UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FAKULTI TEKNOLOGI MAKLUMAT DAN KOMUNIKASI

W O R K S H O P 1

R E P O R T

Name : RECA SENG BINTI MOHD FADZIL SENG

Matric Number : B031910187

Course : BITI

Project Title : UNIVERSITY PARCEL MANAGEMENT SYSTEM

Supervisor Name : DR. SOODABEH SAFA

Supervisor Signature :

Evaluator Name :

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Nowadays, the existence of the Internet has introduced most university students and staff members to the use of online shopping platforms. As university students and staff can purchase any items from the online shopping platform in seconds, items that come in the parcel are expected to arrive in a day or two. To accommodate parcel arrival for thousands of students and staff every day, a parcel management system must be introduced to manage the storage space for parcel arrival. The cost to deliver a parcel from the university mail office to other residential colleges or faculty and manpower needed for delivery of hundreds or thousands of parcels must be put into serious consideration too. However, most of the students and staff are still unable to get notified and track parcel arrival efficiently. The use of the current paper system become a bottleneck as it is harder to check the records of parcel delivered or collected by the university mail office. Thus, incorrect deliveries of parcel to the wrong residential colleges or faculty make it difficult to trace the parcel whereabouts.

Therefore, the objectives of this project are to develop and design a parcel arrival notification system, to assess parcel arrival records with ease, and to make recommendations on the current parcel management system to avoid delayed, incorrect, or lost deliveries happen. The scope of this project discussed mainly on the ability to track a parcel location, the ability to get notified for the collection of parcels, and the ability to track how many parcels of the same recipient should be collected from respective residential colleges or faculty upon arrival.

In conclusion, the project's significance is to be able to access parcel information in real-time, to ensure the security of a parcel, and to manage the allocation of a high volume of parcels promptly.

PROBLEM STATEMENT

a. Users (university students and staff) unable to get notified and track parcel arrival efficiently.
b. Paper system makes it harder to check the records of parcel delivered or collected by the university mail office.
c. Incorrect deliveries of parcel to the wrong residential colleges or faculty makes it harder to trace the parcel whereabouts.

BACKGROUND OF PROJECT

The invention of online shopping platforms has causes most university students and staff members have now turn from physical shopping to digital shopping. Since physical shopping stores are located further from most of the university colleges, students and staff members are required to spend more on their fare by taking buses or Grabcar to go shopping. On the other hand, to shop online, students and staff are required to only pay a shipping fee for items purchase that costs around RM5 to RM10, which is only around 20% of the usual car fare. Besides, online shopping platforms can be accessed anywhere at any time as it only requires mobile data to access it through the Internet. Much attractive product offers, vouchers, free shipping discount has encouraged physical shopper to shop online as it has significantly saved time and reduce hassle to ride public transportation or car to purchase necessities. Therefore, it is a must for every university to have a real-time system solution for parcel management that enables students and staff to get notified of their parcel and track the status and transition of their parcel upon arrival.

OBJECTIVES

1. To develop and design a parcel arrival notification system
2. To assess parcel arrival records with ease
3. To make recommendations on the current parcel management system to avoid delayed, incorrect, or lost deliveries happen

SCOPES

1. Module to be developed

This parcel management system that will be developed in this project comprises of

i) The ability to track a parcel location after it arrives at the university mail office. Students or staff can key in the tracking number of their parcel to monitor the transit and current status and the location of a parcel. During transit, the University Mail Office administrator is required to update the time and location of a parcel to the system. This helps to lower the risk of losing a parcel to another residential colleges or faculty. Thus, students and staff will also get to resolve any issues aroused at a fast pace.

ii) The ability to get notified for the collection of the parcel from respective residential colleges or faculty office. A notification message will be sent automatically to students or staff once their parcel is ready to be collected from respective residential colleges or faculty office. This helps to give the students or staff a confirmation as well as peace of mind about where their parcel is currently stored.

iii) The ability to track how many parcels of the same recipient should be collected from respective residential colleges or faculty offices after its arrival. Since parcels delivery by different local courier service differs, both the university mail office and students or staff get to check the records of the parcel to see how many parcels that belong to them are collected on the same day. This can be done by key in their ID number and the list of parcels will be displayed on the system. This helps to avoid confusion in managing parcels and ensure every parcel gets to be delivered to the right location at the same time.

2. Target User

Students and staff

(i) Able to monitor the transit and current status and the location of a parcel by key in the tracking number of their parcel.

(ii) Able to get notified once the parcel is ready to be collected from respective residential colleges or faculty.

(iii) Able to know how many parcels should be collected at the same time by key in the ID number to the system.

University Mail Office's Administrator

(i) Able to always be up-to-date on every parcel movement during transit.

(ii) Able to check the records of parcel information with its recipient ID number faster.

(iii) Able to ensure every parcel is delivered safely to its recipient

(iv) Able to resolve the losing parcel issues quickly

CHAPTER 2

DETAILED DESCRIPTION OF THE PROBLEM

i) Users (university students and staff) unable to get notified and track parcel arrival efficiently

a) No notification to indicate if the parcel has arrived at University Mail Office.

b) No system to track when parcel should be collected from respective residential or faculty.

c) No system to track how many parcels of the same recipient should be collected at the same time from respective residential or faculty.

ii) Paper system makes it harder to check the records of parcel delivered or collected by the university mail office.

a) To collect a parcel, students have to search page by page to look for the date of arrival and tracking number, which is a waste of time.

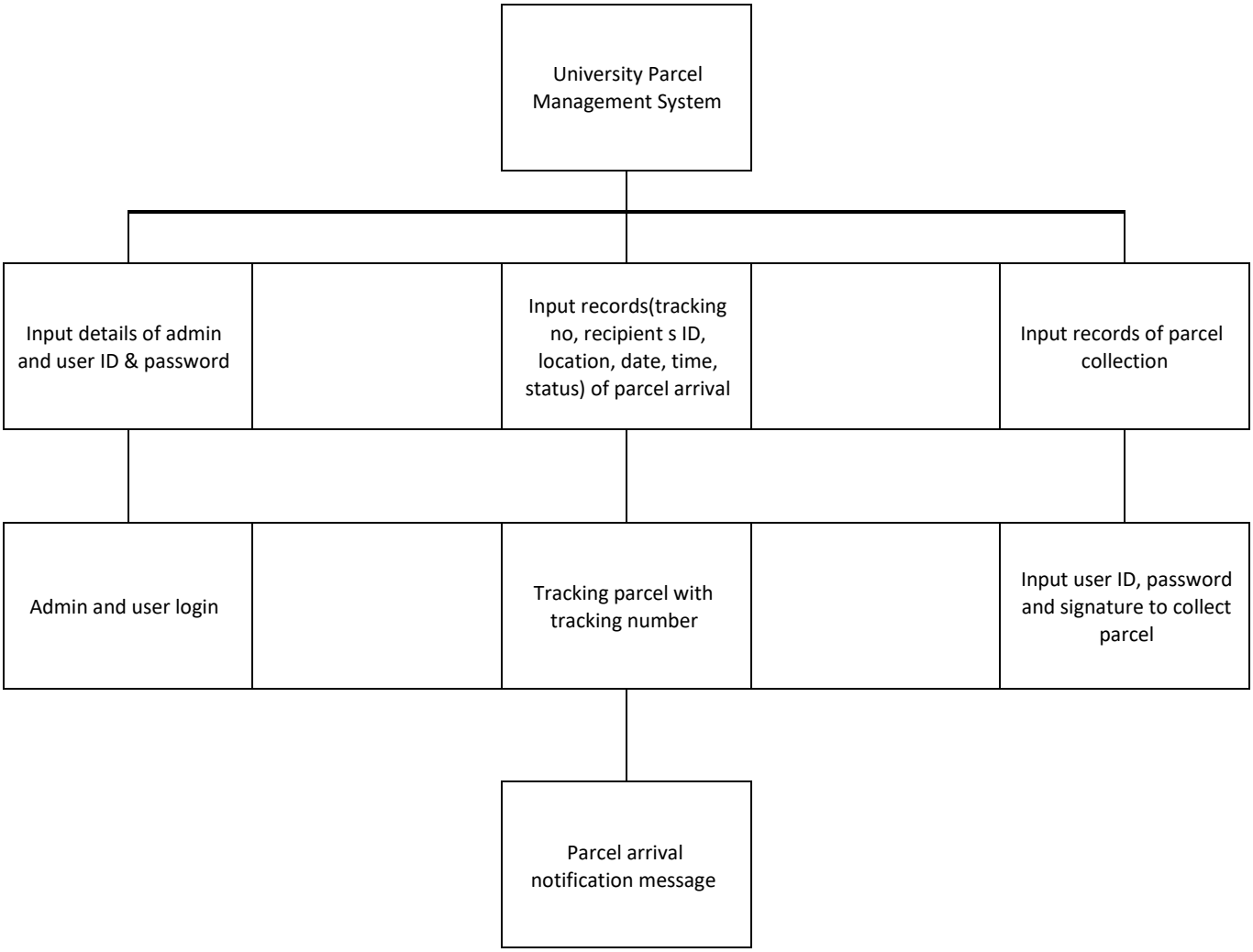
b) If they are a high volume of parcels, students need to wait in a long queue to provide a signature for every parcel which is on several records' pages.

iii) Incorrect deliveries of parcel to the wrong residential colleges or faculty makes it harder to trace the parcel whereabouts.

a) When a parcel has been wrongly sent to another residential or faculty, the students or staffs need to report to the university mail office and awaiting confirmation from their side to know where is the parcel.

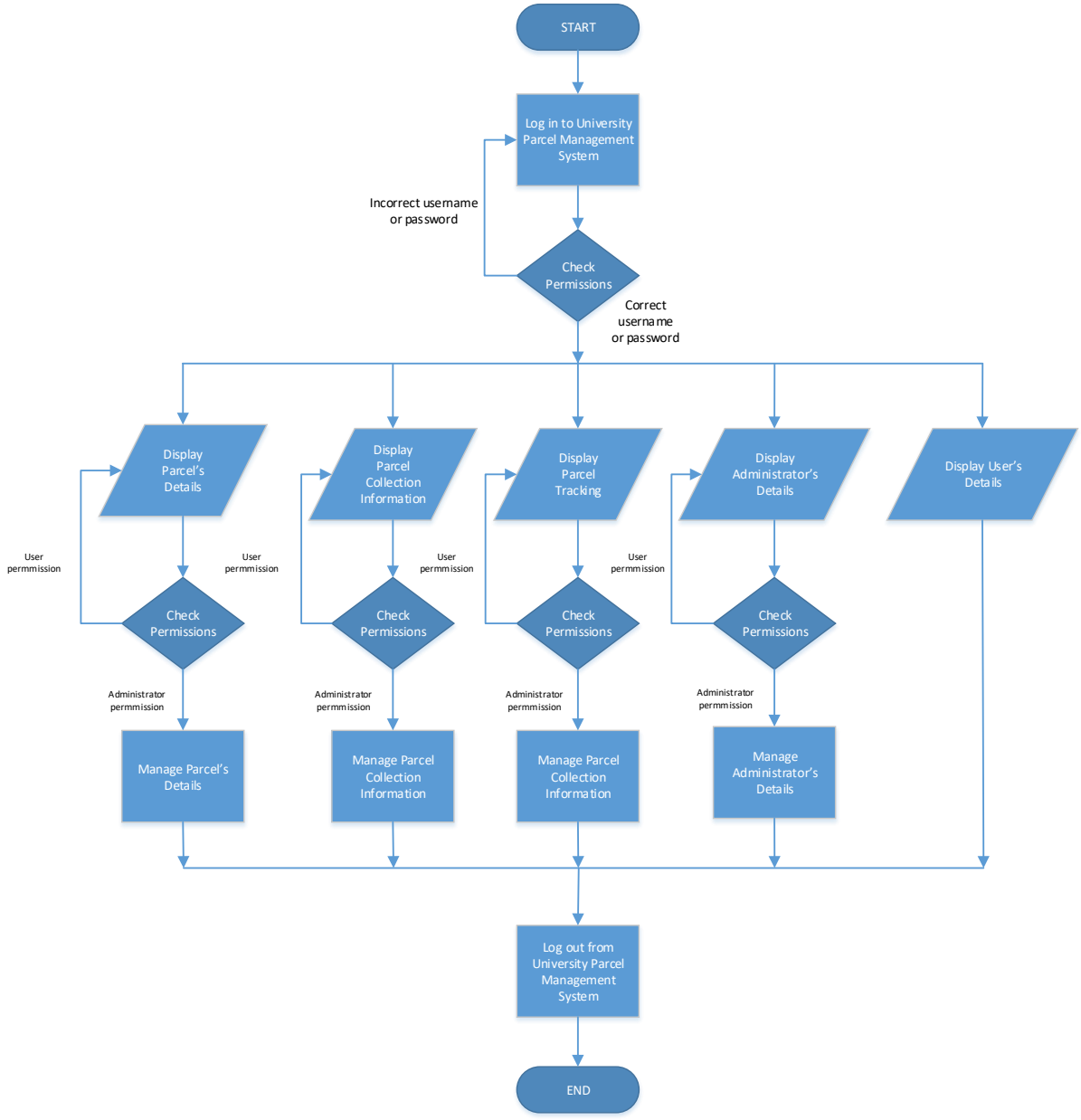
b) If the parcel has been wrongly delivered to further residential colleges or faculty, student or staff are required to spend extra to ride buses or Grabcar to collect the parcel by themselves, which is very costly.

STRUCTURED CHART



CHAPTER 3

FLOWCHART



PSEUDOCODE

(A) Pseudocode for login

1.0 START
2.0 Call **login** function
2.1 Read user_id OR admin_id in string
2.2 Read password in string
2.3 if user_id and user password OR admin_id and admin password valid
        Display system homepage
    else
        Return to login
3.0 END

(B) Pseudocode for parcel details

1.0 START
2.0 Call **parcel** function
2.1 Display parcel details
2.2 if admin_id and admin password valid
    Update parcel details
    else
        print message “Invalid operation.”
        return
3.0 END

(C) Pseudocode for collection details

1.0 START

2.0 Call **collection** function

2.1 Display collection details

2.2 if admin_id and admin password valid

    Update collection details

      else

          print message “Invalid operation.”

          return

3.0 END

(D) Pseudocode for tracking details

1.0 START

2.0 Call **tracking** function

2.1 Display tracking details

2.2 if admin_id and admin password valid

    Update tracking details

      else

          print message “Invalid operation.”

          Return

3.0 END

(E) Pseudocode for administrator details

1.0 START

2.0 Call **administrator** function

2.1 Display administrator details

2.2 if admin_id and admin password valid

Update administrator details

else

print message “Invalid operation.”

Return

3.0 END

(F) Pseudocode for user details

1.0 START

2.0 Call **user** function

2.1 Display user details

2.2 if user_id and user password OR admin_id and admin password valid

Update user details

3.0 END

(G) Pseudocode for system logout

1.0 START

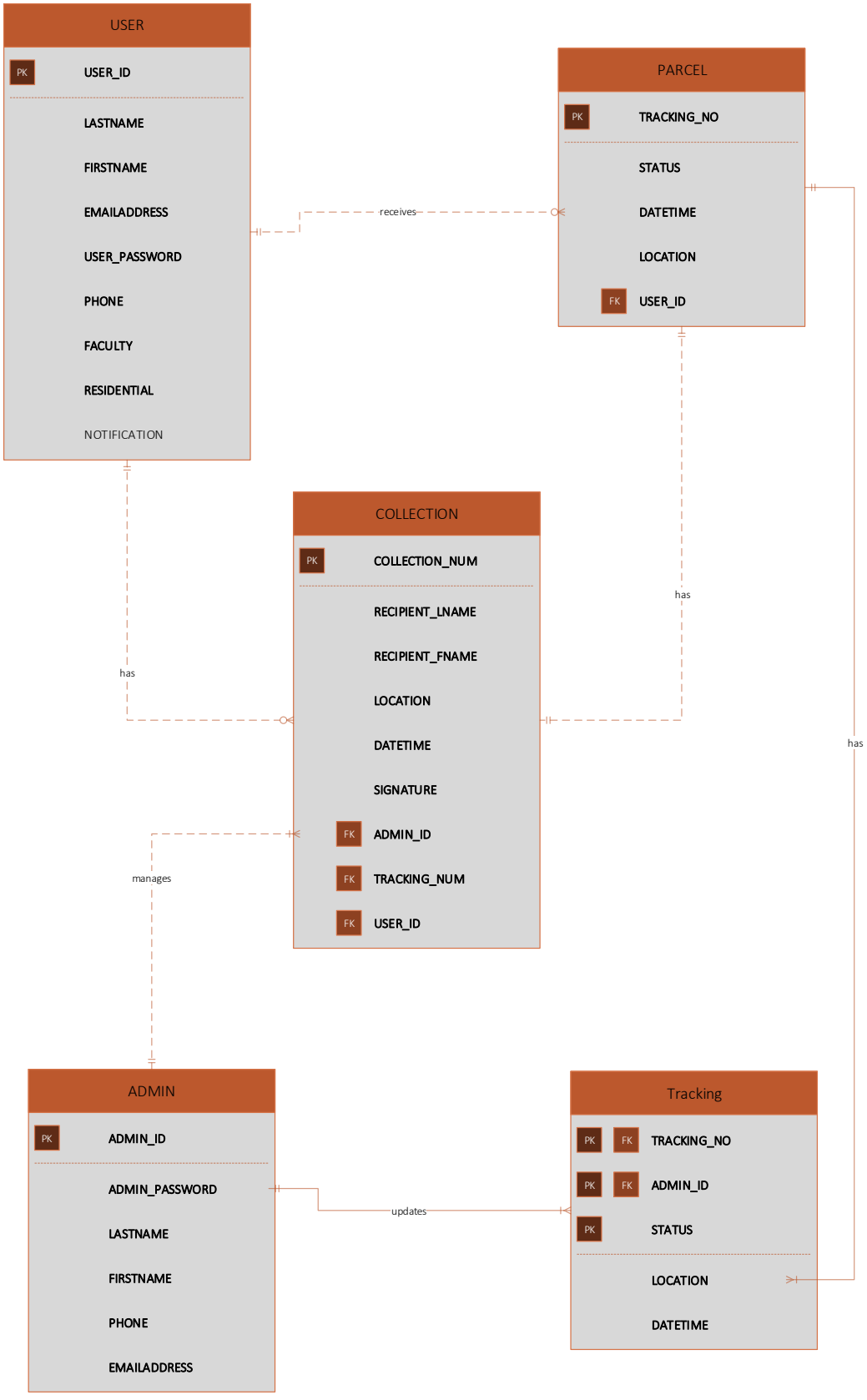
2.0 Call **logout** function

3.0 System logout

4.0 Display system login page

5.0 END

ERD


USER
PK
USER_ID
LASTNAME
FIRSTNAME
EMAILADDRESS
USER_PASSWORD
PHONE
FACULTY
RESIDENTIAL
NOTIFICATION
receives
PARCEL
PK
TRACKING_NO
STATUS
DATETIME
LOCATION
FK
USER_ID
has
has
COLLECTION
PK
COLLECTION_NUM
RECIPIENT_LNAME
RECIPIENT_FNAME
LOCATION
DATETIME
SIGNATURE
FK
ADMIN_ID
FK
TRACKING_NUM
FK
USER_ID
manages
has
ADMIN
PK
ADMIN_ID
ADMIN_PASSWORD
LASTNAME
FIRSTNAME
PHONE
EMAILADDRESS
updates
Tracking
PK
FK
TRACKING_NO
PK
FK
ADMIN_ID
PK
STATUS
LOCATION
DATETIME

DATA DICTIONARY

USER							
ATTRIBUTE NAME	CONTENTS	DATA TYPE AND SIZE	FORMAT	RANGE	REQUIRED?	PK OR FK	REFERENCES
USER_ID	USER'S ID NUMBER	CHAR(25)	xxxxxxxxx		YES	PK	
USER_PASSWORD	USER'S PASSWORD	VARCHAR (50)	xxxxxxxxx		YES		
LASTNAME	USER'S LAST NAME	CHAR(25)	xxxxxxxxx		YES		
FIRSTNAME	USER'S FIRST NAME	CHAR(25)	xxxxxxxxx		YES		
PHONE	USER'S PHONE NUMBER	CHAR (15)	xxxxxxxxx		YES		
EMAILADDRESS	USER'S EMAIL	VARCHAR (100)	xxxxxxxxx		YES		UNIQUE
FACULTY	USER'S FACULTY	CHAR (80)	xxxxxxxxx		YES		
RESIDENTIAL	USER'S RESIDENTIAL	CHAR(50)	xxxxxxxxx		YES		
NOTIFICATION	PARCEL'S NOTIFICATION	VARCHAR (100)	xxxxxxxxx				

ADMINISTRATOR							
ATTRIBUTE NAME	CONTENTS	DATA TYPE AND SIZE	FORMAT	RANGE	REQUIRED?	PK OR FK	REFERENCES
ADMIN_ID	ADMINISTRATOR'S ID NUMBER	CHAR(25)	xxxxxxxxx		YES	PK	
ADMIN_PASSWORD	ADMINISTRATOR'S PASSWORD	VARCHAR(50)	xxxxxxxxx		YES		
FIRSTNAME	ADMINISTRATOR'S FIRST NAME	CHAR(25)	xxxxxxxxx		YES		
LASTNAME	ADMINISTRATOR'S LAST NAME	CHAR(25)	xxxxxxxxx		YES		
PHONE	ADMINISTRATOR'S PHONE NUMBER	CHAR (15)	xxxxxxxxx		YES		
EMAILADDRESS	ADMINISTRATOR'S EMAIL	VARCHAR(100)	xxxxxxxxx		YES		UNIQUE

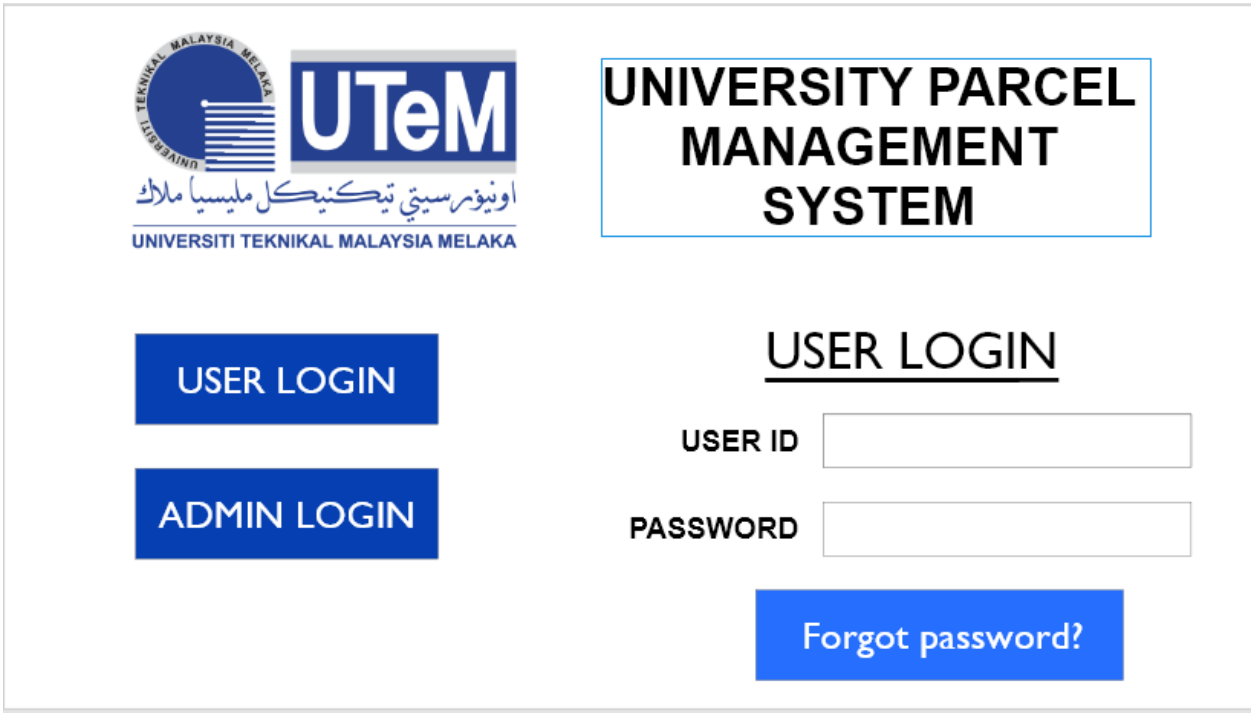
PARCEL							
ATTRIBUTE NAME	CONTENTS	DATA TYPE AND SIZE	FORMAT	RANGE	REQUIRED?	PK OR FK	REFERENCES
TRACKING_NO	PARCEL'S TRACKING NUMBER	CHAR(35)	xxxxxxxxxx		YES	PK	
STATUS	PARCEL'S STATUS	VARCHAR(100)	xxxxxxxxxx		YES		
LOCATION	LOCATION OF PARCEL	VARCHAR(100)	xxxxxxxxxx		YES		
DATETIME	DATE AND TIME OF PARCEL ARRIVAL	TIMESTAMP	YYYY-MM-DD hh:mm:ss		YES		
USER'S ID	USER'S ID NUMBER	CHAR(25)	xxxxxxxxxx		YES	FK	USER

TRACKING							
ATTRIBUTE NAME	CONTENTS	DATA TYPE AND SIZE	FORMAT	RANGE	REQUIRED?	PK OR FK	REFERENCES
TRACKING_NO	PARCEL'S TRACKING NUMBER	CHAR(35)	xxxxxxxxxx		YES	PK & FK	PARCEL
ADMIN_ID	ADMINISTRATOR'S ID NUMBER	CHAR(25)	xxxxxxxxxx		YES	PK & FK	ADMINSTRATOR
STATUS	TRACKING STATUS	VARCHAR(100)	xxxxxxxxxx		YES	PK	
LOCATION	TRACKING LOCATION	VARCHAR(100)	xxxxxxxxxx		YES		
DATETIME	TRACKING DATE AND TIME	TIME	YYYY-MM-DD hh:mm:ss		YES		

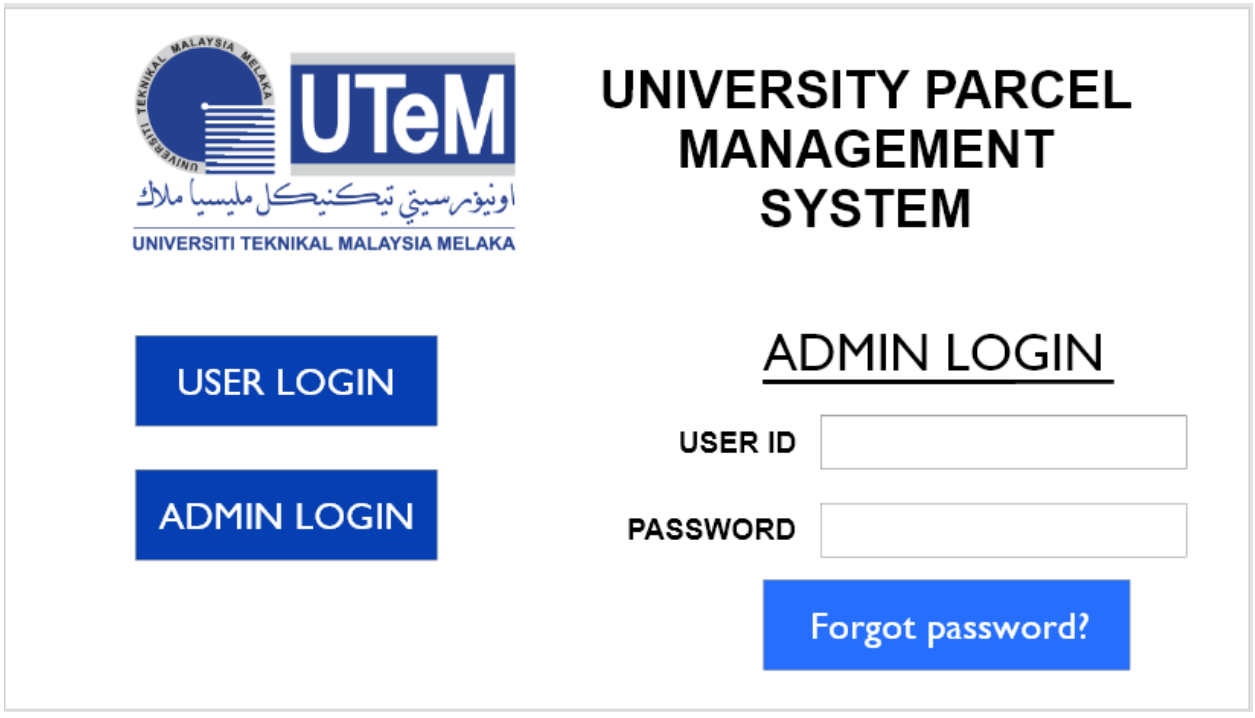
COLLECTION							
ATTRIBUTE NAME	CONTENTS	DATA TYPE AND SIZE	FORMAT	RANGE	REQUIRED?	PK OR FK	REFERENCES
COLLECTION_NUM	COLLECTION NUMBER	INTEGER	xxxxxxxxx		YES	PK	
RECIPIENT_FNAME	RECIPIENT'S FIRST NAME	CHAR(25)	xxxxxxxxx		YES		
RECIPIENT_LNAME	RECIPIENT'S LAST NAME	CHAR(25)	xxxxxxxxx		YES		
COLLECT_LOC	LOCATION OF COLLECTION	VARCHAR2(40)	xxxxxxxxx		YES		
DATETIME	COLLECTION DATE AND TIME	TIMESTAMP	YYYY-MM-DD hh:mm:ss		YES		
SIGNATURE	SIGNATURE (IC NUMBER)	CHAR(25)	xxxxxxxxx		YES		
TRACKING_NO	PARCEL'S TRACKING NUMBER	CHAR(35)	xxxxxxxxx		YES	FK	PARCEL
USER_ID	USER'S ID NUMBER	CHAR(25)	xxxxxxxxx		YES	FK	USER
ADMIN_ID	ADMINISTRATOR'S ID NUMBER	CHAR(25)	xxxxxxxxx		YES	PK	ADMINSTRATOR

INTERFACE DESIGN

LOGIN PAGE (AS USER)

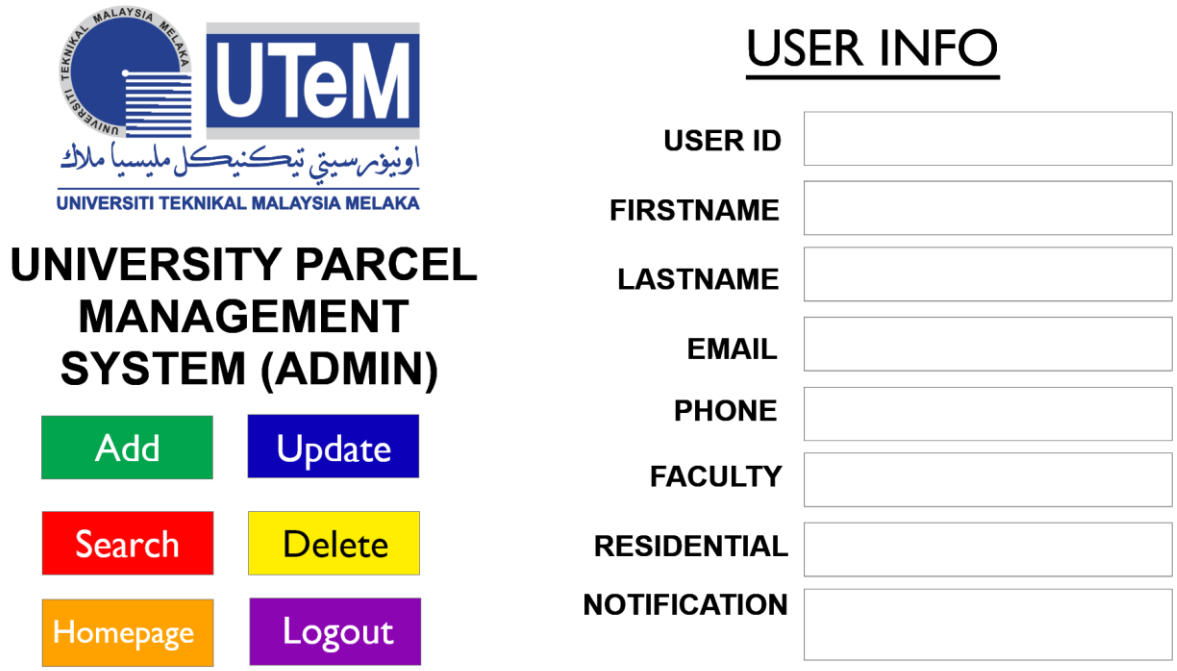


LOGIN PAGE (AS ADMIN)



USER INFO PAGE (AS USER)

USER INFO PAGE (AS ADMIN)



CHAPTER 4

IMPLEMENTATION

Programming Technique:

1. Multiple Way Selection Control Structure is used throughout the coding.

```
system.userDisplayUser(id);
cout << "\nWhich page do you wish to proceed?" << endl;
cout << "1. Display User Information\n2. User Homepage\n3. Logout\nPage: ";
cin >> userPage;
if (userPage == 1)
{
    userOption = 1;
}
else if (userPage == 2)
{
    firstletter = 'B';
    userOption = 0;
}
else if (userPage == 3)
    return 0;
else
{
    cout << "Invalid input!" << endl;
    cout << "\nWhich page do you wish to proceed?" << endl;
    cout << "1. Display User Information\n2. User Homepage\n3. Logout\nPage: ";
    cin >> userPage;
```

Back to Display User Information Page

Back to User Homepage

Logout from the system

If user enter an invalid input, user is required to enter once again

After calling userDisplayUser function, if…else if statement,is used to enable user to choose which page to proceed next.

```
system.displayTracking();
cout << "\nWhich page do you wish to proceed?" << endl;
cout << "1. Tracking Info Page\n2. Administrator Homepage\n3. Logout\nPage: ";
cin >> adminPage;
if (adminPage == 1)
{
    data = 4;
    adminOption = 0;
    adminPage = 0;
}
else if (adminPage == 2)
{
    data = 0; adminOption = 0; adminPage = 0;
    firstletter = 'A';
}
else if (adminPage == 3)
{
    return 0;
}
else
{
    cout << "Invalid input!" << endl;
    cout << "\nWhich page do you wish to proceed?" << endl;
    cout << "1. Tracking Info Page\n2. Administrator Homepage\n3. Logout\nPage: ";
    cin >> adminPage;
}
```

Back to Tracking Info Page

Back to Administrator Homepage

Logout from the system

If admin enter an invalid input, admin is required to enter once again

The same goes to Admin. After calling displayTracking function, if…else if statement,is used to enable admin to choose which page to proceed next.

2. While loop(repetition) is used throughout the coding.

```
while (firstletter != 'A' && firstletter != 'a' && firstletter != 'B' && firstletter != 'b')
{
    id = loginManager.login();
}
```

(expression: firstletter is not equal to ‘A’ AND firstletter is not equal to ‘a’ A’ AND firstletter is not equal to ‘B’ AND firstletter is not equal to ‘b’ is evaluated)

 if true, then statement is executed, and expression is evaluated again which in this case admin or user are required to reenter their id and password again

 if false, then the loop is finished and program statements following statement execute

```
cout << endl << "*****Add New Parcel Information*****" << endl;
cout << "Please enter the new parcel details" << endl
    << "Tracking number:\t";
cin.get();
getline(cin, parcel->tracking_no);
while (parcel->tracking_no.empty())
{
    cout << endl << "Tracking number is a mandatory information" << endl;
    cout << "Tracking number:\t";
    getline(cin, parcel->tracking_no);
}

cout << "Status(Arrived University Mail Office/ Ready for collection/ Collected)\t";
getline(cin, parcel->status);
while (parcel->status.empty())
{
    cout << endl << "Status is a mandatory information" << endl;
    cout << "Status(Arrived University Mail Office/ Ready for collection/ Collected):\t";
    getline(cin, parcel->status);
}
```

(expression: admin did not enter tracking no. or left tracking no. empty for new parcel details)
 if true, admin is required to enter tracking no because it is mandatory
 if false, admin is required to enter the next detail which is the status of the parcel.

3. Pointer is used throughout the coding.

```
Parcel* ParcelManager::selectParcel(string tracking_no)
{
    Parcel* parcel = NULL;
    DatabaseConnection dbConn;
    PreparedStatement* ps = dbConn.prepareStatement("SELECT * FROM Parcel WHERE tracking_no = ?");

    ps->setString(1, tracking_no);

    ResultSet* rs = ps->executeQuery();

    if (rs->next())
    {
        parcel = new Parcel();

        parcel->tracking_no = rs->getString(1);
        parcel->status = rs->getString(2);
        parcel->datetime = rs->getString(3);
        parcel->location = rs->getString(4);
        parcel->userID = rs->getString(5);
        cout << "\n The parcel info: " << endl;
        cout << "Status: " << parcel->status << "." << endl;
        cout << "Date & Time: " << parcel->datetime << "." << endl;
        cout << "Location: " << parcel->location << "." << endl;
        cout << "User ID: " << parcel->userID << "." << endl;
    }
    else
    {
        cout << "Invalid tracking number!" << endl;
    }

    delete rs;
    delete ps;

    return parcel;
}
```

Pointer **parcel** is created and it is now pointing to NULL before having an actual value inside it.

Pointer **ps** is created and it is now pointing to SQL statement and then the **ps** set '?' sign in SQL statement to tracking no

Pointer **rs** is created and it is pointing to the result that pointer **ps** executed.

Pointer **parcel** will then take the result that pointer **rs** carry from the database one by one.

As pointer **ps** and **rs** have finished done its part in the selectParcel function, they are now deleted.

Pointer **parcel** that carry the details of the parcel is now returned.

Pointers are used a lot in the coding especially retrieving data from the database. In the above example, pointer is used to retrieve information and details from the PARCEL table.

The Outcome of the System:

The demonstration of the system is recorded in video format and has been uploaded to Youtube.

1. The system is appropriate to add, update, select(search) and delete from database. (Logic Programming & database)

   Refer to this link: https://youtu.be/kX2x2SBPynA

2. The system has a login function. The user or admin are required to login into the system using their ID and password.

   Refer to this link: https://youtu.be/S5eDVJuSm9M

3. Admin is able to add, update, delete and search for every information in the database while user can only search for the information in each table and update his own user information. (Restrict user and admin permission)

   Refer to this link: https://youtu.be/S5eDVJuSm9M

4. Demonstration of error handling

   Refer to this link: https://youtu.be/S5eDVJuSm9M

5. Demonstration in making user and admin password case-sensitive and also accepting mandatory info whenever a new row is added to a table.

   Refer to this video link: https://youtu.be/xnbLIrOD6nA

6. The number of parcel and tracking info about a parcel is counted for each user. (Complexities of Calculation)

   Refer to this video link: https://youtu.be/5843cdqD7jY

7. By adding a new parcel, tracking info will be automatically generated. Besides, user will automatically get a notification in the user info page. (Report Generation & Analysis)

   Refer to this video link: https://youtu.be/5843cdqD7jY

8. By updating a parcel status, tracking info will be automatically generated. Besides, user will get an updated notification about the latest parcel status at the user info page. (Report Generation & Analysis)

   Refer to this video link: https://youtu.be/5843cdqD7jY

9. Forgot password function is available to user and admin. (Report Generation & Analysis)

   Refer to this video link: https://youtu.be/5843cdqD7jY

CHAPTER 5

CONCLUSION

First and foremost, the problem of not being able to get notified and track parcel arrival efficiently has been solved by including a notification column in the user table. When admin inserts a new parcel into the system, tracking info of the parcel will be created automatically. Thus, user notification column will also automatically update to the latest information which includes the parcel tracking no., parcel status and date time. This feature has not only help users to get the latest information about their parcel through the notification, it has greatly increase the efficiency for admin in updating the information of parcel, tracking and user notification at once without the hassle of inserting manually to each of these tables.

Secondly, the problem of paper system makes it harder to check the records of parcel delivered or collected by the university mail office has been solved by having a database system to record every insert, update, search and delete data operation. For example, when admin inserts a new user info into the system, a new row will be added to the user table in database system. This feature has helped in avoiding the issue of missing data or records while managing the parcel.

Thirdly, the issue of incorrect deliveries of parcel to the wrong residential colleges or faculty makes it harder to trace the parcel whereabouts has been solved by having the admin ID on every parcel tracking table. Through having admin ID for every parcel tracking info, user is able to refer to the admin contact no to search for their parcel and get to know its whereabouts. This feature has helped to avoid confusion between user and admin while tracing the parcel.

In conclusion, the demonstration of all features of the system has achieved all of the objectives of this project which are to develop and design a parcel arrival notification system, to assess parcel arrival records with ease and to make recommendations on the current parcel management system to avoid delayed, incorrect, or lost deliveries happen. However, this system has a few of future improvements to make. The first future improvement of the system is there should be a button for every option or choice displayed in the system instead of entering it manually. The second future improvement of the system is the result of every information should display in table forms to make it easier for admin and user to look for particular row of data in a table. The third future improvement of the system

is every info page can accept every column attribute in the table to search for information, not only primary key or foreign key.