



MUICT Connect

BY

MISS RAMITA DEEPROM 6488018

MR. THITIWUT HARNPHATCHARAPANUKORN 6488025

MR. BURIT SIHABUT 6488079

MR. PONGSAKORN KONGKAEWKRASAMEE 6488215

SECTION 1

INSTRUCTORS

AJ. SNIT SANGLAO

**A Class Project Submitted in Partial Fulfillment of
the Requirements for
ITCS424 Wireless and Mobile Computing**

**Faculty of Information and Communication Technology
Mahidol University**

3rd Year Semester 2/2023

COPYRIGHT OF MAHIDOL UNIVERSITY

ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to the individuals listed below, whose invaluable contributions have been instrumental in the successful completion of this project. Additionally, we extend deep appreciation for the unwavering support we received from both our families and team members throughout this endeavor. Our efforts were dedicated to tirelessly finalizing every aspect of the project. While we acknowledge an oversight in identifying crucial information, we sincerely apologize for any inconvenience caused. Nevertheless, we believe our initiative will be helpful to readers.

Miss Ramita Deeprom 6488018

Mr. Thitiwut Harnphatcharapanukorn 6488025

Mr. Burit Sihabut 6488079

Mr. Pongsakorn Kongkaewrasamee 6488215

MUICT CONNECT

Miss Ramita Deeprom	6488018 ITCS/B
Mr. Thitiwut Harnphatcharapanukorn	6488025 ITCS/B
Mr. Burit Sihabut	6488079 ITCS/B
Mr. Pongsakorn Kongkaewrasamee	6488215 ITCS/B

B.Sc. (INFORMATION AND COMMUNICATION TECHNOLOGY)

INSTRUCTORS: AJ. SNIT SANGLAO

ABSTRACT

This project is a part of the Wireless and Mobile Computing course. The objective of this project was to address the lack of communication inside the Faculty of ICT by developing a centralized platform called MUICT Connect. The purpose of this platform is to simplify the distribution of academic and extracurricular notifications, making it easier for ICT students to stay informed about key events, academic updates, and opportunities. The project aims to improve the university experience by providing a more structured and effective method of delivering information in addition to the various communication channels that exist within the faculty.

CONTENTS

	Page
1 Introduction	1
1.1 Motivation	1
1.2 Problem Statement	1
1.3 Objectives of the Project	2
1.4 Scope of the Project	2
1.5 Expected Benefits	2
1.6 Organization of the Document	3
2 Background	4
2.1 Literature Review	4
3 Analysis and Design	6
3.1 System Architecture Overview	6
3.2 System Structure Chart	8
3.3 Process Analysis and Design	10
3.3.1 Data Flow Diagram	10
3.3.2 Data Dictionary	12
3.3.2.1 Process Description	12
3.3.2.2 Data Stores	16
3.3.2.3 Data Element	19
3.4 Database Analysis and Design	21
3.4.1 ER-Diagram	22
3.4.2 Relational Schema	23
3.4.3 File Structure	24
3.5 I/O Design	31
3.5.1 Interface Design	31
4 Implementation	38

4.1 Hardware and System Environment	38
4.2 Implementation Guide and Techniques	42
5 Testing and Evaluation	44
5.1 Unit Tests	44
5.1.1 Test Performed on No. 1: Register	44
5.1.2 Test Performed on No. 2: Login	45
5.1.3 Test Performed on No. 3: Post Management	46
5.1.4 Test Performed on No. 4: View Post	46
5.2 System Integration Test	47
5.2.1 Test Scenario	47
5.2.1.1 Register	47
5.2.1.2 Login	49
5.2.1.3 Post Management	50
5.2.1.4 View Post	53
6 Conclusions	55
6.1 Benefits	55
6.1.1 Benefits to Project Developers	55
6.1.2 Benefits to Users	55
6.2 Problems and Limitations	56
6.3 Future Work	56

LIST OF TABLES

	Page
Table 3.1: List of all Processes	12
Table 3.2: Process Description of Account Management	13
Table 3.3: Process Description of Post Management	14
Table 3.4: Process Description of Bookmark Management	15
Table 3.5: List of all Data Stores	16
Table 3.6: Data Store Description of User Database	17
Table 3.7: Data Store Description of Bookmark Database	18
Table 3.8: List of All Data Elements	19
Table 3.9: List of all Tables in Our System Database	23
Table 3.10: File Structure of Student	26
Table 3.11: File Structure of Staff	27
Table 3.12: File Structure of Event	28
Table 3.13: File Structure of Announcement	29
Table 3.14: File Structure of Bookmark	30
Table 5.1: Test Performed on No. 1: Register	45
Table 5.2: Test Performed on No. 2: Login	45
Table 5.3: Test Performed on No. 3: Post Management	45
Table 5.4: Test Performed on No. 4: View Post	46

LIST OF FIGURES

	Page
Figure 3.1: System Architecture Overview of MUICT Connect application	6
Figure 3.2: System Structure Chart of MUICT Connect application	8
Figure 3.3: Data Flow Diagram Level 0	10
Figure 3.4: Data Flow Diagram Level 1	11
Figure 3.4: ER-Diagram of MUICT Connect Database	22
Figure 3.5: Relational Schema of MUICT Connect Database	24
Figure 3.5.1.1: Interface Design of MUICT Connect application	31
Figure 3.5.1.2: Registration/Login Interface	32
Figure 3.5.1.3: Interface of Four Announcement Types	32
Figure 3.5.1.4: Bookmark Interface	33
Figure 3.5.1.5: Calendar/Notification Interface	33
Figure 3.5.1.6: Profile, Edit Profile Interface	34
Figure 3.5.1.7: About Us Interface	35
Figure 3.5.1.8: Logout Interface	35
Figure 3.5.1.9: Post management	36
Figure 3.5.2.1: Transition Diagram	36

CHAPTER 1

1 INTRODUCTION

This chapter provides an overview of the project's motivation, problem description, objectives, scope, expected benefits, and document organization. The following elements will provide a comprehensive explanation of our objectives, user issues, project goals, advantages, project scope, project plan, and a summary of the report.

1.1 Motivation

Effective communication plays a crucial role in enhancing the academic experience within the ever-changing field of higher education. MUICT Connect is a platform that aims to bridge the communication gap within the Faculty of ICT by providing fast and comprehensive information dissemination. Our goal in creating this groundbreaking platform is straightforward: to establish a single hub that facilitates effortless access to both academic and extracurricular information. MUICT Connect aims to empower students by providing them with comprehensive information on events, academic updates, and beneficial opportunities that can greatly enhance their university experience. This project demonstrates a dedication to improving the student experience, promoting a feeling of community, and eventually contributing to the overall development of the academic environment within the Faculty of ICT.

1.2 Problem Statement

1. The issue of excessive information is further compounded by repetitive notifications, which decreases the effectiveness of critical communications.
2. In the absence of a centralized platform, students are unable to access comprehensive academic and extracurricular data.
3. Students face challenges in monitoring academic events, deadlines, and extracurricular activities due to the decentralized nature of communication channels.

1.3 Objectives of the Project

- To establish and operate MUICT Connect as the primary platform utilized by the Faculty of ICT to distribute academic and extracurricular information.
- To ensure the efficient and seamless dissemination of information in order to prevent duplication and oversight in student announcements.
- To integrate features that facilitate event monitoring and notifications, thereby supporting students in their efforts to stay organized and engaged in academic opportunities.
- To foster a sense of engagement among university students by providing a forum that instructs and encourages active participation in extracurricular activities.

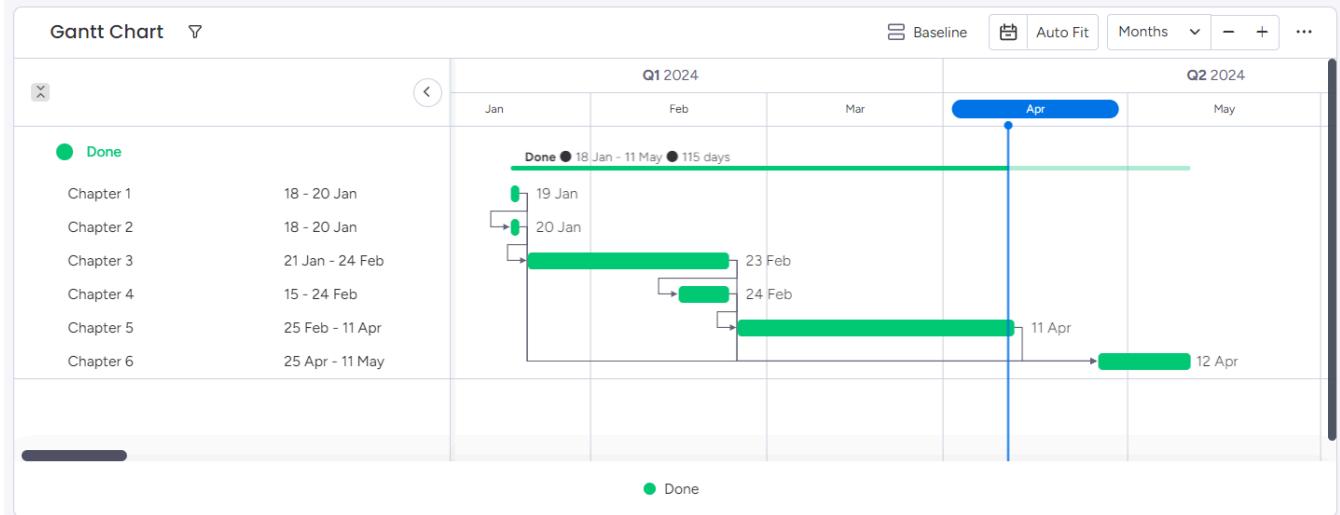
1.4 Scope of the Project

- Communication Centralization:
 - Establish a central platform for distributing academic and extracurricular announcements within for the Faculty of ICT.
 - Streamline existing communication channels for efficiency.
- Information Accessibility:
 - Enhance the delivery of academic updates, event details, and opportunities.
 - Provide easy access to essential information for ICT students.

1.5 Expected Benefits

MUICT Connect is introduced with the objective of improving communication inside the Faculty of ICT. Through the optimization of communication ways and the support of easy access to critical data, the platform guarantees enhanced efficiency and interconnection. MUICT Connect offers a user-friendly interface, smooth integration, and continuous improvements based on user feedback. This platform is designed to provide a future-ready and empowering environment for both students and faculty.

1.6 Organization of the Document



This is the Gantt chart to show our plan to do the project.

This document consists of 6 chapters including:

1. Introduction – The first chapter of this project encompasses the motivation, problem statement, project objectives, project scope, and expected benefits.
2. Background – The second chapter serves as a background, providing an overview of the project and incorporating a literature review summarizing our research.
3. Analysis and Design – The third chapter delves into the analysis and design of the project, emphasizing process and system design aspects. This includes a system architecture overview, system structure chart, process analysis and design, data flow diagram, process description, data stores, data elements, database analysis and design, ER-diagram, etc.
4. Implementation – The fourth chapter covers hardware and system environment details, presenting the implementation guide and techniques employed.
5. Testing and Evaluation – The fifth chapter concentrates on the testing and evaluation of the system, encompassing unit tests, system integration tests, and test scenarios.
6. Conclusion – The final chapter provides a conclusion that discusses the benefits for both project developers and users, addresses problems and limitations, and outlines potential avenues for future work.

CHAPTER 2

2 BACKGROUND

2.1 Literature Review

From the conducted research, one main news distribution approach for ICT students is by using social media platforms like Line, Facebook, and Instagram. On the positive side, utilizing social media provides quick and widespread dissemination of information, keeping students updated on events, announcements, and achievements. One notable downside is the issue of notifications. In a crowded online environment, important updates may get lost amid notifications, leading to students overlooking crucial information. Moreover, some students might ignore or disengage from these notifications, affecting the overall effectiveness of communication. Another challenge associated with posting student news on social media platforms is the potential for students to forget important dates. Information on social media feeds may be fleeting, making it easy for students to overlook deadlines, events, or other time-sensitive details.

Another existing tool is the MyCourses website. Students can access class materials, and news offers distinct advantages and challenges. On the positive side, MyCourses provides a centralized hub for academic resources, fostering organization and easy access to class materials. Students can view news related to their courses, minimizing the risk of missing crucial updates. However, there may be downsides, such as relying solely on MyCourses for communication, and limiting engagement, as students may not check the platform regularly. Technical glitches and delays might hinder students from receiving important updates promptly.

In comparing the two news distribution approaches for ICT students, both social media platforms and MyCourses have their strengths and weaknesses. Social media platforms like Line, Facebook, and Instagram offer quick and widespread dissemination, informing students about various aspects. However, challenges arise with notification issues, potential disengagement, and the risk of students forgetting important dates due to the fleeting nature of information on these platforms. On the other hand, MyCourses provides a centralized and organized space for academic

resources, reducing the likelihood of missing critical updates. Yet, over-reliance on MyCourses may limit student engagement, and technical issues with notifications can hinder timely communication. Striking a balance between these approaches, considering their respective advantages and addressing their drawbacks, is essential for effective news distribution in the dynamic realm of ICT education.

The introduction of MUICT Connect aims to enhance communication within the ICT Faculty. The platform provides quick access to vital data and optimizes communication channels to ensure increased interconnection and efficiency. MUICT Connect provides an easy-to-use interface, seamless integration, and ongoing enhancements based on user input. This platform is intended to give teachers and students a dynamic, future-ready environment.

CHAPTER 3

ANALYSIS AND DESIGN

This chapter is the analysis and design of our application. The reader will learn more about our system architecture, such as system structure chart, data flow diagram, and interface design.

3.1 System Architecture Overview

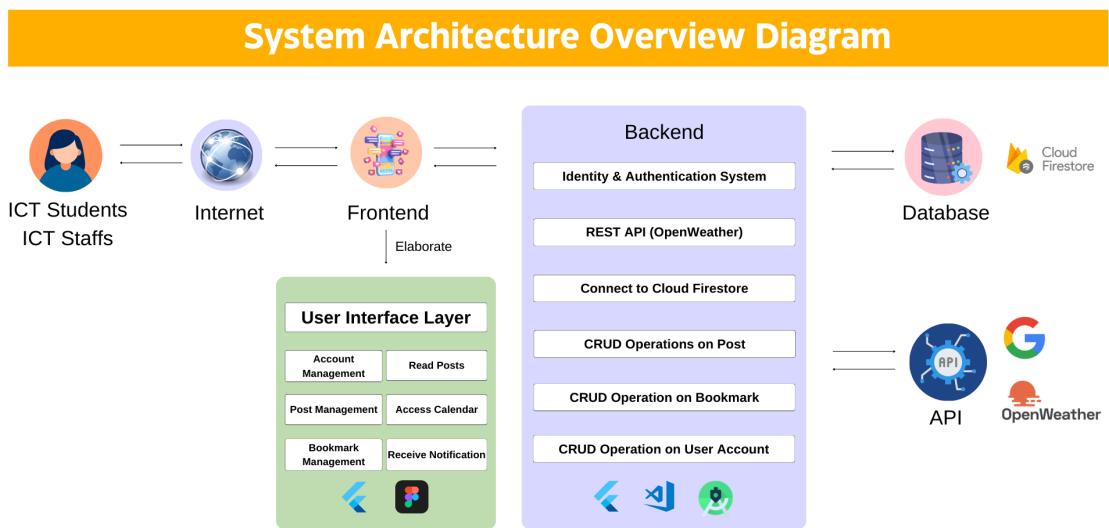


Figure 3.1: System Architecture Overview of MUICT Connect application

The MUICT Connect system is designed to facilitate communication and information exchange within the Faculty of ICT, providing a centralized platform accessible via the Internet. The system employs a client-server architecture with both frontend and backend components.

Frontend - User Interface Layer:

The user interface layer focuses on account management, post management, and bookmark management. Users, specifically ICT students, begin by registering for

accounts and inputting their email and password. The system allows users to edit their account information. Post management features enable staff to create posts, while students can view posts, access the calendar, and receive notifications for new posts. Bookmark management includes the ability for students to add posts to bookmarks and view their bookmarked page. The frontend tools include Flutter and Figma.

Backend:

The backend encompasses the identity and authentication system, a REST API for authentication (Google Sign-In), weather forecast, and connections to Cloud Firestore for database operations. CRUD (Create, Read, Update, Delete) operations are implemented for posts, bookmarks, and user accounts. The backend tools include Flutter, VSCode, and Android Studio.

Database:

The system utilizes Cloud Firestore, a cloud-based relational database, to store and manage data efficiently.

API Integration:

Weather API is integrated into the system through a REST API, allowing users to directly see current weather conditions and temperature updates within the application.

Google Sign-In is integrated into the system through RESTful to handle the actual authentication process.

In summary, the MUICT Connect system employs a robust architecture, combining Flutter, Google Sign-In, Weather API, and Cloud Firestore to deliver a seamless and feature-rich platform that enhances communication and information sharing within the Faculty of ICT. The choice of tools and technologies reflects a modern and efficient approach to application development.

3.2 System Structure Chart

MUICT Connect is a comprehensive system designed to enhance communication within the Faculty of ICT. This application has three main functions, which are Account Management, Post Management, and Bookmark management.

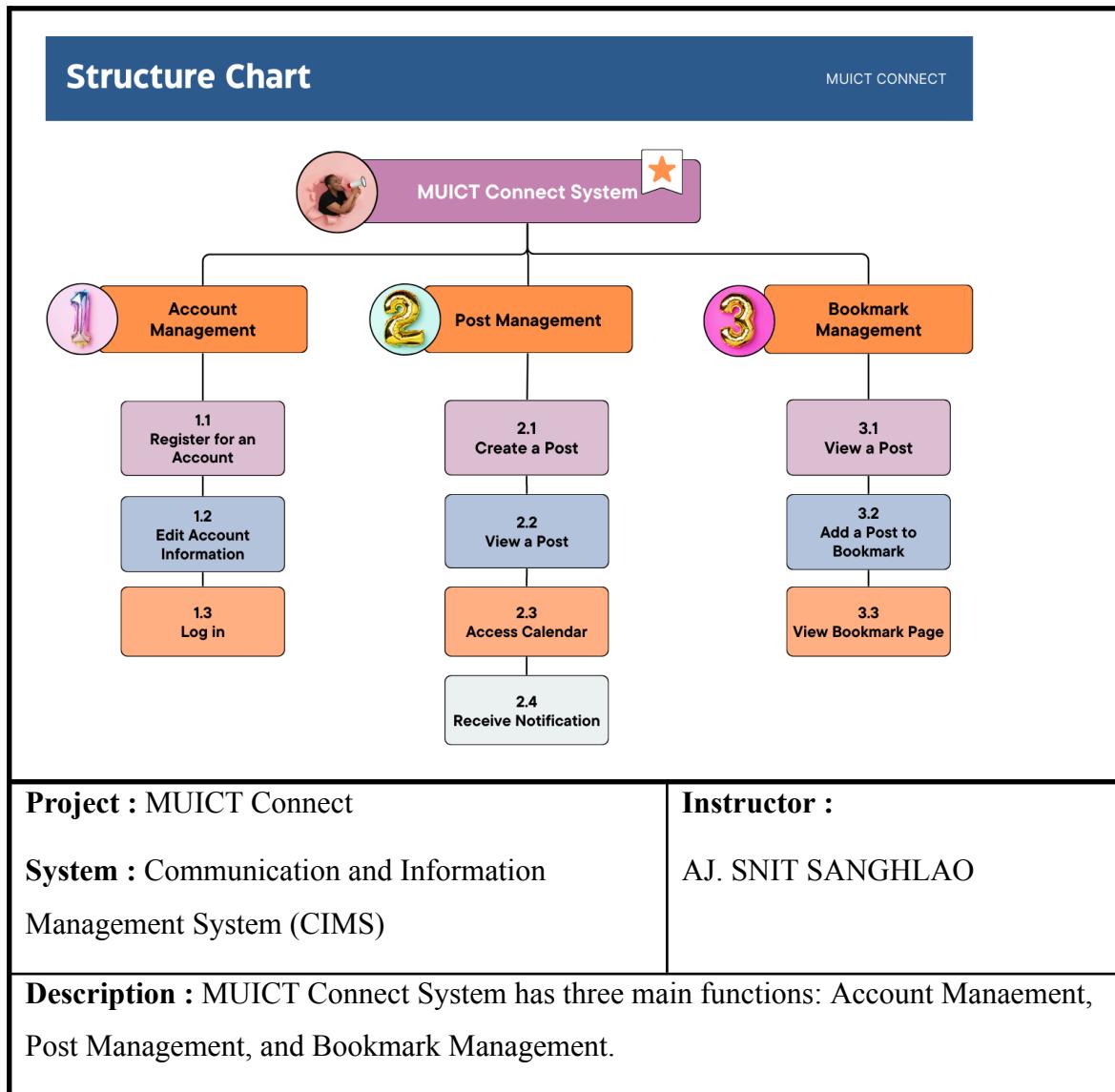


Figure 3.2: System Structure Chart of MUICT Connect application

The detailed description of each subsystem is shown below:

1. Account Management – First step for beginning to use Communication and Information Management System (CIMS)
 - 1.1. Register for an account – User must register to the system by input their email and password.
 - 1.2. Edit Account Information – System must allow user to edit their account.
2. Post Management – This is the post section.
 - 2.1. Create a Post – System must allow staff to create a post.
 - 2.2. View a Post – System must allow students to view a post.
 - 2.3. Access Calendar – System must allow users to access the calendar.
 - 2.4. Receive Notification – Students must receive a notification from new post.
3. Bookmark Management – This is the section for starting bookmark.
 - 3.1. View a post – Students must view a post.
 - 3.2. Add a Post to Bookmark – Students must add a post to Bookmark.
 - 3.3. View Bookmark Page – Students must view Bookmark Page.

3.3 Process Analysis and Design

3.3.1 Data Flow Diagram

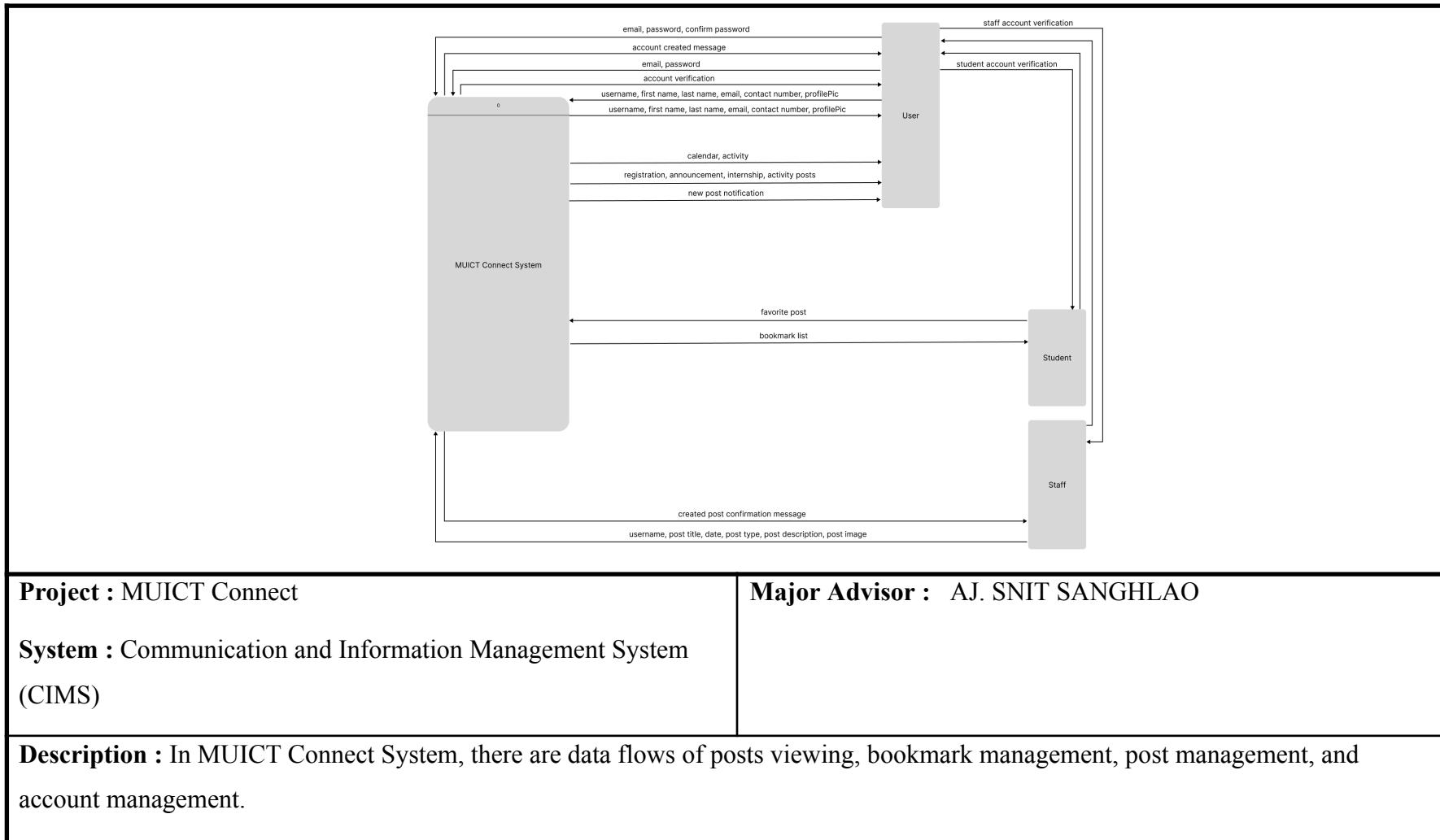


Figure 3.3: Data Flow Diagram Level 0

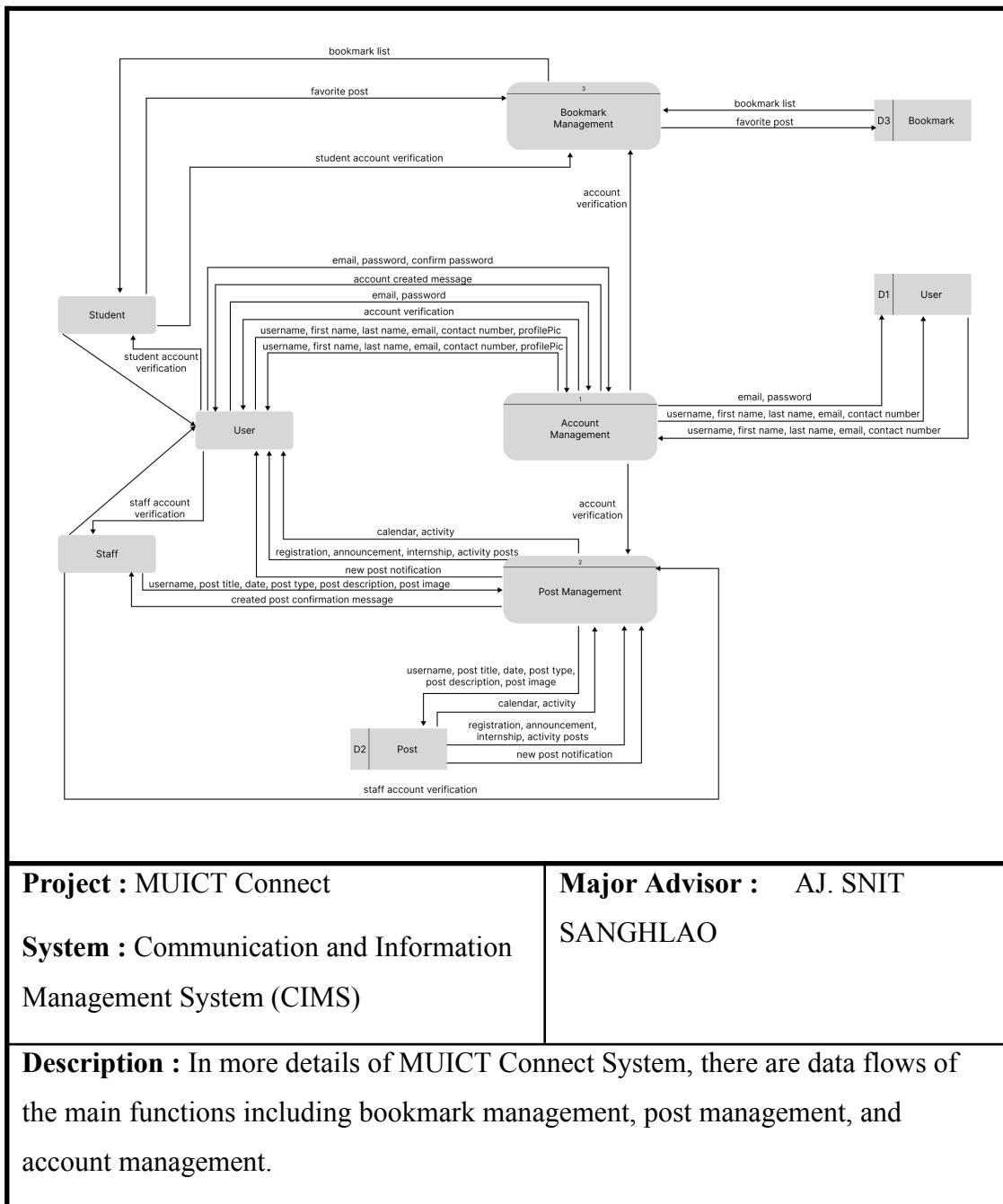


Figure 3.4: Data Flow Diagram Level 1

3.3.2 Data Dictionary

A data dictionary is a way to document and describe Processes, Data Stores, and Data Elements (Data Flow) that occur in a Data Flow Diagram (DFD). It is composed of 3 parts as shown below.

- Process Descriptions
- Data Stores
- Data Elements

3.3.2.1 Process Description

This section will provide a detailed description of each process that exists in this system. It includes Inbound Data, Outbound Data, and Logic Summary.

Table 3.1: List of all Processes

No.	Process	Name	Description
1	1	Account Management	Process for managing user accounts
2	1.1	Register for an Account	Process for sign up
3	1.2	Edit Account Information	Process for modify profile
4	1.3	Log in	Process for sign in
5	2	Post Management	Process for handle user posts
6	2.1	Create a Post	Process for create a post
7	2.2	View a Post	Process for view a post
8	2.3	Access Calendar	Process for visit calendar
9	2.4	Receive Notification	Process for get notification
10	3	Bookmark Management	Process for manage bookmarks
11	3.1	View a Post	Process for view a post
12	3.2	Add a Post to Bookmark	Process for save post to Bookmark
13	3.3	View Bookmark Page	Process for view saved posts to Bookmark

Table 3.2: Process Description of Account Management

Process Name	1 - Account Management
Description	Process for managing user accounts
Inbound data	<ul style="list-style-type: none"> • email • password • confirm password • username • first name • last name • contact number • profilePic • new password • account verification
Outbound Data	<ul style="list-style-type: none"> • account created message • new password • username • first name • last name • email • password • contact number • profilePic • password changed message • account verification
Logic Summary	<pre> graph TD User((User)) -- "email, password, confirm password" --> R1((1.1 Register for an account)) R1 -- "account created message" --> User R1 -- "username, first name, last name, email, contact number, profilePic" --> EAI((1.2 Edit Account Information)) EAI -- "new password" --> R1 EAI -- "password changed message" --> User EAI -- "username, first name, last name, email, contact number, profilePic" --> LogIn((1.3 Log in)) LogIn -- "account verification" --> EAI LogIn -- "account verification" --> User User -- "account verification" --> LogIn User -- "email, password" --> LogIn </pre> <p>The diagram illustrates the logic summary of the Account Management process. It features three main nodes: 'Register for an account' (1.1), 'Edit Account Information' (1.2), and 'Log in' (1.3). The process begins with a 'User' entity interacting with node 1.1. Node 1.1 sends an 'account created message' back to the 'User'. Node 1.1 also sends data ('username, first name, last name, email, contact number, profilePic') to node 1.2. Node 1.2 sends a 'password changed message' back to the 'User'. Node 1.2 also sends data ('new password', 'username, first name, last name, email, contact number, profilePic') to node 1.3. Node 1.3 sends 'account verification' messages to both the 'User' and node 1.2. Finally, the 'User' interacts with node 1.3, sending 'account verification' and 'email, password' data to it.</p>

Table 3.3: Process Description of Post Management

Process Name	2 - Post Management
Description	Process for handle user posts
Inbound data	<ul style="list-style-type: none"> ● username ● post title ● date ● post type ● post description ● post image ● calendar ● activity
Outbound Data	<ul style="list-style-type: none"> ● Post Created Confirmation ● Staff account verification ● username ● post title ● date ● post type ● post description ● post image ● calendar
Logic Summary	<pre> graph TD subgraph "2.1 Create a Post" direction TB A[username, post title, date, post type, post description, post image] --> B[Create a Post] B -- "Post Created Confirmation" --> C[username, post title, date, post type, post description, post image] C -- "staff account verification" --> D[Staff] end subgraph "2.2 View a Post" direction TB E[username, post title, date, post type, post description, post image] --> F[View a Post] F -- "User" --> G[User] G -- "Student" --> H[Student] end subgraph "2.3 Access Calendar" direction TB I[username, post title, date, post type, post description, post image] --> J[Access Calendar] J -- "calendar, activity" --> K[User] K -- "User" --> L[User] L -- "calendar, activity" --> M[User] end subgraph "2.4 Receive Notification" direction TB N[post title, date, post type, post description] --> O[Receive Notification] O -- "User" --> P[User] end C -- "username, post title, date, post type, post description, post image" --> E F -- "username, post title, date, post type, post description, post image" --> I I -- "username, post title, date, post type, post description, post image" --> N G -- "User" --> L L -- "User" --> M K -- "User" --> P </pre>

Table 3.4: Process Description of Bookmark Management

Process Name	3 - Bookmark Management
Description	Process for manage bookmarks
Inbound data	<ul style="list-style-type: none"> • username • post title • date • post type • post description • post image
Outbound Data	<ul style="list-style-type: none"> • username • post title • date • post type • post description • post image • student account verification
Logic Summary	<pre> graph TD Student((Student)) --> 3_1((3.1 View a Post)) 3_1 -- "username, post title, date, post type, post description, post image" --> 3_2((3.2 Add a post to bookmark)) 3_2 -- "username, post title, date, post type, post description, post image" --> 3_3((3.3 View Bookmark)) 3_3 -- "post title, date, post type, post description" --> Student </pre>

3.3.2.2 Data Stores

This section describes the data stores that exist in the data flow diagram and consists of the Data Store Name, Description, Inbound Data, and Outbound Data.

Table 3.5: List of all Data Stores

No.	Data Store	Name	Description
1	User	User Database	Collects information related to user accounts, including email, password, and confirmation details. Manages account creation messages and verification processes.
2	Post	Post Database	Collect data associated with posts, such as post title, date, type, description, and images. Handles notifications for new posts and confirms post creation.
3	Bookmark	Bookmark Database	Collects data related to user bookmarks, including favorite posts and bookmark lists. Handles user interactions with bookmarked content.

Table 3.6: Data Store Description of User Database

Data Store Name	User - User Database
Description	Collect user accounts, including email, password, and confirmation details. Handles account creation, verification, and messages.
Inbound data	<ul style="list-style-type: none"> • username • first name • last name • email • contact number • profilePic • account verification • password • new password
Outbound Data	<ul style="list-style-type: none"> • username • first name • last name • email • contact number • profilePic • new password • password

Table 3.6: Data Store Description of Post Database

Data Store Name	Post - Post Database
Description	Collect post details like title, date, type, and images. Manages notifications for new posts and post confirmation.
Inbound data	<ul style="list-style-type: none"> • username • post title • date • post type

	<ul style="list-style-type: none"> • post description • post image • calendar • activity
Outbound Data	<ul style="list-style-type: none"> • username • post title • date • post type • post description • post image

Table 3.7: Data Store Description of Bookmark Database

Data Store Name	Bookmark - Bookmark Database
Description	Collect user bookmarks, including favorite posts and bookmark lists. Manages interactions with bookmarked content.
Inbound data	<ul style="list-style-type: none"> • post title • date • post type • post description
Outbound Data	<ul style="list-style-type: none"> • username • post title • date • post type • post • description • post image

3.3.2.3 Data Element

This section describes the data elements or data flows that exist in this system. The table below contains the list of all data elements belonging to their data element name, starting process/source/data store, and ending process/source/data store.

Table 3.8: List of All Data Elements

SEQ	Data Element Name	From Process/Source/Data Store	To Process/Source/Data Store
1	email	1.1 - Register For An Account	1.3 - Login
2	password	1.1 - Register For An Account	1.3 - Login
3	confirm password	1.1 - Register For An Account	1.3 - Login
4	username	1.1 - Register For An Account	1.3 - Login
5	first name	1.1 - Register For An Account	1.3 - Login
6	last name	1.1 - Register For An Account	1.3 - Login
7	contact number	1.1 - Register For An Account	1.3 - Login
8	profilePic	1.1 - Register For An Account	1.3 - Login
9	new password	1.1 - Register For An Account	1.3 - Login
10	account verification	1.1 - Register For An Account	1.3 - Login
11	Post Created Confirmation	2.1 - Create a Post	2.2 View a Post
12	Staff account verification	2.1 - Create a Post	2.2 View a Post

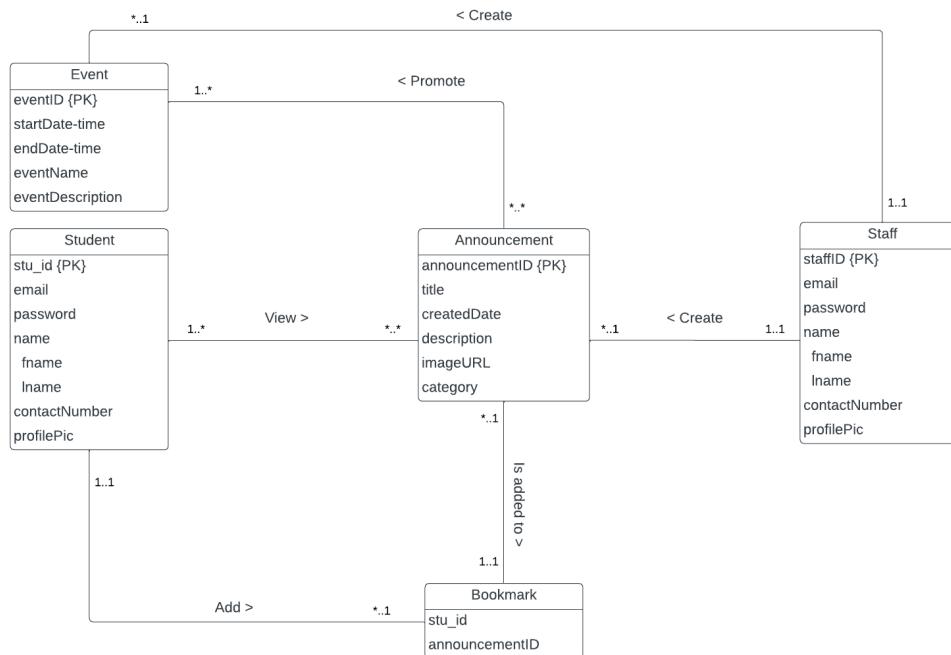
SEQ	Data Element Name	From Process/Source/Data Store	To Process/Source/Data Store
13	username	2.1 - Create a Post	2.2 View a Post
14	post title	2.1 - Create a Post	2.2 View a Post
15	date	2.1 - Create a Post	2.2 View a Post
16	post type	2.1 - Create a Post	2.2 View a Post
17	post description	2.1 - Create a Post	2.2 View a Post
18	post image	2.1 - Create a Post	2.2 View a Post
19	calendar	2.1 - Create a Post	2.2 View a Post
20	post title	3.1 - View a Post	3.2 - Add a Post To Bookmark
21	date	3.1 - View a Post	3.2 - Add a Post To Bookmark
22	post type	3.1 - View a Post	3.2 - Add a Post To Bookmark
23	post description	3.1 - View a Post	3.2 - Add a Post To Bookmark

3.4 Database Analysis and Design

This section will describe the topic of database analysis and design for MUICT Connect application including ER diagram, relational schema, and file structure. Each of these is fully described in detail along with diagrams and illustrations.

MACROBUTTON AcceptAllChangesInDoc <Lastname> MACROBUTTON
 AcceptAllChangesInDoc <Lastname> and MACROBUTTON
 AcceptAllChangesInDoc <Firstnames> MACROBUTTON
 AcceptAllChangesInDoc <Lastname> STYLEREF "Heading 1" V-
 MERGEFORMAT Analysis and Design / PAGE 18

3.4.1 ER-Diagram



Project : MUICT Connect

System : MUICT Connect

Major Advisor : AJ. SNIT SANGHLAO

Description : This is an ER-Diagram in UML notation of the MUICT Connect database system consisting of 5 entities.

Figure 3.4: ER-Diagram of MUICT Connect Database

3.4.2 Relational Schema

This section describes the attributes of the tables in the database. The attribute notation is shown below.

- **Attributes** – which are bold and underlined are the Primary Keys
- *Attributes* – which are Italic are the Foreign Keys
- **Attributes** – which are bold, italic and underlined are both Primary Keys and Foreign Keys

Tables in this system can be divided into 3 groups as follows:

- Master File Table
- Base File Table
- Transaction File Table

Table 3.9: List of all Tables in Our System Database

Table#	Table Name	Table Type	Description
1	Student	Base	Stores student's information
2	Staff	Base	Stores staff's information
3	Announcement	Master	Stores announcement content
4	Event	Master	Stores event's schedule and content
5	Bookmark	Transaction	Stores bookmarked announcement posts made by student

1. Relational Schema of Master File Tables
 - Announcement
 - Event
2. Relational Schema of Base File Tables
 - Student
 - Staff
3. Relational Schema of Transaction File Tables
 - Bookmark

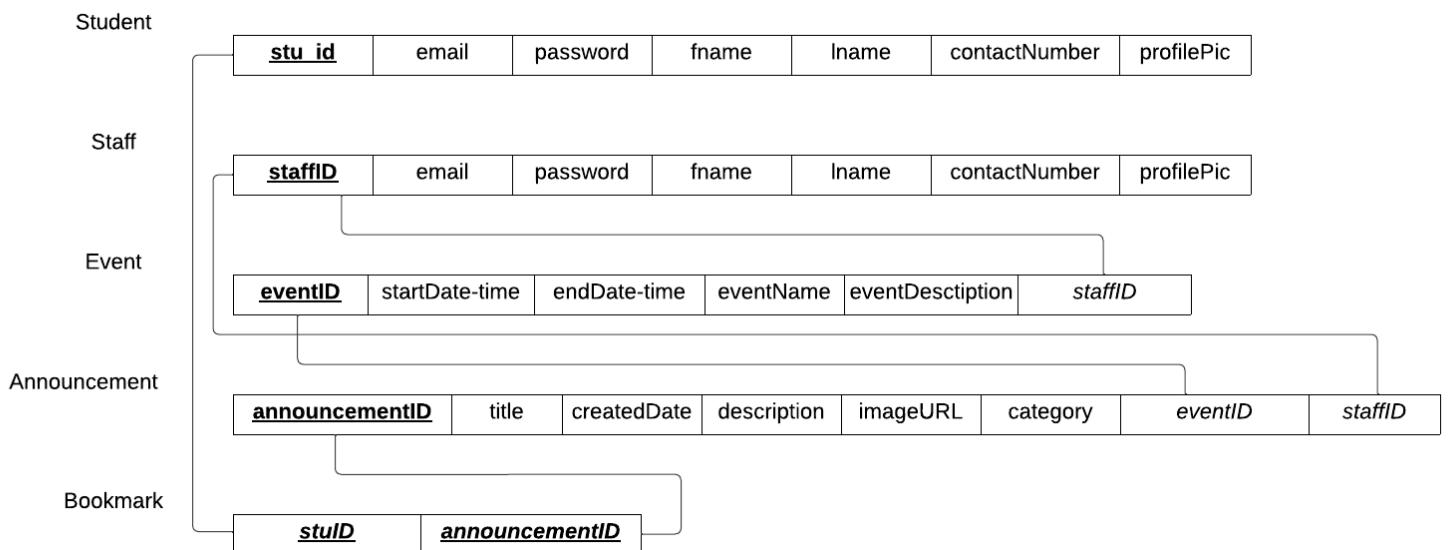


Figure 3.5: Relational Schema of MUICT Connect Database

3.4.3 **File Structure**

This section shows the details of each file component including field name, field description, field data type, field length, null value, primary key and foreign key.

Table 3.10: File Structure of Student

Table Name : Student							Table# 1
Table Type : Base							
Description : This is the file structure of 'Student' that stores student's information							
Field Name	Type	Length	Description	Key	Reference	Null	
stu_ID	INT	1 Byte	Student ID (e.g. 6488xxx)	PK	-	-	
email	VARCHAR(50)	50 Bytes	Student email (xxx@student.mahidol.edu)	-	-	-	
password	VARCHAR(20)	20 Bytes	Student's password	-	-	-	
fname	VARCHAR(50)	50 Bytes	Student's first name	-	-	-	
lname	VARCHAR(50)	50 Bytes	Student's last name	-	-	-	
contactNumber	VARCHAR(15)	15 Bytes	Student's contact number (e.g. 08xxxxxxxx)	-	-	-	
profilePic	BLOB	Upto 65,535 Bytes	Profile picture to be displayed	-	-	Y	
Total		~66	KBytes				

Table 3.11: File Structure of Staff

Table Name : Staff							Table# 2
Table Type : Base							
Description : This is the file structure of ‘Staff’ that stores staff’s information							
Field Name	Type	Length	Description	Key	Reference	Null	
staffID	INT	1 Byte	Staff ID	PK	-	-	
email	VARCHAR(50)	50 Bytes	Staff email (xxx@mahidol.edu)	-	-	-	
password	VARCHAR(20)	20 Bytes	Staff’s password	-	-	-	
fname	VARCHAR(50)	50 Bytes	Staff’s first name	-	-	-	
lname	VARCHAR(50)	50 Bytes	Staff’s last name	-	-	-	
contactNumber	VARCHAR(15)	15 Bytes	Staff’s contact number (e.g. 08xxxxxxxx)	-	-	-	
profilePic	BLOB	Upto 65,535 Bytes	Profile picture to be displayed	-	-	-	Y
Total		~66	KBytes				

Table 3.12: File Structure of Event

Table Name : Event							Table# 3
Table Type : Master							
Description : This is the file structure of 'Event' that stores event's information							
Field Name	Type	Length	Description	Key	Reference	Null	
eventID	INT	1 Byte	Uniquely identified event ID	PK	-	-	
startDate-time	DATETIME	8 Bytes	Start date and time of the event	-	-	-	
endDate-time	DATETIME	8 Bytes	End date and time of the event	-	-	-	
eventName	VARCHAR(50)	50 Bytes	Name of the event	-	-	-	
eventDescription	VARCHAR(150)	150 Bytes	Short description of the event	-	-	-	
staffID	INT	1 Bytes	Staff who is responsible for the event	FK	Staff	-	
Total		218	Bytes				

Table 3.13: File Structure of Announcement

Table Name : Announcement						Table# 4
Table Type : Master						
Description : This is the file structure of ‘Announcement’ that stores announcement’s information						
Field Name	Type	Length	Description	Key	Reference	Null
announcementID	INT	1 Byte	Uniquely identified announcement ID	PK	-	-
title	VARCHAR(50)	50 Bytes	Title of the announcement	-	-	-
createdDate	DATETIME	8 Bytes	Created date and time of the event	-	-	-
description	VARCHAR(1000)	1000 Bytes	Short description of the announcement	-	-	-
imageURL	VARCHAR(500)	500 Bytes	URL or Path to the image	-	-	-
category	VARCHAR(20)	20 Bytes	Category of the announcement	-	-	-
eventID	INT	1 Byte	Event specified in the announcement	FK	Event	-
staffID	INT	1 Byte	Staff who is responsible for the post	FK	Staff	-
Total		1581	Bytes			

Table 3.14: File Structure of Bookmark

Table Name :	Bookmark						Table# 5
Table Type :	Transaction						
Description :	This is the file structure of ‘Bookmark’ that stores bookmark’s information						
Field Name	Type	Length	Description	Key	Reference	Null	
stu_ID	INT	1 Byte	Student ID (e.g. 6488xxx)	PK, FK	Student	-	
announcement_ID	INT	1 Byte	Uniquely identified announcement ID	PK, FK	Announcement	-	
Total		2	Bytes				

3.5 I/O Design

This section provides an explanation of the Input and Output User Interface's design. The segment is divided into two components: the interface design and the transition diagram, which illustrates the sequential progression within the system.

3.5.1 Interface Design

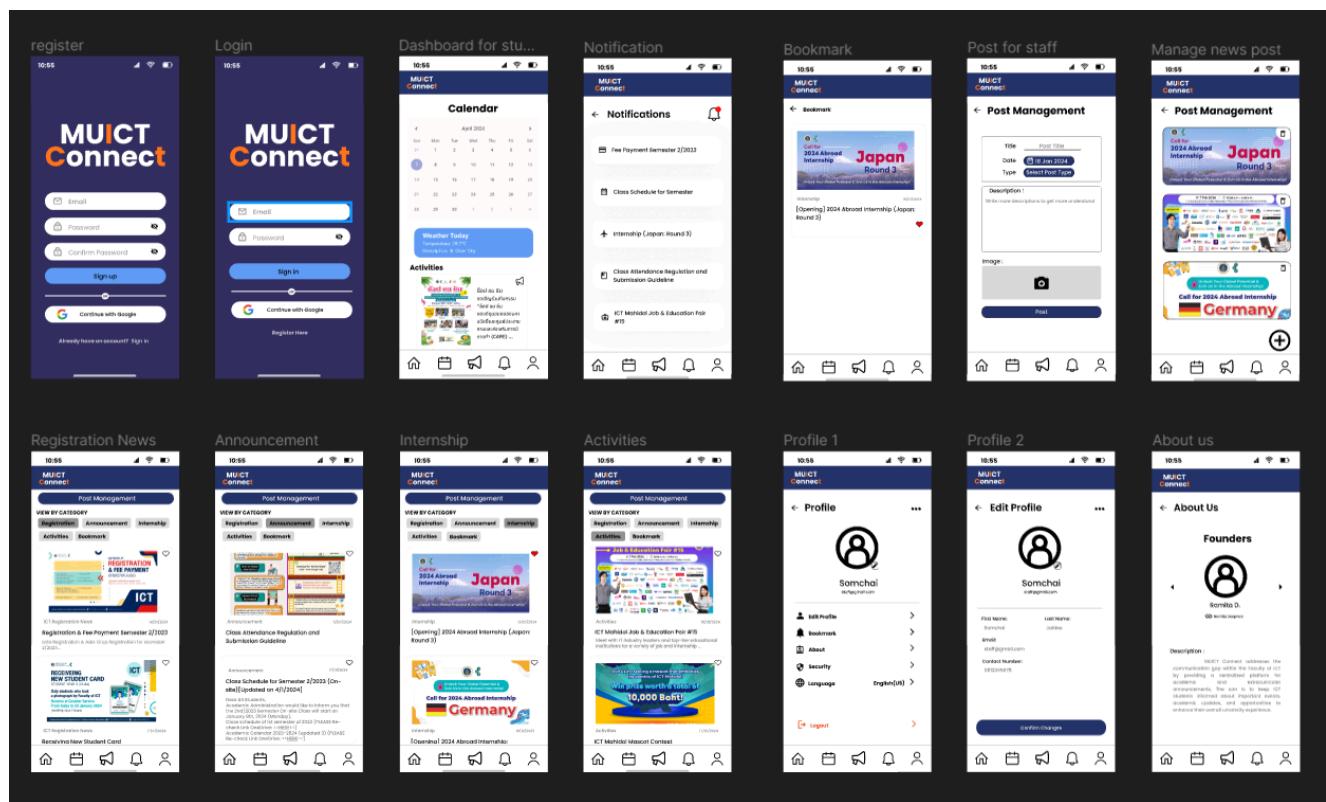


Figure 3.5.1.1: Interface Design of MUICT Connect application

Figma Link: [MUICT Connect User Interface Design](#)

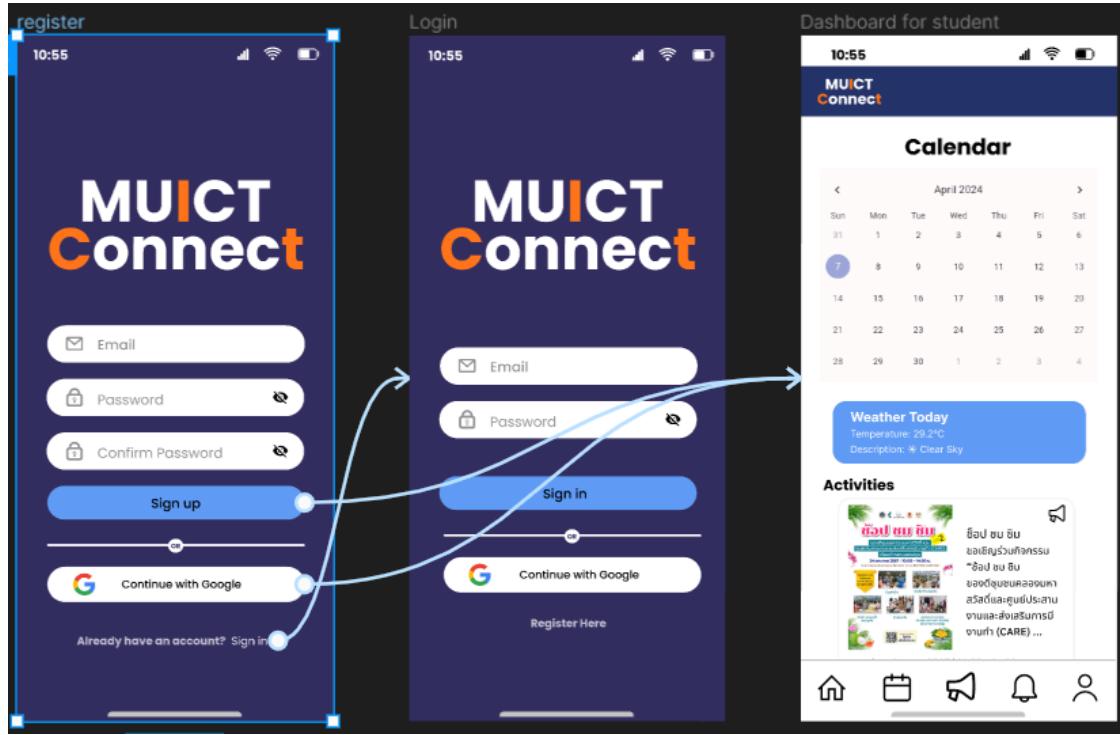


Figure 3.5.1.2: Registration/Login Interface

Registration/Login is the primary interface of our application. Users are redirected to the dashboard page, which displays a calendar of recent and important activities once they have successfully logged in.

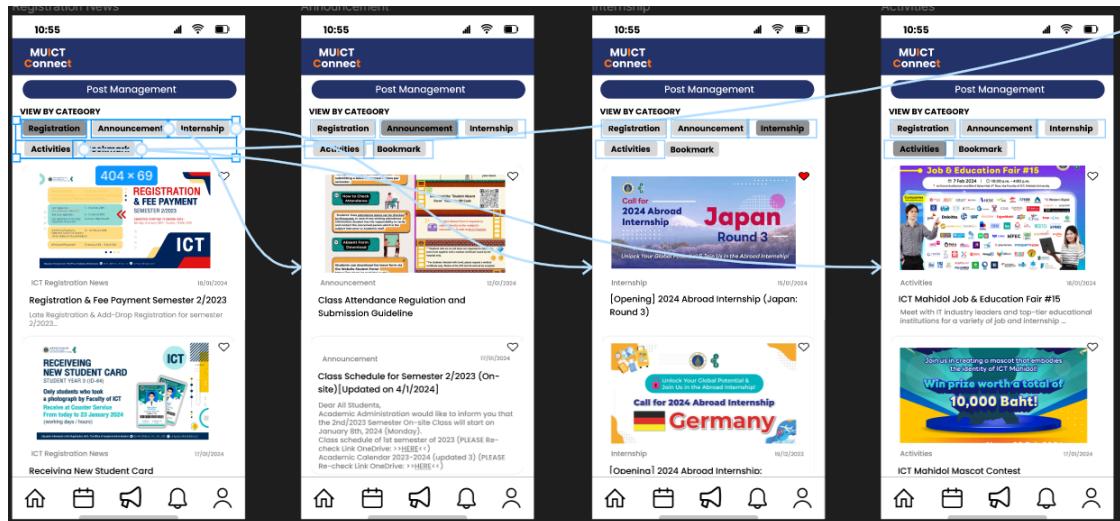


Figure 3.5.1.3: Interface of Four Announcement Types

The news page proceeds as follows. News posts categorized as Registration, Announcement, Internship, or Activity are also accessible to users on this page. By clicking "Add to Bookmarks," users may indicate their interest in a specific post.

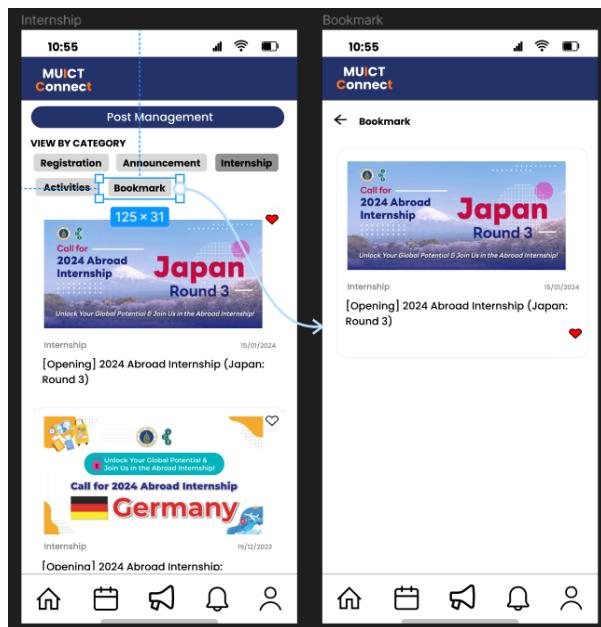


Figure 3.5.1.4: Bookmark Interface

The post will be added to the bookmark category, which users can access later, upon clicking the "Add to Bookmarks" icon.

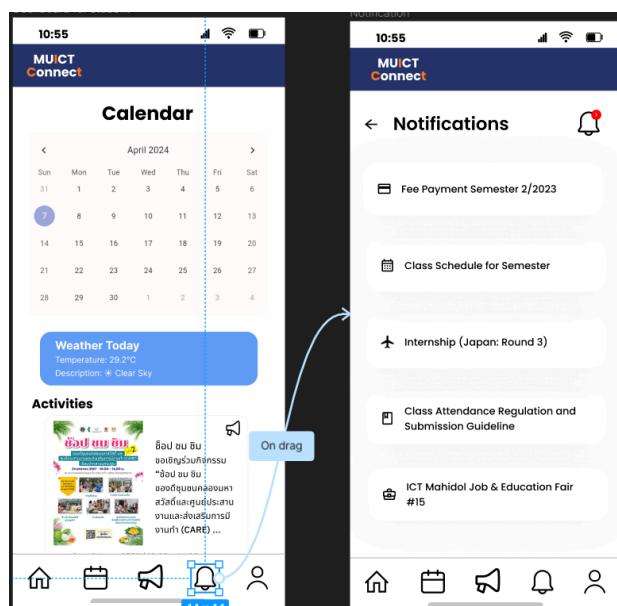


Figure 3.5.1.5: Calendar/Notification Interface

Notifications may also be delivered to consumers by the application. The notifications displayed can be accessed by users through the notification icon in the navigation bar at the bottom of the screen.

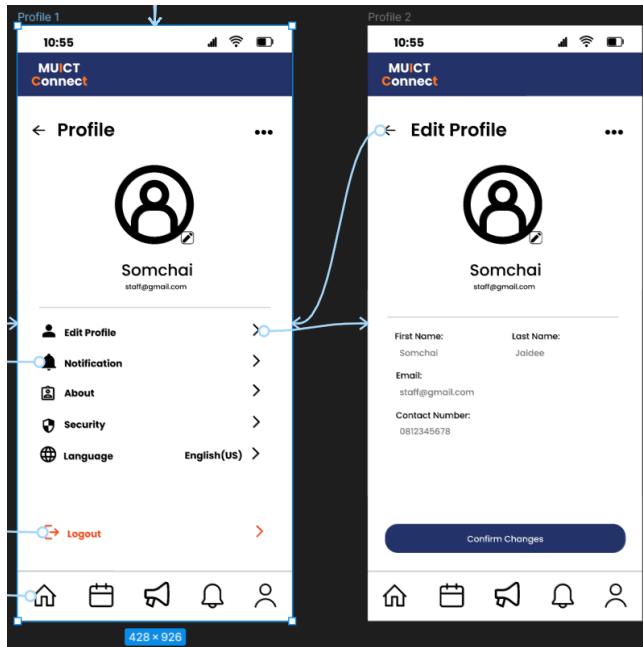


Figure 3.5.1.6: Profile, Edit Profile Interface

Users can view their profiles in this particular section. Through the "Edit Profile" link, they may also modify their profile.

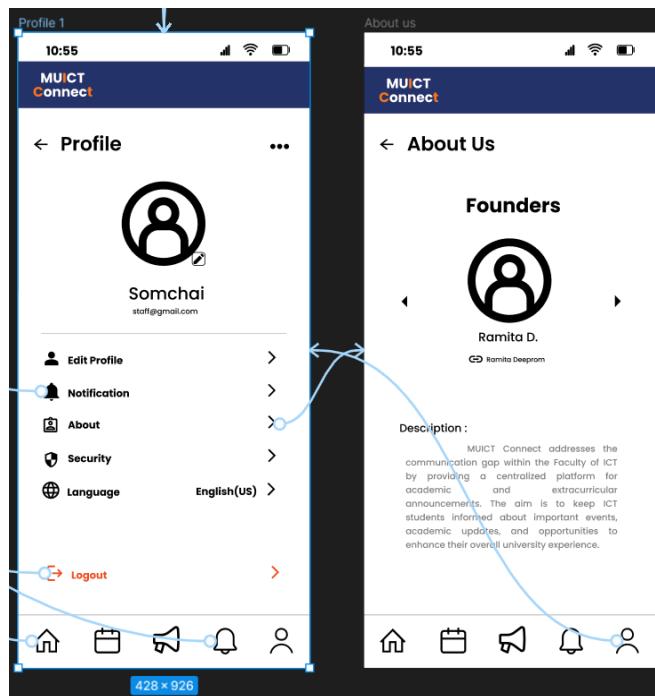


Figure 3.5.1.7: About Us Interface

The "About Us" page, which provides information about the founders and developers of MUICT Connect, will be accessed when the "About" button is clicked by the user.

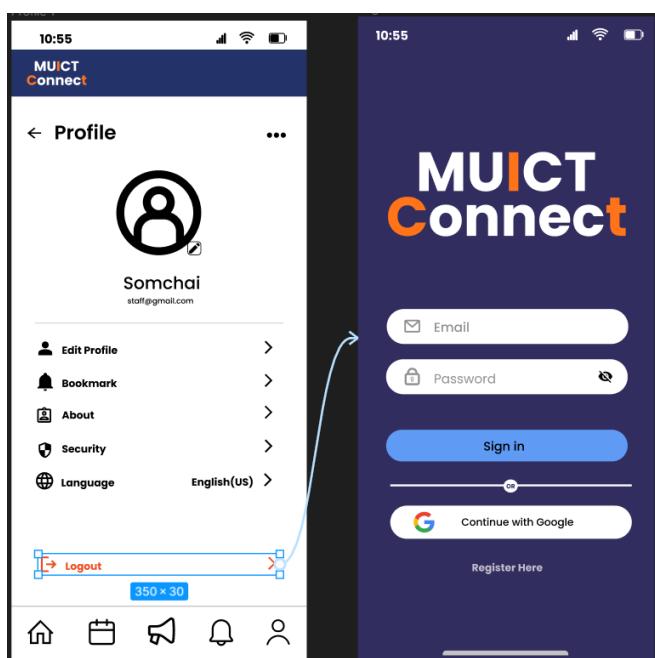


Figure 3.5.1.8: Logout Interface

By selecting the "Logout" button, users will be logged out of the system and directed back to the "Login" page.

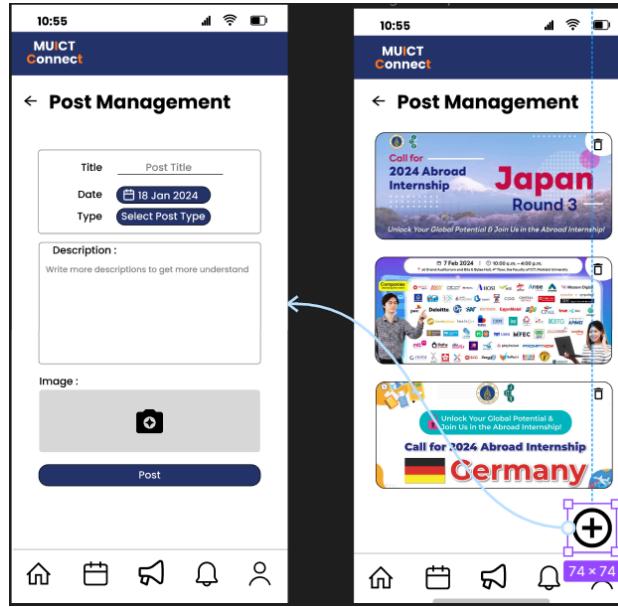


Figure 3.5.1.9: Post management

Users have the ability to execute CRUD operations (i.e., create, modify, and delete posts) on the posts through the application menu by selecting the Megaphone icon.

3.5.2 Transition Diagram

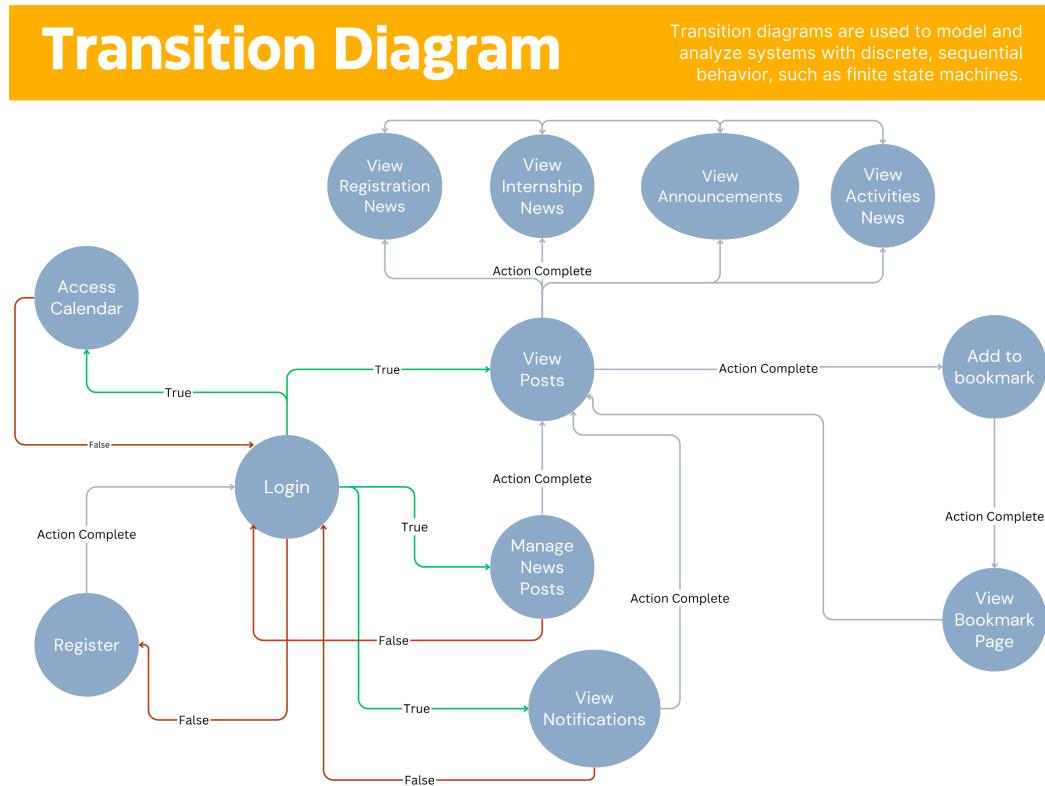


Figure 3.5.2.1: Transition Diagram

This is the transition diagram for the MUICT Connect application. Start with the login state; users must log in before accessing the application. If not, they must complete registration. Users can conduct a variety of things after logging in, including viewing news postings, accessing the calendar, managing news posts (staff), and seeing notifications.

After viewing the news posts, visitors can bookmark any posts that attract their interest for after viewing on the bookmark page.

CHAPTER 4

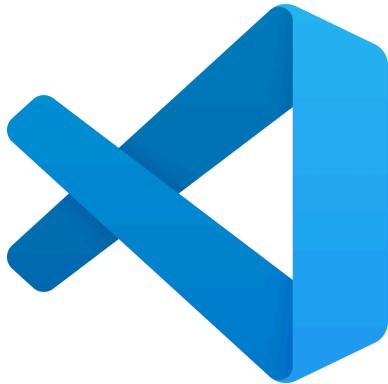
4 IMPLEMENTATION

This chapter discusses the hardware and software environments, as well as the implementation guide and methodologies for our system.

4.1 Hardware and System Environment

- Operating System and Utilities Applications
 - Laptop hardware 1: MacOs M1, Core 16 CPU 3.2GHz, RAM 16 GB, Storage 512 GB
 - Laptop hardware 2: MacOs M2, Core 32 CPU 3.2GHz, RAM 32 GB, Storage 512 GB
 - Laptop hardware 3: Lenovo Ryzen9, Core 16 CPU 2.9GHz, RAM 32 GB, Storage 1TB
- Software System
 - Operation system: macOS Sonoma Version 14.3.1 (23D60)
 - Android studio, Firebase, Flutter, Visual Studio, Microsoft Word, Google Chrome, Safari.
 - Laptop hardware 3: Lenovo Ryzen9, Core 16 CPU 2.9GHz, RAM 32 GB, Storage 1TB
- Web Server Software
 - Google Cloud allows users to grow their app to billions of clients. As a result, users may quickly integrate Google Cloud products as their team or infrastructure grows.
 - Consider a Google Cloud Platform project as a virtual container for data, code, configuration, and services, independent of how it was generated. When users establish a Firebase project, they are generating a Google Cloud project in the background.
 - Firebase and Google Cloud share three products: Cloud Firestore, Cloud Functions, and Cloud Storage.

- o This project focuses primarily on the implementation of the Cloud Firestore, a massively scalable real-time NoSQL database.
- Editor
 - o Visual Studio Code



Visual Studio Code (VS Code) is a lightweight and versatile code editor developed by Microsoft. While it's not a full-fledged integrated development environment (IDE) like Android Studio, it has become popular among developers for various programming languages, including web development, Python, and more. Cross-Platform

Compatibility: VS Code is accessible on Windows, macOS, and Linux, making it a versatile solution for developers who work with multiple operating systems. Extensions: VS Code offers a large number of extensions, allowing developers to customize their environment with numerous plugins and tools, thereby improving their productivity.

Git Integration: VS Code effortlessly interacts with Git, allowing you to use version control capabilities right from the editor. Debugger: It includes a built-in debugger that supports many languages, allowing developers to quickly detect and resolve code bugs. Visual Studio Code has a Live Share feature that allows for collaborative coding sessions in real-time, which can be handy for team-based projects.

- o Android Studio



The key program for carrying out this project is Android Studio. We utilize this as the main program because Android Studio is currently receiving multiple updates that expand its features. It is well-suited for fast-paced development while maintaining high-quality app output across all Android devices.

The advantage of Android Studio, as we can see, is that it enables speedier coding. Instant Run, for example, detects changes in the code as they happen. This modification is accessible without the need to restart or rebuild your application.

Furthermore, an Intelligent Code Editor allows for significantly speedier development. While the user is creating code, the editor suggests sophisticated code compilation and analysis. The dropdown selection makes it easy to check the recommendation.

- Database Management System (DBMS)

- o For database management systems, we use Google Cloud Firestore to alter and connect the application to the database. The Firebase Cloud Firestore serves as an intermediate between database and application data. Even if we inadvertently destroy the database or some app parts, data will remain in the Cloud.

For example, registering will keep the user's information in Firebase to prevent data loss while also transferring learning and quiz content to the Firestore database for display in the main application.

- Programming and Scripting Tools

- Flutter



Flutter is a UI toolkit and framework created by Google to create natively built applications for mobile, web, and desktop from a single codebase.

- Dart



Dart is the primary programming language used in the project's implementation. We chose Dart as our primary programming language since it is a client-oriented programming language that enables users to build quick apps on any platform. It aims to provide the most

productive programming language for cross-platform development, as well as a versatile runtime platform for app frameworks.

Dart may be of assistance to us. A language is defined by its technical envelope, which includes the choices made during development that shape the language's capabilities and strengths. Dart is designed for client development, focusing on both development and high-quality production experiences across a wide range of compilation targets (web, mobile, and desktop).

- Components

- This section describes the components of the application “MUICT Connect” The first stage is to register users, which requires them to enter information such as their name, email address, and password. In the second phase, known as login, the user inputs their username and password. If they are entered correctly, they lead to the main page; if not, they need to enter them again.
- On the main page (Dashboard), Calendar and Activity are displayed. On the bottom bar, you can click to go to the home page, bookmarks page, notification page, and profile. We can press the profile page to edit our personal information and to check upcoming activities, including the bookmark page where we have saved information for viewing. Our app mainly focuses on notifications. About various types of news.

4.2 Implementation Guide and Techniques

This project requires the installation of certain plugins, including FlutterFire. FlutterFire is a collection of Flutter plugins that connect your Flutter project to Firebase. Before we can use any Firebase services, we must first install the 'firebase_core' plugin. As a result, we must run the command 'flutter pub add firebase_core' which will install the plugin as displayed on the box's first line.

To use Cloud Firestore in our Flutter project, we must also install the 'cloud_firestore' plugin. As a result, we also need to run the command 'flutter pub add cloud_firestore,' then it will install the plugin as shown on the second line in the box.

4.2.1 Connecting Guide

```
firebase_core: ^1.12.0
```

```
cloud_firestore: ^1.0.5
```

4.2.2 ListView Technique

The initial step is to enter information into the system. The data will then be displayed in a list view, allowing users to select which lists to learn or quiz. This list was generated with the ListView.builder() function.

The ListView.builder() function was used to generate this list. Begin with the listview before proceeding to the procedure for returning the listview. As a result, we utilized the content concerning learning and quizzes to create the list. It will appear as a scrollable row at the bottom of the page.

```
ListView.builder()
```

CHAPTER 5

5 TESTING AND EVALUATION

This chapter discusses the methodology employed by the researchers to evaluate the specific functionalities of this application. This chapter is segmented into two sections. The initial section consists of the unit tests, whereas the subsequent section comprises the system integration tests. Hence, the readers will acquire a deeper understanding of the testing procedure, the outcomes of the tests, the illustrative instances, and a comprehensive elucidation on how to utilize the program.

5.1 Unit Tests

For the unit tests, we selected some important and critical processes for formal unit testing. The selected processes include:

- Process No. 1: Register
- Process No. 2: Login
- Process No. 3: Post Management
- Process No. 4: View Post

5.1.1 Test Performed on No. 1: Register

For the Register function, there are 3 fields for users to input the data which are E-mail, Password and Confirmation Password.

Operation Performed	Condition Tested	Actual Result
Enter valid email and password	All fields are valid	Registration succeeds
Leave email field empty	Email field is a blank	Error message shown
Enter invalid email format	Email format is incorrect	Error message shown
Leave password field empty	Password field is a blank	Error message shown
Leave confirmation password field empty	Confirmation Password field is a blank	Error message shown

Enter a password with less than minimum required characters	Password too short	Error message shown
Use previously registered email	Email already exists in the system	Error message shown
Register with Google	Login	

Table 5.1: Test Performed on No. 1: Register

5.1.2 Test Performed on No. 2: Login

For the Login function, there are 2 fields for the user to input the data, which are Email and Password.

Operation Performed	Condition Tested	Actual Result
Enter valid email and password	All fields are valid	Login succeeds, navigates to Dashboard
Enter incorrect email	Email is incorrect	Error message shown
Enter incorrect password	Password is incorrect	Error message shown
Leave email field empty	Email field is blank	Error message shown
Leave password field empty	Password field is blank	Error message shown
Enter email in invalid format	Email format is incorrect	Error message shown
Enter correct email with wrong password from a different account	Correct Email with Password from a different account	Error message shown
Try to login with Google account	Successful Google Auth	Login succeeds, navigates to Dashboard
Try to login with Google but cancel authentication	Google Auth cancelled or failed	Error message shown

Table 5.2: Test Performed on No. 2: Login

5.1.3 Test Performed on No. 3: Post Management

Operation Performed	Condition Tested	Actual Result
Delete a post: Users click on the delete button at the post that they would like to delete	The post is deleted	Pass
Create a post: A warning message: All fields - Users submitted nothing to the system	The system shows a warning message under every empty box to inform users to enter their information completely.	Pass
Create a post: A warning message: All fields - Users submitted all appropriate registered data to the system	The system shows a warning message box below the screen to notify users if the registration is successful: “Successful !”	Pass

Table 5.3: Test Performed on No. 3: Post Management

5.1.4 Test Performed on No. 4: View Post

Operation Performed	Condition Tested	Actual Result
View by category: Users click on the category they want to see.	The system only shows posts from the chosen category	Pass
Add to bookmark. Users click the heart button on the post that they are interested in.	The system adds the post to the bookmark and it can be seen in the bookmark page	Pass

Table 5.4: Test Performed on No. 4: View Post

5.2 System Integration Test

This activity is performed after the system is completely integrated. The purpose of this testing is to check whether the system can operate correctly according to the required functions or not.

5.2.1 Test Scenario

In order to test all functional aspects of the system thoroughly, we had set up a test scenario which consisted of phases as shown below.

- Register: Users can register in the system.
- Login: Users authenticate themselves to gain access to the system.
- Post Management: Users can enter and edit posts.
- View Post: Users can view posts and can bookmark them.

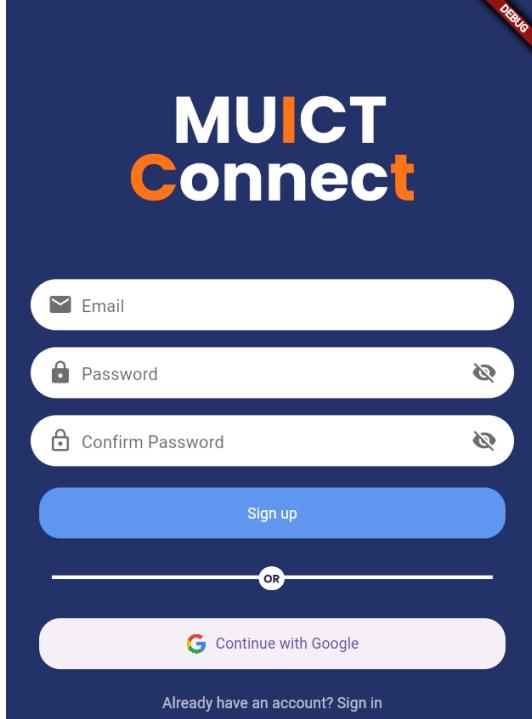
Moreover, the test scenario can be used as a user guideline because it covers all the steps necessary to use our system. The details of each phase are shown in the next section.

5.2.1.1 Register

- This is a registration page where users can sign up to join the "MUICT Connect" platform, presumably a community or service. The page is designed for users to create their own account by entering personal credentials.

The form comprises

- Email: Please enter your valid email address here. This will be used as your username to log into the platform.
- Password: Create a password for your account. It should be unique and secure, meeting any specified

	<p>criteria for strength.</p> <ul style="list-style-type: none">● Confirm Password: Re-enter your password here to confirm it. This step ensures that there are no typos in your password.● Sign Up Button: After filling in your email and password, click this button to create your new account.
	<ul style="list-style-type: none">● Continue with Google: For quicker registration, use this option to sign up with your existing Google account. This will link your Google account credentials for easy login access in the future.● Below the main sign-up options is a link for users who already have an account to sign in, indicating that they can navigate to the login page if they already possess account credentials.

5.2.1.2 Login

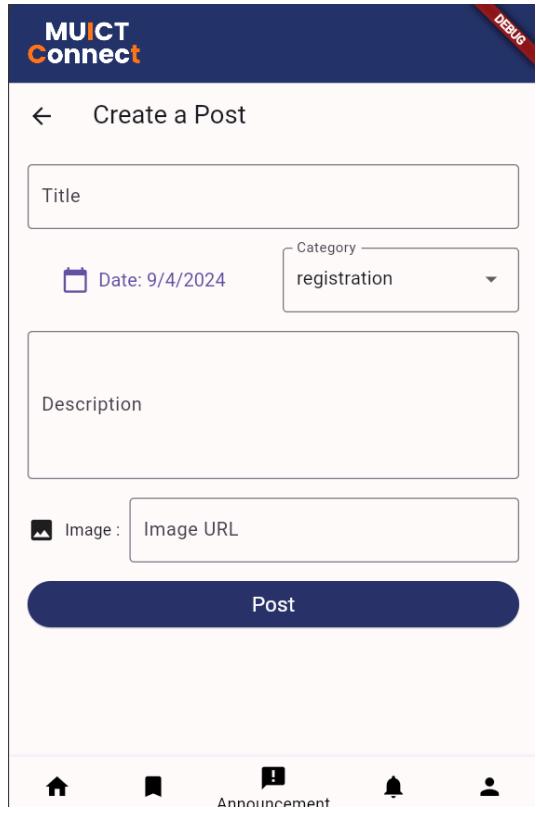
The screenshot shows two overlapping windows. The top window is the MUICT Connect login page, featuring a dark blue header with the "MUICT Connect" logo. It has fields for "Email" and "Password", a "Sign in" button, and a "Continue with Google" button. Below these is a "Register here" link. The bottom window is a Google sign-in modal titled "Sign in - Google Accounts - Person 1 - Microsoft Edge". It displays a message: "Google sign-in has a new look. We've improved the sign-in page with a more modern design." It includes "Learn more" and "Dismiss" buttons, and a "Sign in with Google" button. At the bottom of the modal, there's a note: "To continue, Google will share your name, email address, language preference, and profile picture with mucit-connect-GoogleAuth."

- This is a registration page where users can log in to the "MUICT Connect" platform, presumably a community or service. The page is designed for users to input their credentials and gain access.

The form comprises

- Email: Please enter your valid email address here. This will be used as your username to log into the platform.
- Password: Enter a password for your account. It should be unique and secure, meeting any specified criteria for strength.
- Continue with Google: For quicker registration, use this option to sign up with your existing Google account. This will link your Google account credentials for easy login access in the future.
- Below the main login options is a link for users who have yet to register, indicating that they can navigate to the register page if they need to.

5.2.1.3 Post Management

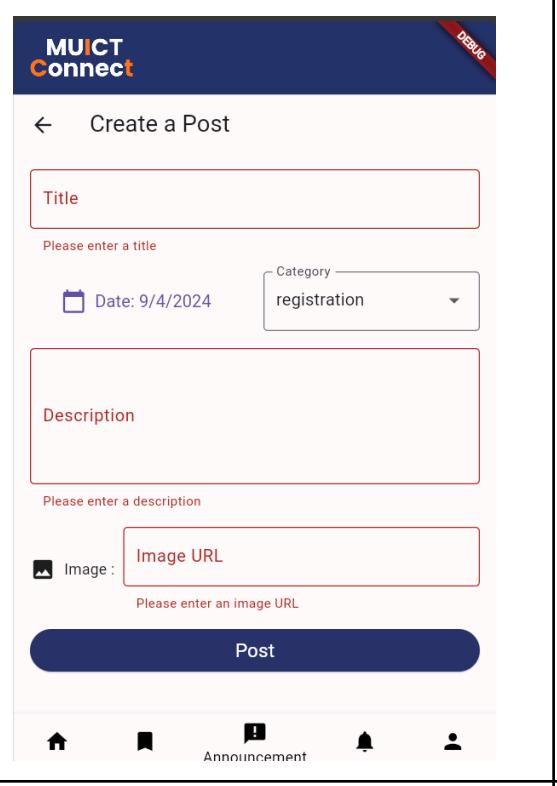
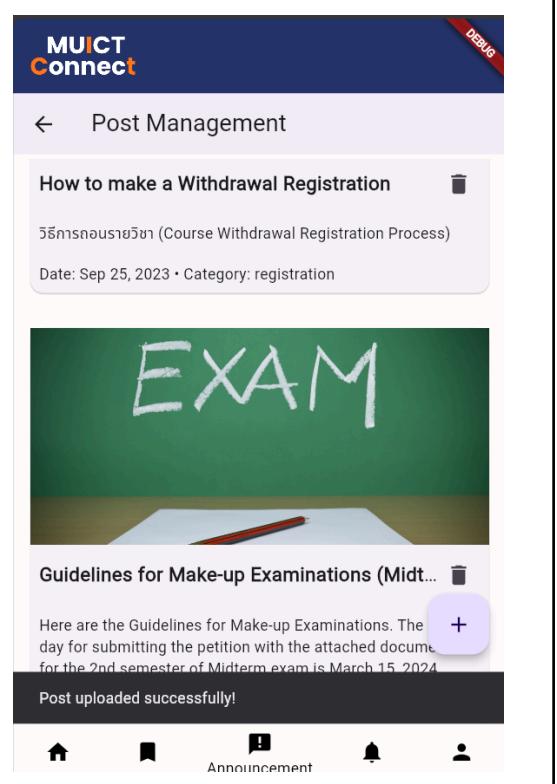


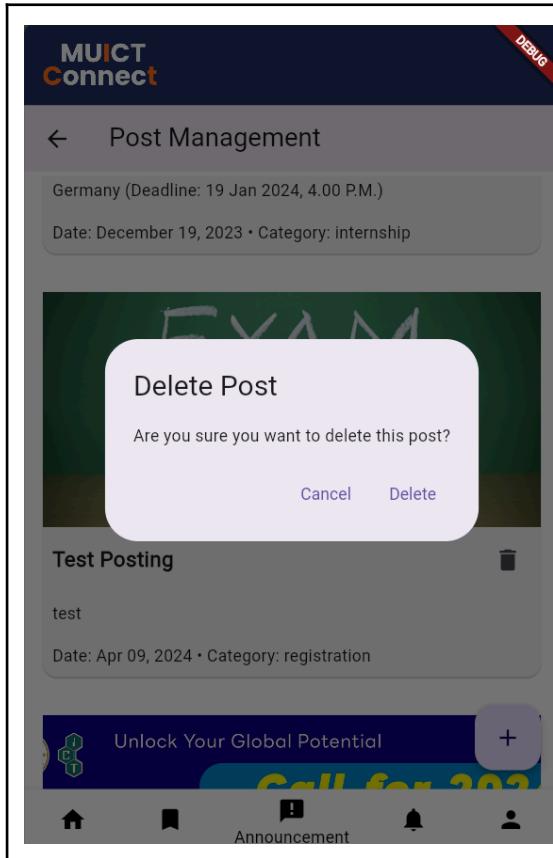
The screenshot shows the 'Create a Post' page of the MUICT Connect application. At the top, there is a header bar with the MUICT Connect logo and a 'DEBUG' button. Below the header, the title 'Create a Post' is displayed. The form consists of several input fields: 'Title' (empty), 'Date: 9/4/2024' (with a calendar icon), 'Category' (set to 'registration'), 'Description' (empty), and 'Image URL' (empty). A large blue 'Post' button is at the bottom of the form. Below the form, there is a navigation bar with icons for Home, Bookmarks, Announcements, Notifications, and Profile, and the word 'Announcement'.

- This is a post page where users can enter post information and create a new post.

The form comprises

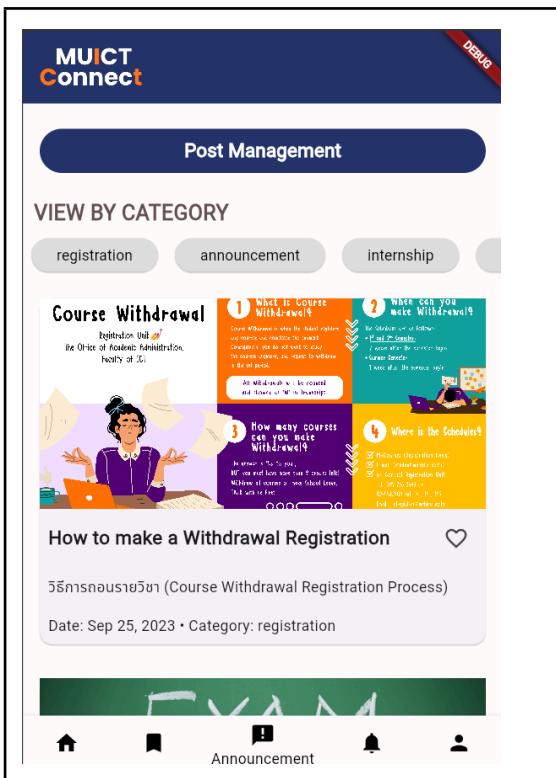
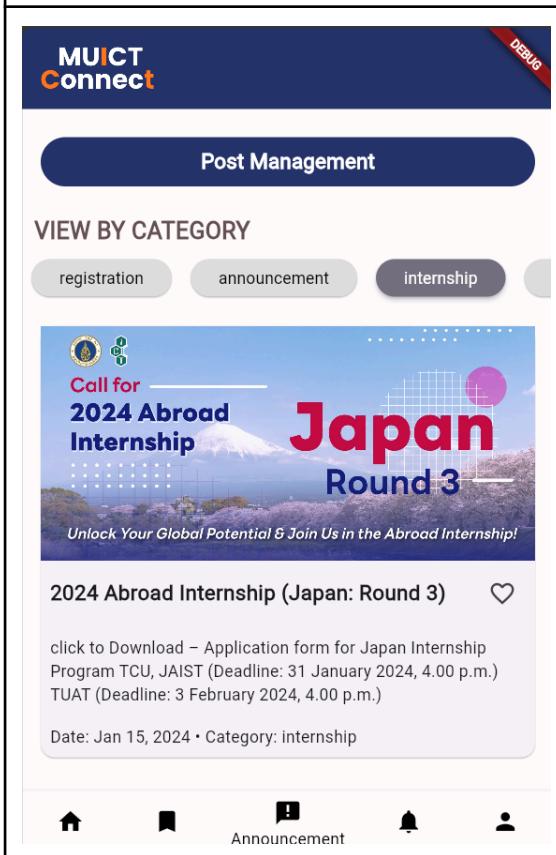
- Title: Please enter the title of your post here.
- Date: This section displays the current date.
- Category: This is a selection menu where you may choose the category for your article.
- Purpose: This is a section where you may enter a description or content for your post.
- Pictures: This feature allows you to add a picture to your post. You can enter the image URL in the designated text area.
- Post Button: After completing the essential information, click this button to submit or publish your post.

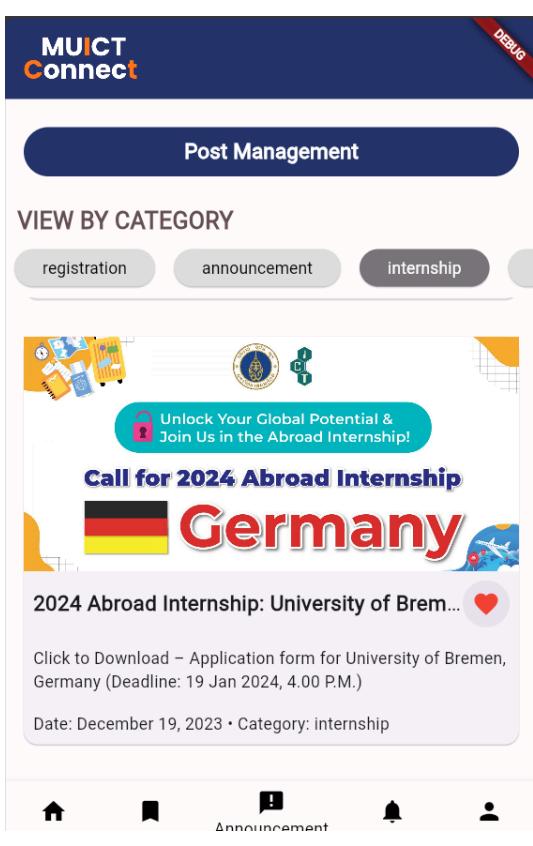
	<ul style="list-style-type: none"> If a user unintentionally clicks "submit" without entering any data, a warning message will appear below each empty field.
	<ul style="list-style-type: none"> Finally, it will appear the message “Successful!” below the screen when the users successfully create a post.



- Users can click on the delete button to delete a post. Once clicking on “Delete,” the post will be deleted.

5.2.1.4 View Post

 <p>Course Withdrawal Registration Unit The Office of Academic Administration, Faculty of ICT</p> <p>How to make a Withdrawal Registration</p> <p>วิธีการถอนรายวิชา (Course Withdrawal Registration Process)</p> <p>Date: Sep 25, 2023 • Category: registration</p>	<ul style="list-style-type: none"> This is a page to view a post. Each post shows a title, description, date, and category. Users can filter the posts by category. They can click on each category type to see only the posts related to each category. The page also contains bookmark functionality.
 <p>Call for 2024 Abroad Internship Japan Round 3</p> <p>Unlock Your Global Potential & Join Us in the Abroad Internship!</p> <p>2024 Abroad Internship (Japan: Round 3)</p> <p>click to Download – Application form for Japan Internship Program TCU, JAIST (Deadline: 31 January 2024, 4.00 p.m.) TUAT (Deadline: 3 February 2024, 4.00 p.m.)</p> <p>Date: Jan 15, 2024 • Category: internship</p>	<ul style="list-style-type: none"> Users can filter the posts by category. They can click on each category type to see only the posts related to each category. For example, once the users click on the “internship” category, they will see only the posts related to internship opportunities.

 <p>The screenshot shows the 'Post Management' section of the MUICT Connect app. At the top, there's a header with the MUICT Connect logo and a 'DEBUG' button. Below it is a 'Post Management' button. Underneath is a 'VIEW BY CATEGORY' section with three tabs: 'registration', 'announcement', and 'internship'. A large post card is displayed, featuring icons for travel and study, the university seal, and the German flag. The text reads: 'Unlock Your Global Potential & Join Us in the Abroad Internship!', 'Call for 2024 Abroad Internship', and 'Germany'. Below the post card, the text '2024 Abroad Internship: University of Brem...' is followed by a heart icon. Further down, it says 'Click to Download – Application form for University of Bremen, Germany (Deadline: 19 Jan 2024, 4.00 P.M.)' and 'Date: December 19, 2023 • Category: internship'. At the bottom of the screen are navigation icons for Home, Bookmarks, Announcement, and Profile.</p>	<ul style="list-style-type: none">• Users can click on the heart button to add the posts that they are interested in bookmarking.
 <p>The screenshot shows the 'Bookmarks' page of the MUICT Connect app. It features a back arrow and the word 'Bookmarks'. Below is the same post card as the previous screenshot, with the text '2024 Abroad Internship: University of Brem...', a heart icon, and the download information. At the bottom are navigation icons for Home, Bookmarks, Announcement, and Profile.</p>	<ul style="list-style-type: none">• After that, the post can be viewed on the bookmark page.

CHAPTER 6

6 CONCLUSIONS

The introduction of MUICT Connect represents a notable advancement in resolving the communication difficulties encountered by the Faculty of ICT. The project aims to streamline communication channels and improve access to academic and extracurricular information by creating a single platform for information dissemination. The platform can significantly assist students and faculty members in the academic community if well planned and executed.

6.1 Benefits

This topic will elucidate the advantages of the project to both the researcher and the readers.

6.1.1 Benefits to Project Developers

- Improved proficiency in project planning, effective communication, and collaborative teamwork.
- Hands-on experience in developing a valuable tool for academic communities.
- With careful development and execution, the platform has the potential to tremendously benefit both students and faculty members in the academic community.

6.1.2 Benefits to Users

- Enhanced availability of crucial data.
- Increased participation in both academic and extracurricular activities.
- Efficient communication pathways result in decreased confusion and duplication.

6.2 Problems and Limitations

- Possible technological obstacles may arise throughout the process of developing and implementing a platform.
- Some users exhibit resistance to change.
- Continuous updates and maintenance are necessary to ensure the platform's effectiveness.

6.3 Future Work

- Iterative improvement of the platform guided by user feedback.
- Enhancement of functionalities to accommodate changing requirements.
- Integration with other university systems allows for seamless information interchange.
- Adding notification feature.
- Adding complete app setting page.
- Adding report feature for post reporting.

REFERENCES

1. Faculty of ICT, Mahidol University. Internship / Exchange abroad [Internet]. 2024 [cited 2024 Access Date]; [Description]. Available from:
<https://www.ict.mahidol.ac.th/en/student/internship/internshipexchange-abroad/>

2. Faculty of ICT, Mahidol University. MUICT News Updates [Internet]. 2024 [cited 2024 Access Date]; [Description]. Available from:
<https://www.ict.mahidol.ac.th/news/>

BIOGRAPHIES

NAME

Miss Ramita Deeprom

INSTITUTIONS ATTENDED

Chonkanyanukoon School, 2019:

High School Diploma

Mahidol University, 2023:

Bachelor of Science (ICT)

NAME

Mr. Thitiwut Harnphatcharapanukorn

INSTITUTIONS ATTENDED

Potisarn Pittayakorn School, 2019:

High School Diploma

Mahidol University, 2023:

Bachelor of Science (ICT)

NAME

Mr. Burit Sihabut

INSTITUTIONS ATTENDED

Assumption College, 2019:

High School Diploma

Mahidol University, 2023:

Bachelor of Science (ICT)

NAME

Mr. Pongsakorn Kongkaewrasamee

INSTITUTIONS ATTENDED

Suankularb Wittayalai, 2019:

High School Diploma

Mahidol University, 2023:

Bachelor of Science (ICT)