GROUP PROJECT COVER PAGE

COURSE CODE   : SECD 2523

COURSE NAME   : DATABASE

SECTION   : 06

LECTURER'S NAME   : DR IZYAN IZZATI BINTI KAMSANI

SESSION/SEMESTER   : 2023-2024/1

ASSIGNMENT TITLE   : Phase 3 (P3) – Database Logical Design & SQL

GROUP NUMBER   : 02

MEMBER DETAILS   :

NAME	MATRIC NUMBER
TAM JIA HAO	**A22EC0106**
PUA ZHI YING	**A22EC0103**
TAN YOU CHUN	**A22EC0108**
TRINAATH A/L ATHINARAYANARAO	**A22EC0289**
PUJJAA JANANI A/P SEGAR	**A22EC0260**

Table of Contents

1.0 Introduction

Based on the case study, MBIP has faced some challenges in their current way of collecting data for the Iskandar Puteri Calendar Competition. To address these issues, we have proposed a design that has a number of features that provides MBIP an efficient and reliable platform. In this project, MBIP is the user of the current system, IPRK System, which is a platform that can help MBIP to address the challenges through the integration of efficient carbon footprint calculations, registration and login processes, and monitoring dashboards. Emphasizing inclusivity, the system aims to provide user-friendly interfaces to encourage participation from individuals of all age groups. During the business process, MBIP faced some difficulties while using the current system. We wish to upgrade the current system to help MBIP for integration of efficient carbon footprint calculations, registration and login processes, and monitoring dashboards so that the system can run smoothly and thrive over time. We will provide our plans for the development of the current system, which is unsatisfactory from our standpoint as a client or owner. Thus, we are assigned to design a database system that can improve the system productivity. In this phrase, we will update the business rule and data dictionary based on previous phrases to make sure the IPRK system is more suitable for MBIP. This phase is more focused about database logical design to use in to be system based on the data model.

2.0 Overview of project

In this phase, we will describe some of what we got in the previous phase, such as updating business rules, conceptual ERD, enhanced ERD, logical enhanced ERD and updating some of them at once. The final logical ERD must be created to achieve the normalization process to the Boyce-Codd Normal Form (BCNF) relation. After completing phase 2 which is the database conceptual design phase, we have proceeded to phase 3 to implement the database logical design and produce SQL statements (both DDL and DML) for our IPRK System. In this paper, we will transform the conceptual ERD in phase 2 into logical ERD and followed by producing the relation schema. Normalization will also be done to minimize the duplicated data so that it can enhance the data stored logically. After that, we will update the data dictionary for the IPRK System website by referring to the normalized relation. Lastly, the database for the website will be created using SQL statements to transform the requirements into complete and detailed system design specifications.

3.0 Database conceptual design

3.1 Updated business rule

- Each community user can submit one or more recycle details.
- Each recycle detail can be submitted by one community user.

- Each community user can submit one or more consumptions.
- Each consumption can be submitted by one community user.
- Each community user can have one or more monthly footprints.
- Each monthly footprint can be owned by one community user.

- Each community user can have one total footprint.
- Each total footprint can be owned by one community user.

- Community user can submit their personal information for registration
- Community user can submit their userID and password for login

- Community user can submit their consumption for each month
- Community user can submit their recycle details for each month

- Community user can view their consumption and recycle details for each month
- Community user can view their overall monthly total footprint and monthly water, electric and recycle footprints.
- Community user can view their overall total footprint and total water, electric and recycle footprints.

- Community user can view their full name and userID

- MBIP_Admin can view the lowest total footprint and the related community user's userID
- MBIP_Admin can submit their userID and password for login

- MBIP_Admin can view all community user's total footprint combined as well as total water, electric and recycle footprints combined.
- MBIP_Admin can view all community user's total monthly footprint combined as well as total monthly water, electric and recycle footprints combined.
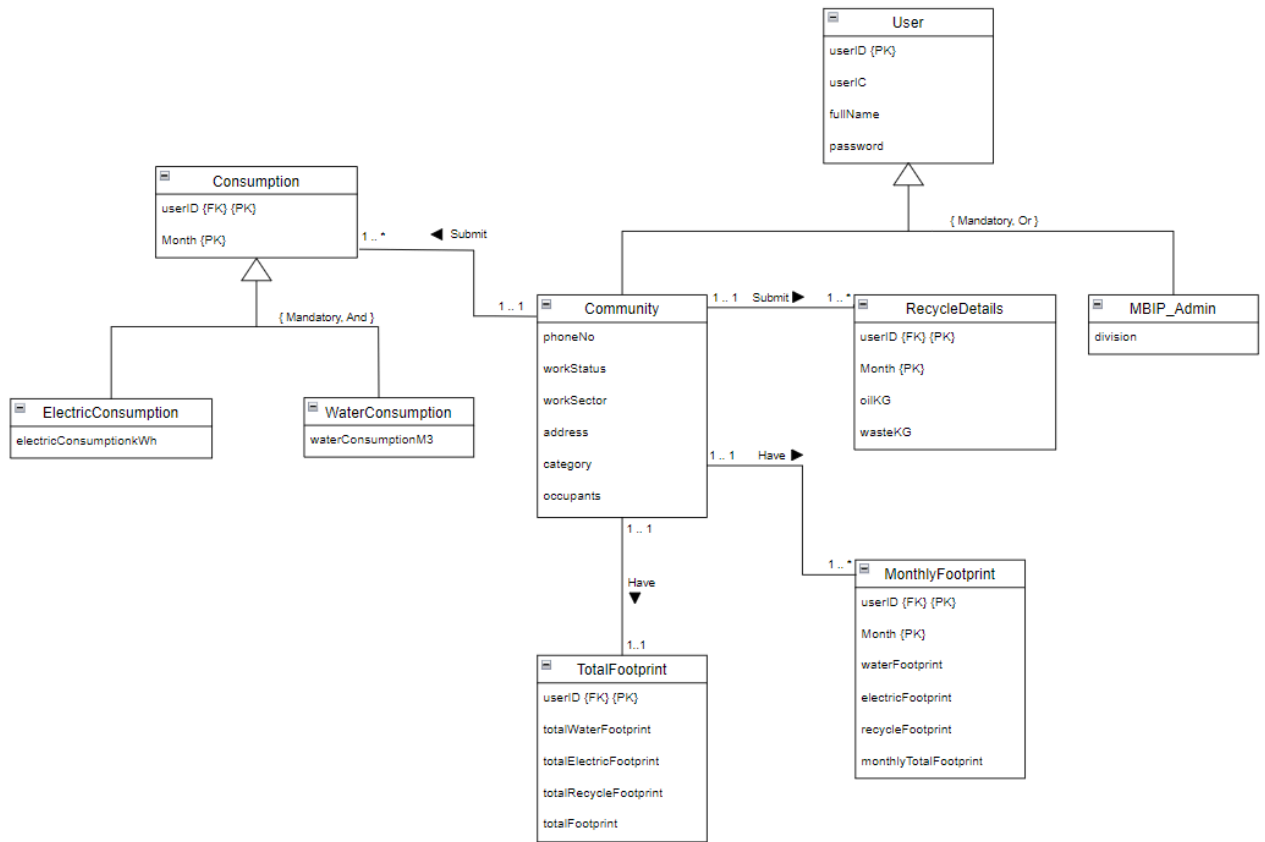- MBIP_Admin can view their full name, userID and division

3.2 Conceptual ERD

Figure 1: Conceptual ERD
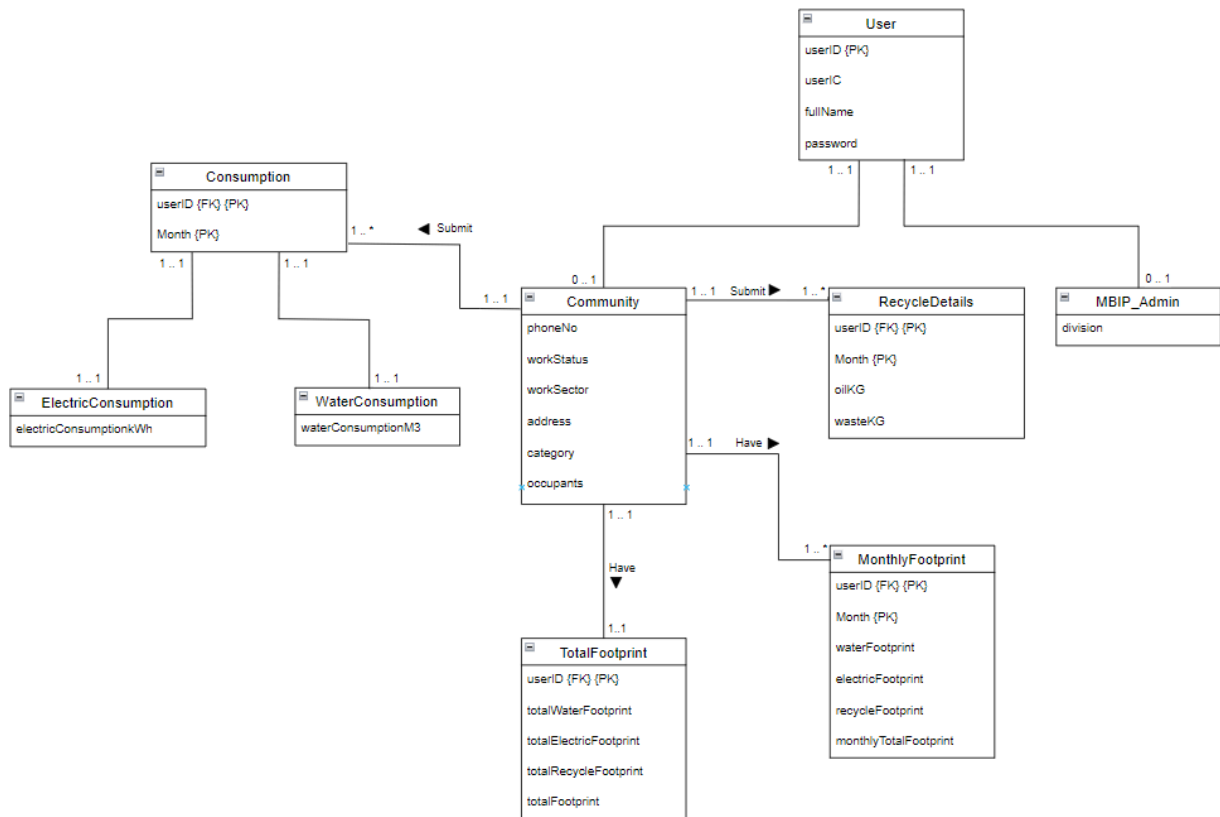
4.0 DB logical design

4.1 Logical ERD

Figure 2: Logical ERD

Relation Schemas

- ElectricWaterConsumption (<u>userID, Month</u>, electricConsumptionkWh, waterConsumptionM3)
     PK: userID, Month
     FK: userID references CommunityUser_TotalFoot (userID)

- RecycleDetails (<u>userID, Month</u>, oilKG, wasteKG)
     PK: userID, Month
     FK: userID references CommunityUser_TotalFoot (userID)

- MonthlyFootprint (<u>userID, Month</u>, waterFootprint, electricFootprint, recycleFootprint, monthlyTotalFootprint)
     PK: userID, Month
     FK: userID references CommunityUser_TotalFoot (userID)

- CommunityUser_TotalFoot (<u>userID</u>, userIC, fullName, password, phoneNo, workStatus, workSector, address, category, occupants, totalWaterFootprint, totalElectricFootprint, totalRecycleFootprint, totalFootprint)

PK: userID

- MBIP_AdminUser (<u>userID</u>, userIC, fullName, password, division)
    PK: userID

4.2 Updated Data Dictionary

4.2.1 Description of Entity

Entity	Description	Occurrence
ElectricWaterConsumption	Hold the data of electric and water consumption	Record the electric and water consumption.
RecycleDetails	Hold the data of recycle details	Record the total amount of waste and oil.
MonthlyFootprint	Hold the data of monthly footprint	Record the monthly footprint.
CommunityUser_TotalFoot	Hold the data of the user and total footprint	Users fill in their information and total footprint.
MBIP_AdminUser	Hold the data of MBIP admin user	MBIP admin users record their information.

4.2.2 Description of Relationship

Entity	Multiplicity	Relationship	Multiplicity	Entity
Community	1..1	Submit	1..*	consumption
	1..1	Submit	1..*	RecycleDetails
	1..1	Have	1..1	TotalFootprint
	1..1	Have	1..*	MonthlyFootprint
Consumption	1..1		1..1	ElectricConsumption
	1..1		1..1	WaterConsumption
User	1..1		0..1	Community
	1..1		0..1	MBIP_Admin

4.2.1 Description of Attributes

Entity	Attribute	Description	Data Type	Constraint
ElectricWaterConsumption	userID	Uniquely identified user	VARCHAR2(10)	FOREIGN KEY REFERENCES: CommunityUser_TotalFoot
	Month	Month for consumption	VARCHAR2(10)	PRIMARY KEY
	electricConsumptionkWh	Total electric consumption	NUMBER(5)	NOT NULL
	waterConsumptionM3	Total water consumption	NUMBER(10,2)	NOT NULL
RecycleDetails	userID	Uniquely identified user	VARCHAR2(10)	FOREIGN KEY REFERENCES: CommunityUser_TotalFoot
	Month	Month for consumption	VARCHAR2(10)	PRIMARY KEY
	oilKG	Total oil	NUMBER(5)	NOT NULL
	wasteKG	Total waste	NUMBER(5)	NOT NULL
MonthlyFootprint	userID	Uniquely identified user	VARCHAR2(10)	FOREIGN KEY REFERENCES: CommunityUser_TotalFoot
	Month	Month for consumption	VARCHAR2(10)	PRIMARY KEY
	waterFootprint	Water footprint	NUMBER(10)	NOT NULL
	electricFootprint	Electric footprint	NUMBER(10)	NOT NULL
	recycleFootprint	Recycle footprint	NUMBER(10)	NOT NULL
	monthlyTotalFootprint	Total footprint in a month	NUMBER(38)	NOT NULL
CommunityUser_TotalFoot	userID	Uniquely identified user	VARCHAR2(10)	PRIMARY KEY
	userIC	User ic number	VARCHAR2(14)	NOT NULL
	fullName	User name	VARCHAR2(50)	NOT NULL

	password	User password	VARCHAR2(20)	NOT NULL
	phoneNo	Community phone number	VARCHAR2(15)	NOT NULL
	workStatus	Work status for community	VARCHAR2(10)	NOT NULL
	workSector	Work sector for community	VARCHAR2(20)	NOT NULL
	address	Address for community	VARCHAR2(150)	NOT NULL
	category	Category for community	VARCHAR2(10)	NOT NULL
	occupants	Occupants of community	NUMBER(2)	NOT NULL
	totalWaterFootprint	Total water footprint	NUMBER(10)	
	totalElectricFootprint	Total electric footprint	NUMBER(10)	
	totalRecycleFootprint	Total recycle footprint	NUMBER(10)	
	totalFootprint	Total footprint	NUMBER(38)	
MBIP_Admin User	userID	Uniquely identified user	VARCHAR2(10)	PRIMARY KEY
	userIC	User ic number	VARCHAR2(14)	NOT NULL
	fullName	User name	VARCHAR2(50)	NOT NULL
	password	User password	VARCHAR2(20)	NOT NULL
	division	MBIP admin division	VARCHAR2(50)	NOT NULL

4.3 Normalization

1. **ElectricWaterConsumption**
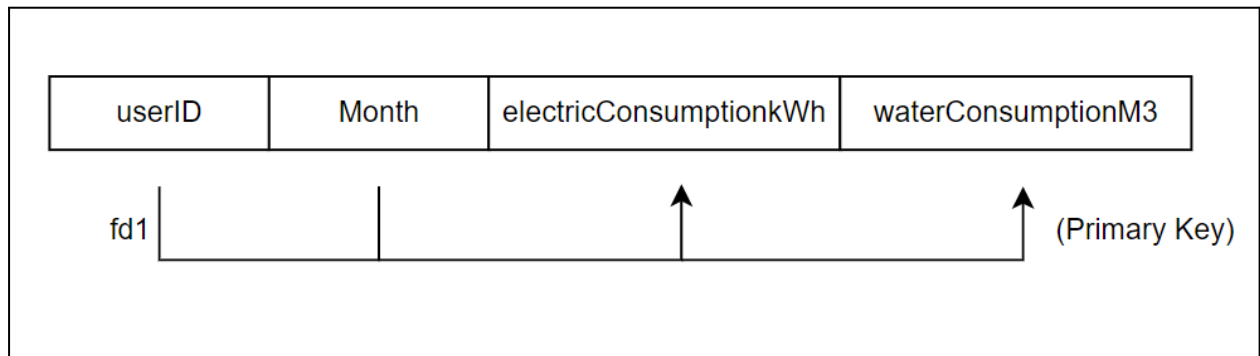
1NF

ElectricWaterConsumption(<u>userID, Month</u>, electricConsumptionkWh, waterConsumptionM3)
      PK: userID, Month
      FK: userID references CommunityUser_TotalFoot (userID)

2NF

ElectricWaterConsumption(<u>userID, Month</u>, electricConsumptionkWh, waterConsumptionM3)
      PK: userID, Month
      FK: userID references CommunityUser_TotalFoot (userID)

3NF

ElectricWaterConsumption(<u>userID, Month</u>, electricConsumptionkWh, waterConsumptionM3)
      PK: userID, Month
      FK: userID references CommunityUser_TotalFoot (userID)

BCNF

ElectricWaterConsumption(<u>userID, Month</u>, electricConsumptionkWh, waterConsumptionM3)
      PK: userID, Month
      FK: userID references CommunityUser_TotalFoot (userID)

2. RecycleDetails
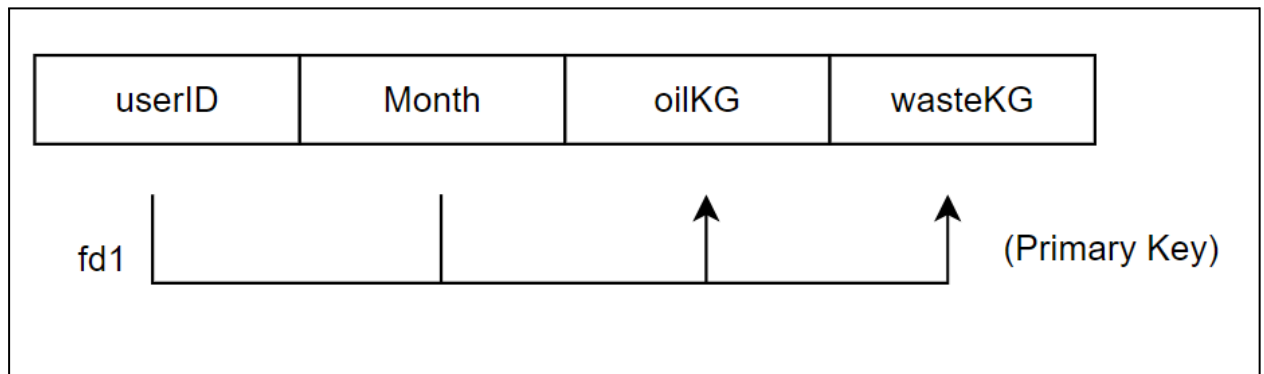
1NF

RecycleDetails (<u>userID, Month</u>, oilKG, wasteKG)

      PK: userID, Month

      FK: userID references CommunityUser_TotalFoot (userID)

2NF

RecycleDetails (<u>userID, Month</u>, oilKG, wasteKG)

      PK: userID, Month

      FK: userID references CommunityUser_TotalFoot (userID)

3NF

RecycleDetails (<u>userID, Month</u>, oilKG, wasteKG)

      PK: userID, Month

      FK: userID references CommunityUser_TotalFoot (userID)

BCNF

RecycleDetails (<u>userID, Month</u>, oilKG, wasteKG)

      PK: userID, Month

      FK: userID references CommunityUser_TotalFoot (userID)

3. MonthlyFootprint

userID	Month	waterFootprint	electricFootprint	recycleFootprint	monthlyTotalFootprint

fd1   (Primary Key)
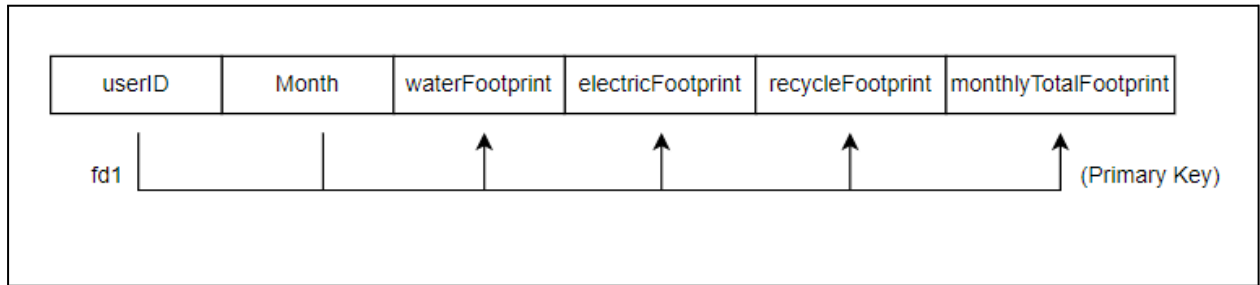
1NF
MonthlyFootprint (<u>userID</u>, <u>Month</u>, waterFootprint, electricFootprint, recycleFootprint, monthlyTotalFootprint)

      PK: userID, Month

      FK: userID references CommunityUser_TotalFoot (userID)

2NF
MonthlyFootprint (<u>userID</u>, <u>Month</u>, waterFootprint, electricFootprint, recycleFootprint, monthlyTotalFootprint)

      PK: userID, Month

      FK: userID references CommunityUser_TotalFoot (userID)

3NF
MonthlyFootprint (<u>userID</u>, <u>Month</u>, waterFootprint, electricFootprint, recycleFootprint, monthlyTotalFootprint)

      PK: userID, Month

      FK: userID references CommunityUser_TotalFoot (userID)

BCNF
MonthlyFootprint (<u>userID</u>, <u>Month</u>, waterFootprint, electricFootprint, recycleFootprint, monthlyTotalFootprint)
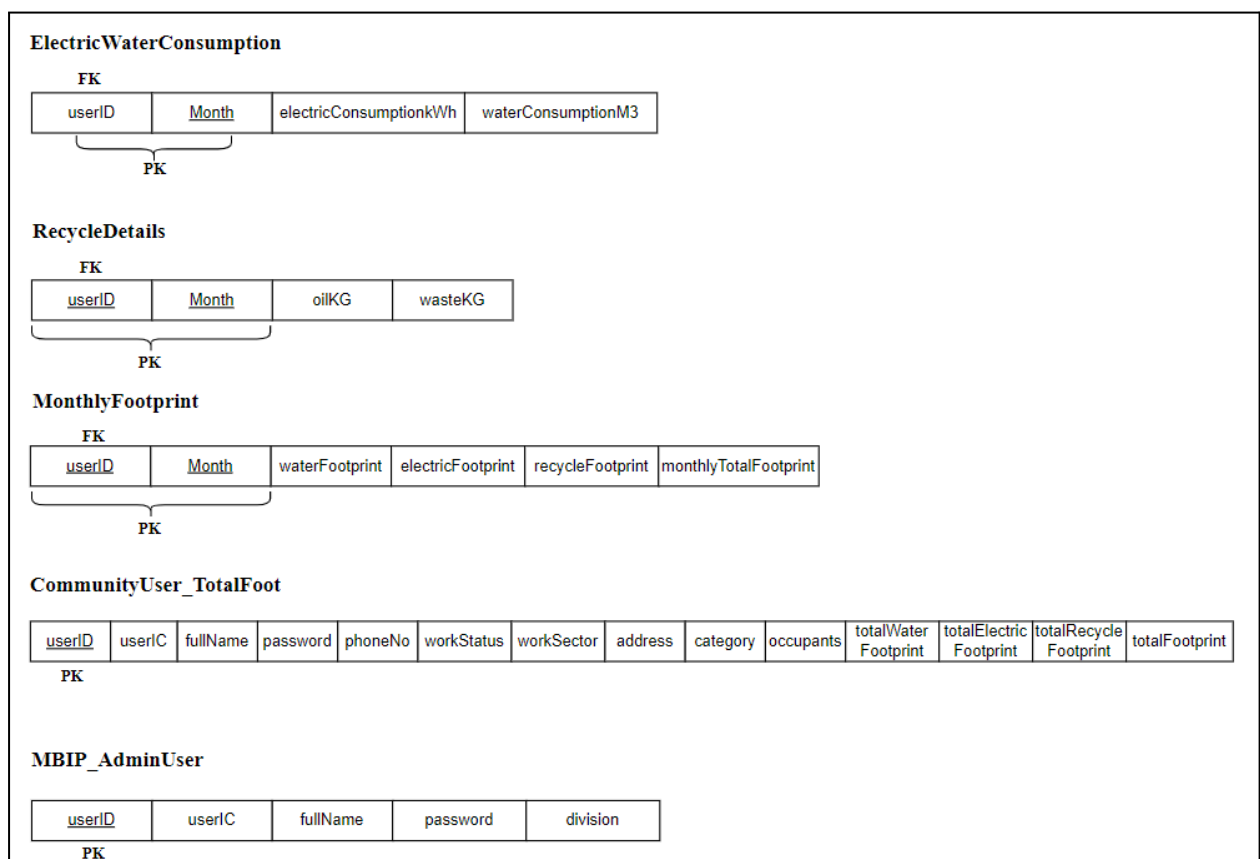
      PK: userID, Month

      FK: userID references CommunityUser_TotalFoot (userID)

4. CommunityUser_TotalFoot

userID	userIC	fullName	password	phoneNo	workStatus	workSector	address	category	occupants	totalWater Footprint	totalElectric Footprint	totalRecycle Footprint	totalFootprint

fd1 ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ (Primary Key)

fd2 ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ (Candidate Key)
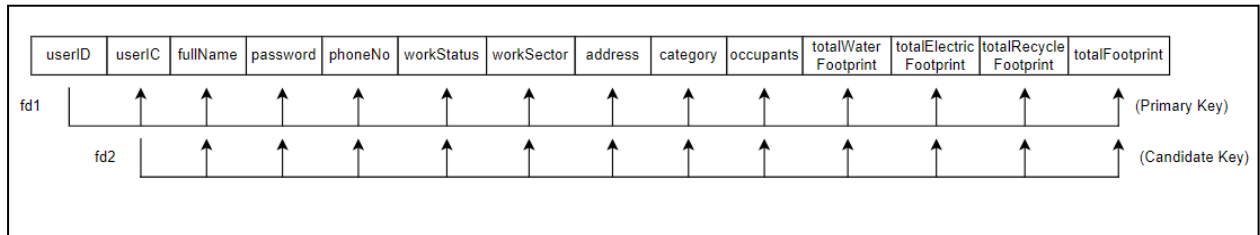
1NF

CommunityUser_TotalFoot (<u>userID</u>, userIC, fullName, password, phoneNo, workStatus, workSector, address, category, occupants, totalWaterFootprint, totalElectricFootprint, totalRecycleFootprint, totalFootprint)

      PK: userID

2NF

CommunityUser_TotalFoot (<u>userID</u>, userIC, fullName, password, phoneNo, workStatus, workSector, address, category, occupants, totalWaterFootprint, totalElectricFootprint, totalRecycleFootprint, totalFootprint)

      PK: userID

3NF

CommunityUser_TotalFoot (<u>userID</u>, userIC, fullName, password, phoneNo, workStatus, workSector, address, category, occupants, totalWaterFootprint, totalElectricFootprint, totalRecycleFootprint, totalFootprint)

      PK: userID

BCNF

CommunityUser_TotalFoot (<u>userID</u>, userIC, fullName, password, phoneNo, workStatus, workSector, address, category, occupants, totalWaterFootprint, totalElectricFootprint, totalRecycleFootprint, totalFootprint)

      PK: userID

5. MBIP_AdminUser
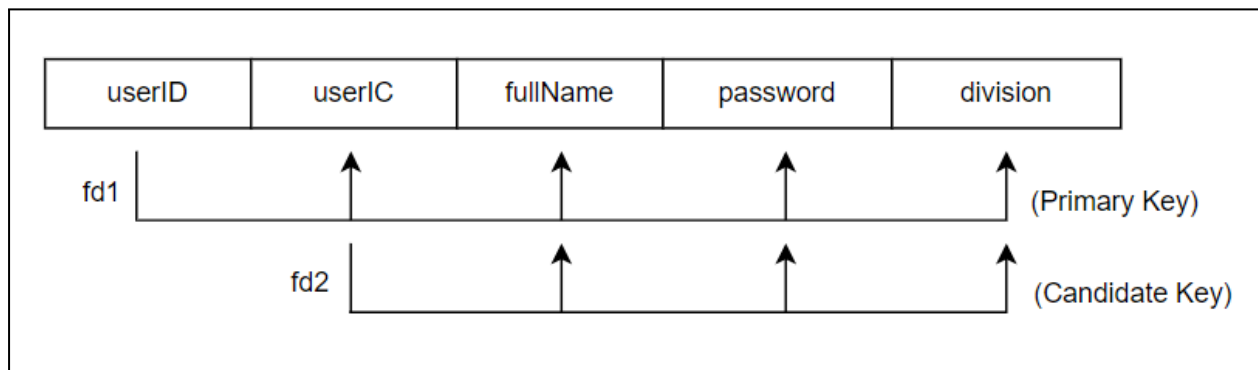
1NF

MBIP_AdminUser (<u>userID</u>, userIC, fullName, password, division)
    PK: userID

2NF

MBIP_AdminUser (<u>userID</u>, userIC, fullName, password, division)
    PK: userID

3NF

MBIP_AdminUser (<u>userID</u>, userIC, fullName, password, division)
    PK: userID

BCNF

MBIP_AdminUser (<u>userID</u>, userIC, fullName, password, division)
    PK: userID

5.0 Relational DB Schemas (after normalization)

ElectricWaterConsumption(<u>userID, Month</u>, electricConsumptionkWh, waterConsumptionM3)
        PK: userID, Month
        FK: userID references CommunityUser_TotalFoot (userID)

RecycleDetails (<u>userID, Month</u>, oilKG, wasteKG)
        PK: userID, Month
        FK: userID references CommunityUser_TotalFoot (userID)

MonthlyFootprint (<u>userID, Month</u>, waterFootprint, electricFootprint, recycleFootprint, monthlyTotalFootprint)
        PK: userID, Month
        FK: userID references CommunityUser_TotalFoot (userID)

CommunityUser_TotalFoot (<u>userID</u>, userIC, fullName, password, phoneNo, workStatus, workSector, address, category, occupants, totalWaterFootprint, totalElectricFootprint, totalRecycleFootprint, totalFootprint)
        PK: userID

MBIP_AdminUser (<u>userID</u>, userIC, fullName, password, division)
        PK: userID

ElectricWaterConsumption

FK			
userID	<u>Month</u>	electricConsumptionkWh	waterConsumptionM3

PK (userID, Month)

RecycleDetails

FK			
userID	<u>Month</u>	oilKG	wasteKG

PK (userID, Month)

MonthlyFootprint

FK				
userID	<u>Month</u>	waterFootprint	electricFootprint	recycleFootprint	monthlyTotalFootprint

PK (userID, Month)

CommunityUser_TotalFoot

userID	userIC	fullName	password	phoneNo	workStatus	workSector	address	category	occupants	totalWater Footprint	totalElectric Footprint	totalRecycle Footprint	totalFootprint

PK (userID)

MBIP_AdminUser

userID	userIC	fullName	password	division

PK (userID)

5.1 Final Logical ERD (based on 5.0 schemas)

ElectricWaterConsumption
- userID {FK} {PK}
- Month {PK}
- electricConsumptionkWh
- waterConsumptionM3

1 .. * ◄Submit 1 .. 1

CommunityUser_TotalFoot
- userID {PK}
- userIC
- fullName
- password
- phoneNo
- workStatus
- workSector
- address
- category
- occupants
- totalWaterFootprint
- totalElectricFootprint
- totalRecycleFootprint
- totalFootprint

1 .. 1 Submit ► 1 .. *

RecycleDetails
- userID {FK} {PK}
- Month {PK}
- oilKG
- wasteKG

1 .. 1 Have ► 1 .. *

MonthlyFootprint
- userID {FK} {PK}
- Month {PK}
- waterFootprint
- electricFootprint
- recycleFootprint
- monthlyTotalFootprint

MBIP_AdminUser
- userID {PK}
- userIC
- fullName
- password
- division

Figure 3

6.0 SQL Statements (DDL & DML)

Create all tables
#The purpose of these SQL statements are to create the table of the entity and its attributes.

```
CREATE TABLE ElectricWaterConsumption(
     userID VARCHAR2(10) NOT NULL,
     Month VARCHAR2(10) NOT NULL,
     electricConsumptionkWh NUMBER(5) NOT NULL,
     waterConsumptionM3 NUMBER(10,2) NOT NULL,
     CONSTRAINT PK_ElectricWaterConsumption PRIMARY KEY (userID,
Month)
);

CREATE TABLE RecycleDetails(
     userID VARCHAR2(10) NOT NULL,
     Month VARCHAR2(10) NOT NULL,
     oilKG NUMBER(5) NOT NULL,
     wasteKG NUMBER(5) NOT NULL,
     CONSTRAINT PK_RecycleDetails PRIMARY KEY (userID, Month)
);

CREATE TABLE MonthlyFootprint(
     userID VARCHAR2(10) NOT NULL,
     Month VARCHAR2(10) NOT NULL,
     waterFootprint NUMBER(10) NOT NULL,
     electricFootprint NUMBER(10) NOT NULL,
     recycleFootprint NUMBER(10) NOT NULL,
     monthlyTotalFootprint NUMBER(38) NOT NULL,
     CONSTRAINT PK_MonthlyFootprint PRIMARY KEY (userID, Month)
);

CREATE TABLE CommunityUser_TotalFoot(
     userID VARCHAR2(10),
     userIC VARCHAR2(14) NOT NULL,
     fullName VARCHAR2(50) NOT NULL,
     password VARCHAR2(20) NOT NULL,
     phoneNo VARCHAR2(15) NOT NULL,
     workStatus VARCHAR2(10) NOT NULL,
     workSector VARCHAR2(20) NOT NULL,
     address   VARCHAR2(150) NOT NULL,
     category VARCHAR2(10) NOT NULL,
     occupants NUMBER(2) NOT NULL,
     totalWaterFootprint NUMBER(10),
     totalElectricFootprint NUMBER(10),
```

```
        totalRecycleFootprint NUMBER(10),
        totalFootprint NUMBER(38),
        CONSTRAINT PK_CommunityUser_TotalFoot PRIMARY KEY (userID)
);


CREATE TABLE MBIP_AdminUser (
        userID VARCHAR2(10) NOT NULL,
        userIC VARCHAR2(14) NOT NULL,
        fullName VARCHAR2(50) NOT NULL,
        password VARCHAR2(20) NOT NULL,
        division VARCHAR2(50) NOT NULL,
        CONSTRAINT PK_MBIP_AdminUser PRIMARY KEY (userID)
);
```

Add Foreign Key to 3 tables
#The purpose of these SQL statements are to add the foreign key to the table..

```
ALTER TABLE ElectricWaterConsumption
ADD CONSTRAINT FK_ElectricWaterConsumption FOREIGN KEY(userID)
REFERENCES CommunityUser_TotalFoot(userID);


ALTER TABLE RecycleDetails
ADD CONSTRAINT FK_RecycleDetails FOREIGN KEY(userID)
REFERENCES CommunityUser_TotalFoot(userID);


ALTER TABLE MonthlyFootprint
ADD CONSTRAINT FK_MonthlyFootprint FOREIGN KEY(userID)
REFERENCES CommunityUser_TotalFoot(userID);
```

Insert all values into the created tables (10 community users and 2 admins)
#The purpose of these SQL statements is to insert the information of the user into CommunityUser_TotalFoot.

```
INSERT INTO CommunityUser_TotalFoot
VALUES('kuantong69', '030101-01-0001', 'KUAN JI TONG', 'KUAN123#',
'012-3456789', 'Bekerja', 'Sektor Awam', 'No. 26, Jln. 25/9, Taman
Sri Muda, Shah Alam, 40400, Shah Alam, Selangor', 'B1', 1, null,
null, null, null);


INSERT INTO CommunityUser_TotalFoot
VALUES('faridul33', '030202-02-0002', 'MD FARIDUL ISLAM',
'FARIDUL123#', '013-5121 311', 'Pelajar', 'Sektor Awam', '4, jalan
```

Tengku Ampuan Zabedah, Seksyen 9, Shah Alam, 40000, Shah Alam,
Selangor', 'B2', 2, null, null, null, null);

INSERT INTO CommunityUser_TotalFoot
VALUES('saifudin23', '030301-01-0003', 'AHMAD SAIFUDIN BIN NARDI
SUSANTO', 'SAIFUDIN123#', '012-98402784', 'Bekerja', 'Sektor
Swasta', 'Lot 4133, Jalan Perkhidmatan, Kampung Baru Sungai Buloh,
47000, Sungai Buloh, Selangor', 'B2', 3, null, null, null, null);

INSERT INTO CommunityUser_TotalFoot
VALUES('hariz99', '030412-05-0041', 'ALI HARIZ BIN ANUARI',
'ALI123#', '013-6156 181', 'Bekerja', 'Sektor Awam', 'Lot 4133,
Jalan Perkhidmatan, Kampung Baru Sungai Buloh, 47000, Sungai Buloh,
Selangor', 'A1', 6, null, null, null, null);

INSERT INTO CommunityUser_TotalFoot
VALUES('wang89', '030121-06-0301', 'WANG LEI', 'WANGLEI123#',
'012-9483965', 'Bekerja', 'Sektor Awam', 'J-31-G, Block J, Jaya One,
72A, Jalan Universiti, 46200, Petaling Jaya, Selangor', 'A2', 3,
null, null, null, null);

INSERT INTO CommunityUser_TotalFoot
VALUES('jianheng66', '040516-03-0023', 'KEW JIAN HENG', 'KEW123#',
'012-5569546', 'Pelajar', 'Sektor Swasta', 'No. 132, Jalan Kundor,
05400, Alor Setar', 'B1', 4, null, null, null, null);

INSERT INTO CommunityUser_TotalFoot
VALUES('yanhong23', '030605-01-8952', 'LEOW YAN HONG', 'LEOW123#',
'011-5269426', 'Bekerja', 'Sektor Awam', 'Jalan Iman, 81310 Skudai,
Johor', 'A2', 2, null, null, null, null);

INSERT INTO CommunityUser_TotalFoot
VALUES('mingze45', '031230-02-1001', 'LIM MING ZE', ':LIM123#',
'013-889 1111', 'Pelajar', 'Sektor Swasta', 'Lot G32, Ground Floor,
No 1, Jalan Tun Abdul Razak, 75400, Tesco Melaka, Peringgit,
Melaka', 'A2', 2, null, null, null, null);

INSERT INTO CommunityUser_TotalFoot
VALUES('aqil12', '030228-01-5630', 'MUHAMMAD AQIL HAZIQ BIN
ZULKARNAIN', 'AQIL123#', '018-561589', 'Bekerja', 'Sektor Swasta',
'Stesyen Minyak Shell, Adjacent To Wisma Innoprise, Jalan Kota
Kinabalu-Sulaman Highway, Kota Kinabalu, 89257, Tamparuli, Sabah',
'B2', 1, null, null, null, null);

```
INSERT INTO CommunityUser_TotalFoot
VALUES('fahim78', '030501-03-5363', 'MUHAMMAD FAHIM BIN ZULKIFLI
AMIR', 'FAHIM123#', '019-2901624', 'Pelajar', 'Sektor Awam', '8,
Jln. Beserah, 25000 Kuantan, Pahang', 'A1', 4, null, null, null,
null);
```

#The purpose of these SQL statements is to insert the information of the admin user into MBIP_AdminUser.

```
INSERT INTO MBIP_AdminUser
VALUES('aiman111', '000221-01-0032' , 'MOHAMAD AIMAN BIN SAMSUL',
'AIMAN123', 'GIS and Development Planning Division');
```
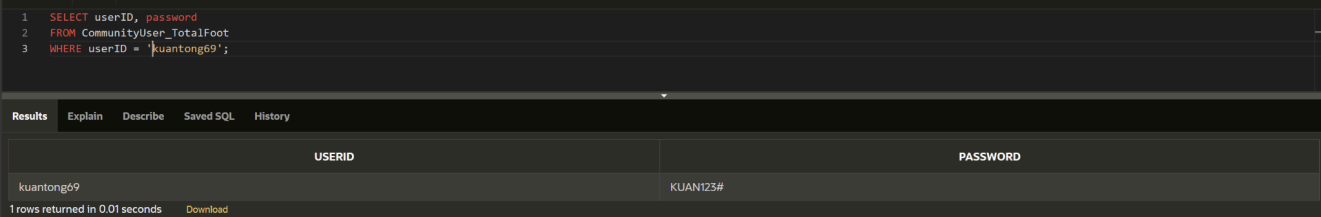
```
INSERT INTO MBIP_AdminUser
VALUES('iddin888', '980502-01-0320' , 'MOHAMAD IDDIN BIN SHED AIMAN',
'IDDIN666', 'Planning Control Division');
```

Login
#The purpose of these SQL statements are to display user ID and password for each user.

#These SQL statements will display the user ID and password of the user.

```
SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'kuantong69';
```


Figure 4: Query to display community user's userID and password

```
SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'faridul33';
```

```
SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'saifudin23';
```

```
SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'hariz99';

SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'wang89';

SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'jianheng66';

SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'yanhong23';

SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'mingze45';


SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'aqil12';

SELECT userID, password
FROM CommunityUser_TotalFoot
WHERE userID = 'fahim78';
```
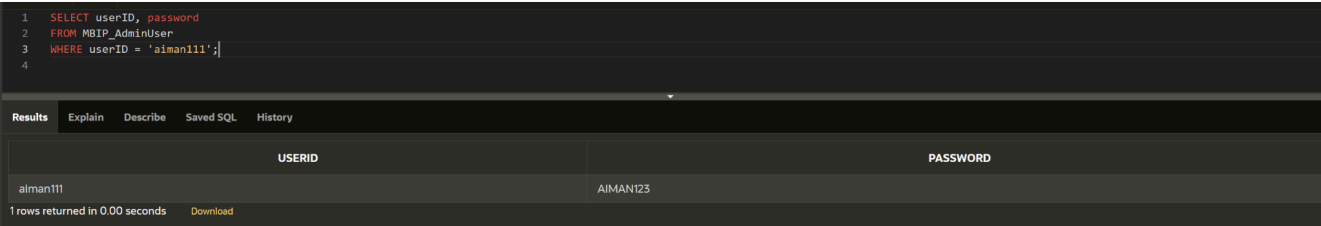
#These SQL statements will display the user ID and password of the Admin user.

```
SELECT userID, password
FROM MBIP_AdminUser
WHERE userID = 'aiman111';
```


Figure 5: Query to display MBIP admin's userID and password

```
SELECT userID, password
FROM MBIP_AdminUser
WHERE userID = 'iddin888';
```

Record Consumption Details
#The purpose of these SQL statements are to insert user's consumption data into ElectricWaterConsumption table

```
INSERT INTO ElectricWaterConsumption
VALUES('kuantong69', 'Januari', 474, 22);

INSERT INTO ElectricWaterConsumption
VALUES('faridul33', 'Januari', 450, 18);

INSERT INTO ElectricWaterConsumption
VALUES('saifudin23', 'Januari', 451, 19);

INSERT INTO ElectricWaterConsumption
VALUES('hariz99', 'Januari', 452, 20);

INSERT INTO ElectricWaterConsumption
VALUES('wang89', 'Januari', 453, 21);

INSERT INTO ElectricWaterConsumption
VALUES('jianheng66', 'Januari', 454, 22);

INSERT INTO ElectricWaterConsumption
VALUES('yanhong23', 'Januari', 455, 23);

INSERT INTO ElectricWaterConsumption
VALUES('mingze45', 'Januari', 456, 24);

INSERT INTO ElectricWaterConsumption
VALUES('aqil12', 'Januari', 457, 25);

INSERT INTO ElectricWaterConsumption
VALUES('fahim78', 'Januari', 458, 26);

INSERT INTO ElectricWaterConsumption
VALUES('kuantong69', 'Februari', 459, 27);

INSERT INTO ElectricWaterConsumption
VALUES('faridul33', 'Februari', 460, 28);
```

```
INSERT INTO ElectricWaterConsumption
VALUES('saifudin23', 'Februari', 461, 29);

INSERT INTO ElectricWaterConsumption
VALUES('hariz99', 'Februari', 462, 30);

INSERT INTO ElectricWaterConsumption
VALUES('wang89', 'Februari', 463, 31);

INSERT INTO ElectricWaterConsumption
VALUES('jianheng66', 'Februari', 464, 32);

INSERT INTO ElectricWaterConsumption
VALUES('yanhong23', 'Februari', 465, 33);

INSERT INTO ElectricWaterConsumption
VALUES('mingze45', 'Februari', 466, 34);

INSERT INTO ElectricWaterConsumption
VALUES('aqil12', 'Februari', 467, 35);

INSERT INTO ElectricWaterConsumption
VALUES('fahim78', 'Februari', 468, 36);
```

#The purpose of these SQL statements are to insert user's recycle details into RecycleDetails table

```
INSERT INTO RecycleDetails
VALUES('kuantong69', 'Januari', 25, 25);

INSERT INTO RecycleDetails
VALUES('faridul33', 'Januari', 20, 12);

INSERT INTO RecycleDetails
VALUES('saifudin23', 'Januari', 19, 13);

INSERT INTO RecycleDetails
VALUES('hariz99', 'Januari', 18, 14);

INSERT INTO RecycleDetails
VALUES('wang89', 'Januari', 17, 15);

INSERT INTO RecycleDetails
VALUES('jianheng66', 'Januari', 16, 16);
```

```sql
INSERT INTO RecycleDetails
VALUES('yanhong23', 'Januari', 15, 17);

INSERT INTO RecycleDetails
VALUES('mingze45', 'Januari', 14, 18);

INSERT INTO RecycleDetails
VALUES('aqil12', 'Januari', 13, 19);

INSERT INTO RecycleDetails
VALUES('fahim78', 'Januari', 12, 20);

INSERT INTO RecycleDetails
VALUES('kuantong69', 'Februari', 11, 21);

INSERT INTO RecycleDetails
VALUES('faridul33', 'Februari', 10, 22);

INSERT INTO RecycleDetails
VALUES('saifudin23', 'Februari', 9, 23);

INSERT INTO RecycleDetails
VALUES('hariz99', 'Februari', 8, 24);

INSERT INTO RecycleDetails
VALUES('wang89', 'Februari', 22, 25);

INSERT INTO RecycleDetails
VALUES('jianheng66', 'Februari', 23, 26);

INSERT INTO RecycleDetails
VALUES('yanhong23', 'Februari', 24, 27);

INSERT INTO RecycleDetails
VALUES('mingze45', 'Februari', 25, 28);

INSERT INTO RecycleDetails
VALUES('aqil12', 'Februari', 26, 29);

INSERT INTO RecycleDetails
VALUES('fahim78', 'Februari', 27, 30);
```
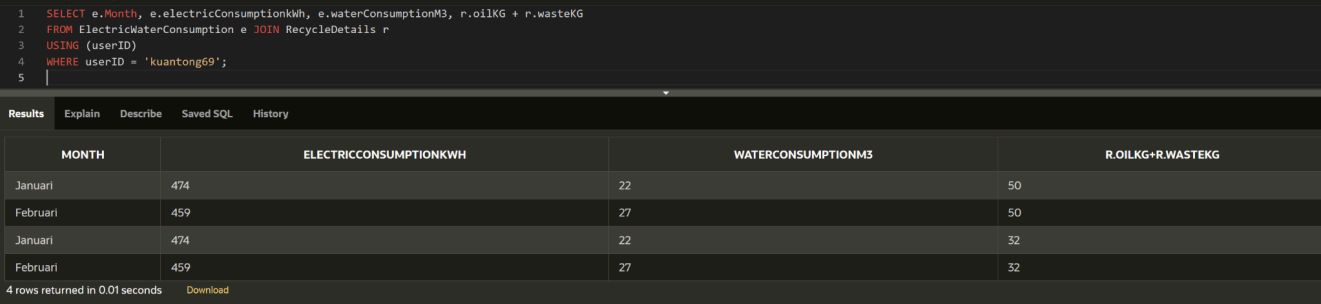
Calculate Carbon Footprint

#The purpose of these sql statements is to display the water consumption, electricity consumption and total recycled waste by month

```
SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'kuantong69';
```


Figure 6: Query to display monthly consumptions and recycle details of a user

```
SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'faridul33';


SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'saifudin23';


SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'hariz99';


SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'wang89';
```

```sql
SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'jianheng66';


SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'yanhong23';



SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'mingze45';


SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'aqil12';


SELECT DISTINCT e.Month, e.electricConsumptionkWh,
e.waterConsumptionM3, r.oilKG + r.wasteKG
FROM ElectricWaterConsumption e JOIN RecycleDetails r
USING (userID)
WHERE userID = 'fahim78';
```

===

#The purpose of these sql statements is to insert the calculated carbon footprint values for water, electricity and recycle waste for each users by month into the MonthlyFootprint table

```sql
INSERT INTO MonthlyFootprint
VALUES('kuantong69', 'Januari', 9.2, 276.8, 143, 143);

INSERT INTO MonthlyFootprint
VALUES('kuantong69', 'Februari', 11.3, 268.1, 91.5, 187.9);
```

```sql
INSERT INTO MonthlyFootprint
VALUES('faridul33', 'Januari', 7.5, 262.8, 91.5, 178.8);

INSERT INTO MonthlyFootprint
VALUES('faridul33', 'Februari', 11.7, 268.6, 91.5, 188.8);

INSERT INTO MonthlyFootprint
VALUES('saifudin23', 'Januari', 8, 263.4, 91.5, 179.9);

INSERT INTO MonthlyFootprint
VALUES('saifudin23', 'Februari', 12.2, 269.2, 91.5, 189.9);

INSERT INTO MonthlyFootprint
VALUES('hariz99', 'Januari', 8.4, 264, 91.5, 180.9);

INSERT INTO MonthlyFootprint
VALUES('hariz99', 'Februari', 12.6, 269.8, 91.5, 190.9);

INSERT INTO MonthlyFootprint
VALUES('wang89', 'Januari', 8.8, 264.6, 91.5, 181.9);
INSERT INTO MonthlyFootprint
VALUES('wang89', 'Februari', 13, 270.4, 134.4, 149);

INSERT INTO MonthlyFootprint
VALUES('jianheng66', 'Januari', 9.2, 265.1, 91.5, 182.8);

INSERT INTO MonthlyFootprint
VALUES('jianheng66', 'Februari', 13.4, 271, 140.1, 144.3);

INSERT INTO MonthlyFootprint
VALUES('yanhong23', 'Januari', 9.6, 265.7, 91.5, 183.8);

INSERT INTO MonthlyFootprint
VALUES('yanhong23', 'Februari', 13.8, 271.6, 145.9, 139.5);

INSERT INTO MonthlyFootprint
VALUES('mingze45', 'Januari', 10.1, 266.3, 91.5, 184.9);

INSERT INTO MonthlyFootprint
VALUES('mingze45', 'Februari', 14.2, 272.1, 151.6, 134.7);

INSERT INTO MonthlyFootprint
VALUES('aqil12', 'Januari', 10.5, 266.9, 91.5, 185.9);
```

```
INSERT INTO MonthlyFootprint
VALUES('aqil12', 'Februari', 14.7, 272.7, 157.3, 130.1);


INSERT INTO MonthlyFootprint
VALUES('fahim78', 'Januari', 11, 267.5, 91.5, 187);


INSERT INTO MonthlyFootprint
VALUES('fahim78', 'Februari', 15.1, 273.3, 163, 125.4);
```

==

**#The purpose of these sql statements is to update CommunityUser_TotalFoot table
with the calculated total carbon footprint values for each users**

```
UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 20.5, totalElectricFootprint = 544.9,
totalRecycleFootprint = 234.5, totalFootprint = 330.9
WHERE userID = 'kuantong69';


UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 19.2, totalElectricFootprint = 531.4,
totalRecycleFootprint = 183, totalFootprint = 367.6
WHERE userID = 'faridul33';


UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 20.2, totalElectricFootprint = 532.6,
totalRecycleFootprint = 183, totalFootprint = 369.8
WHERE userID = 'saifudin23';


UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 21, totalElectricFootprint = 533.8,
totalRecycleFootprint = 183, totalFootprint = 371.8
WHERE userID = 'hariz99';


UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 21.8, totalElectricFootprint = 535,
totalRecycleFootprint = 225.9, totalFootprint = 330.9
WHERE userID = 'wang89';


UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 22.6, totalElectricFootprint = 536.1,
totalRecycleFootprint = 231.6, totalFootprint = 327.1
WHERE userID = 'jianheng66';
```

```
UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 23.4, totalElectricFootprint = 537.3,
totalRecycleFootprint = 237.4, totalFootprint = 323.3
WHERE userID = 'yanhong23';


UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 24.3, totalElectricFootprint = 538.4,
totalRecycleFootprint = 243.1, totalFootprint = 319.6
WHERE userID = 'mingze45';


UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 25.2, totalElectricFootprint = 539.6,
totalRecycleFootprint = 248.8, totalFootprint = 316
WHERE userID = 'aqil12';


UPDATE CommunityUser_TotalFoot
SET totalWaterFootprint = 26.1, totalElectricFootprint = 540.8,
totalRecycleFootprint = 254.5, totalFootprint = 312.4
WHERE userID = 'fahim78';
```
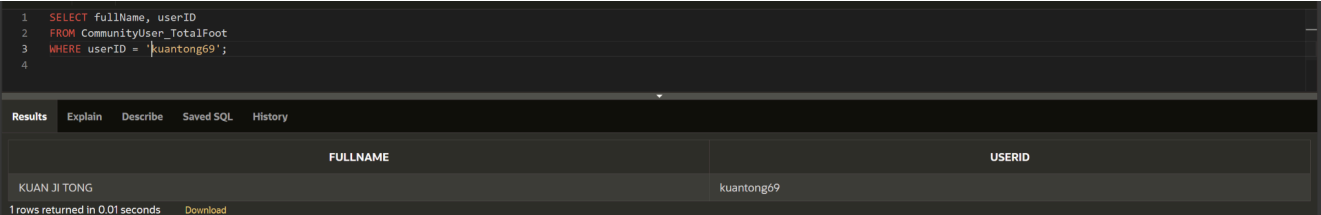
Display User Dashboard

#The purpose of these SQL statements are to display all the values in the admin dashboard as shown in the interface.

#These SQL statements will display the full name and userID of the user.

```
SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'kuantong69';
```


Figure 7: Query to display community user's full name and userID

```
SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'faridul33';
```

```
SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'saifudin23';


SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'hariz99';


SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'wang89';


SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'jianheng66';


SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'yanhong23';


SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'mingze45';


SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'aqil12';


SELECT fullName, userID
FROM CommunityUser_TotalFoot
WHERE userID = 'fahim78';
```

==

#These SQL statements will display the total footprint values of the user.

```
SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'kuantong69';
```

Figure 8: Query to display community user's total footprint for all consumptions and recycles

```
SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'faridul33';


SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'saifudin23';


SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'hariz99';


SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'wang89';


SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'jianheng66';


SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'yanhong23';


SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'mingze45';
```
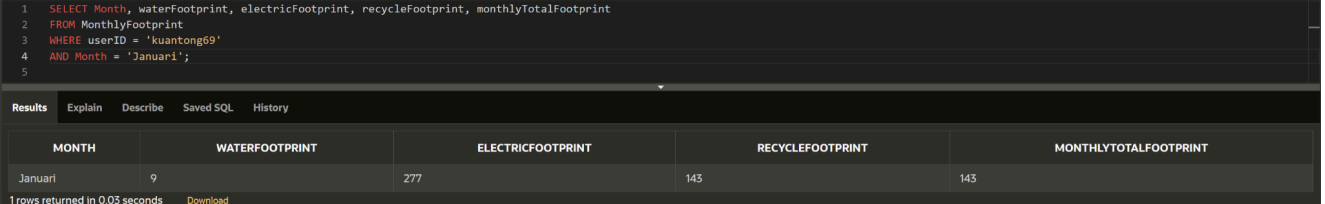
```
SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'aqil12';


SELECT totalFootprint, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint
FROM CommunityUser_TotalFoot
WHERE userID = 'fahim78';
```

==

#These SQL statements will display the monthly footprint values of the user

```
SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'kuantong69'
AND Month = 'Januari';
```


Figure 9: Query to display community user's monthly footprint values

```
SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'kuantong69'
AND Month = 'Februari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'faridul33'
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'faridul33'
AND Month = 'Februari';
```

```sql
SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'saifudin23'
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'saifudin23'
AND Month = 'Februari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'hariz99'
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'hariz99'
AND Month = 'Februari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'wang89'
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'wang89'
AND Month = 'Februari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'jianheng66'
AND Month = 'Januari';
```
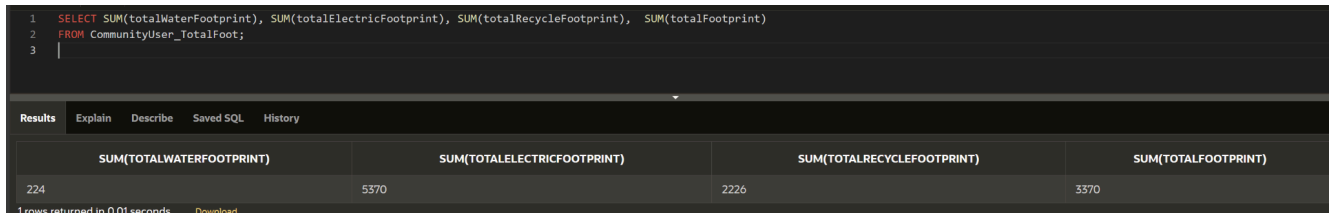
```sql
SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'jianheng66'
AND Month = 'Februari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'yanhong23'
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'yanhong23'
AND Month = 'Februari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'mingze45'
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'mingze45'
AND Month = 'Februari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'aqil12'
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'aqil12'
AND Month = 'Februari';
```

```
SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'fahim78'
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint
WHERE userID = 'fahim78'
AND Month = 'Februari';
```
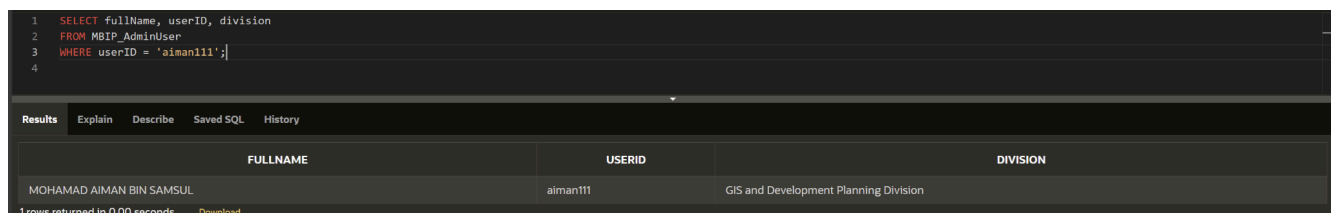
<u>Display Admin Dashboard</u>

#The purpose of these SQL statements are to display all the values in the admin dashboard as shown in the interface.

#These SQL statements will display full name, userID and the division of the admin.

```
SELECT fullName, userID, division
FROM MBIP_AdminUser
WHERE userID = 'aiman111';
```


Figure 10: Query to display MBIP admin's full name, userID and division

```
SELECT fullName, userID, division
FROM MBIP_AdminUser
WHERE userID = 'iddin888';


=================================================
```

#This SQL statement will display the total number of users in the system.

```
SELECT COUNT(userID) AS "Total Users"
FROM CommunityUser_TotalFoot;
```

Figure 11: Query to display the total community users in the system

==

#These SQL statements will display the total footprint values as well as the monthly footprint values of the user with the lowest total footprint value.

```
SELECT fullName, userIC, totalWaterFootprint, totalElectricFootprint,
totalRecycleFootprint,  totalFootprint
FROM CommunityUser_TotalFoot
WHERE totalFootprint IN (SELECT MIN(totalFootprint) FROM
CommunityUser_TotalFoot);
```


Figure 12: Query to display the total footprint values of the user with the lowest total footprint value

```
SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint JOIN CommunityUser_TotalFoot
USING(userID)
WHERE totalFootprint IN (SELECT MIN(totalFootprint) FROM
CommunityUser_TotalFoot)
AND Month = 'Januari';


SELECT Month, waterFootprint, electricFootprint, recycleFootprint,
monthlyTotalFootprint
FROM MonthlyFootprint JOIN CommunityUser_TotalFoot
USING(userID)
WHERE totalFootprint IN (SELECT MIN(totalFootprint) FROM
CommunityUser_TotalFoot)
AND Month = 'Februari';


================================================
```

#These SQL statements will display the total monthly footprints and total footprints of all users in total

```
SELECT SUM(totalWaterFootprint), SUM(totalElectricFootprint),
SUM(totalRecycleFootprint),  SUM(totalFootprint)
FROM CommunityUser_TotalFoot;
```


Figure 13: Query to display the total monthly footprints and total footprints of all users in total

```
SELECT Month, SUM(waterFootprint), SUM(electricFootprint),
SUM(recycleFootprint), SUM(monthlyTotalFootprint)
FROM MonthlyFootprint
WHERE Month = 'Januari'
GROUP BY Month;


SELECT Month, SUM(waterFootprint), SUM(electricFootprint),
SUM(recycleFootprint), SUM(monthlyTotalFootprint)
FROM MonthlyFootprint
WHERE Month = 'Februari'
GROUP BY Month;
```

7.0 Summary

In this phase of the project, we did an extensive update on our system, solving the data management issues faced by the Iskandar Puteri Calendar Competition. We enhance some features like efficient carbon footprint calculations, registration and login processes, and monitoring dashboards. We update the business rules, conceptual ERD, logical ERD in the database conceptual design part, and the most important part is the normalization part which we achieved with the BCNF relations to ensure the database performance and organization.

To reduce the data redundancy, normalization methods are used in entities which are MonthDay,ElectricWaterConsumption, RecycleDetails, MonthlyFootprint, CommunityUser_TotalFoot and MBIP_AdminUser. The relational database schema provides a more clearer structure for data organisation, lowering the possibility of anomalies. Then, we translate the conceptual ERD into SQL statements (DDL & DML) to produce a more detailed system design. Lastly, this phase provides a structure for a more efficient and easily usable system, customized to fulfil the requirements for the Iskandar Puteri Calendar Competition and MBIP.

8.0 Interfaces

<u>Register Community User</u>

Log-in Community User and MBIP Admin

Community user record consumption and recycle details

Calculate Carbon footprints for Community User using recorded consumption and recycle details
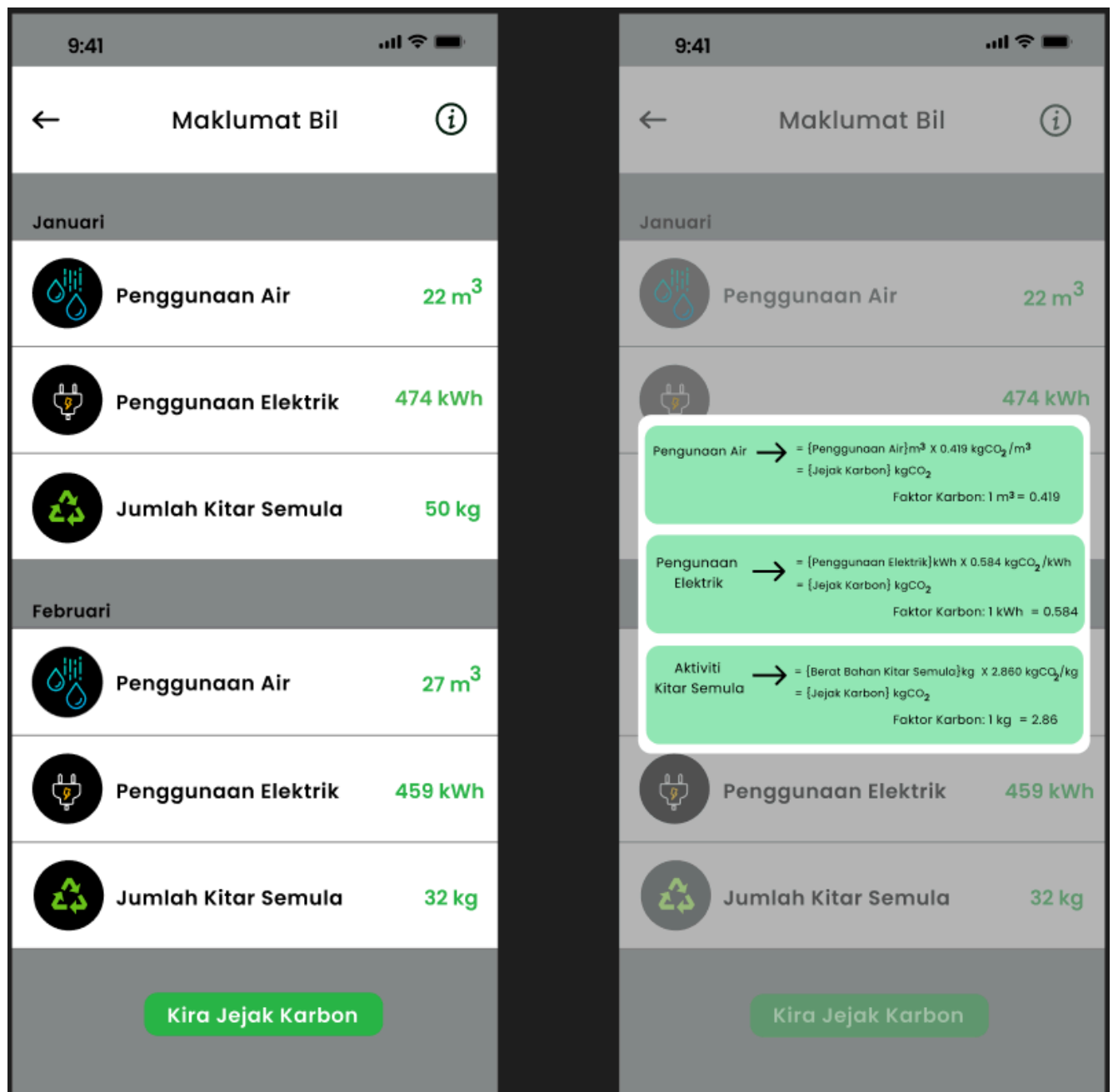
Community User dashboard (Screen 1) and MBIP Admin dashboard (Screen 2 & 3)

9:41

Mohamad Aiman bin Samsul
aiman111
GIS and Development Planning Division

Statistik

Jumlah Pengguna: 10
Jejak Karbon paling rendah: MUHAMMAD FAHIM BIN ZULKIFLI AMIR

MAKLUMAT TERPERINCI

Nama Penuh: MUHAMMAD FAHIM BIN ZULKIFLI AMIR

No. IC: 030501-03-5363

Jumlah Jejak Karbon: 312 kgCO2

26 kgCO$_2$ 541 kgCO$_2$ 255 kgCO$_2$

Jejak Karbon Bulanan

Januari

11 kgCO$_2$ 268 kgCO$_2$ 92 kgCO$_2$

Total : 187 kgCO$_2$

Februari

15 kgCO$_2$ 273 kgCO$_2$ 163 kgCO$_2$

Total : 125 kgCO$_2$

Jumlah Jejak Karbon

Pelepasan Karbon

224 kgCO$_2$

3370 kgCO$_2$

5370 kgCO$_2$

Pengurangan Karbon

2226 kgCO$_2$

Jejak Karbon Bulanan

Januari

Pelepasan	Pengurangan	Jumlah
93 kgCO$_2$	971 kgCO$_2$	1790 kgCO$_2$
2664 kgCO$_2$		

Februari

Pelepasan	Pengurangan	Jumlah
132 kgCO$_2$	1260 kgCO$_2$	1581 kgCO$_2$
2707 kgCO$_2$		