storage who is the user and what is their role?

Lecturer to upload the document related. The head of program, the quality panel, and dean/deputy dean will check and verify the course file.

9) Who should become admin in web – based system for course file storage ?

The assistant registrar because they have the list of the lecturer information.

10) Who should check the course file quality and approve it?

Head of program, quality panel, dean/deputy dean.

- 11) How the report format that should be generated by the report generator of web – based system for course file storage?

Based on the FCI Course File Checklist

- 12) What is the features or function that you are hoping in the web – based system for course file storage?

No special features as long as it work like the offline course file preparation, and store the course file in systematic way.

