PES UNIVERSITY

100 feet Ring Road, BSK 3rd Stage, Bengaluru 560085

Department of Computer Science and Engineering

Jan – May 2020

UE18CS252
Database Management Systems

Project Report

Electricity Billing Management System

PES1201801880
Ravikiran D K
4th Sem
Sec: I
Roll No: 49

PROJECT SUMMARY

The goal of this Electric billing system is to store the billing information of the Electricity consumer.

ER diagram and Schema are fully Normalized to 3rd NF.

This model is made in MYSQL

Triggers will trigger upon changing units,rate and paid.

This system reduces the manual work by ⅔ rd so increases automation

Introduction

The goal of this Electric billing system is to store the billing information of the Electricity consumer.

So entities like Electricity Supply Companies and Electricity Supply Branches store the information about companies and branches respectively.

The Locations of the Branches are stored in Region Located entity,and the average rate of electricity given by various regions are put in Rate entity.

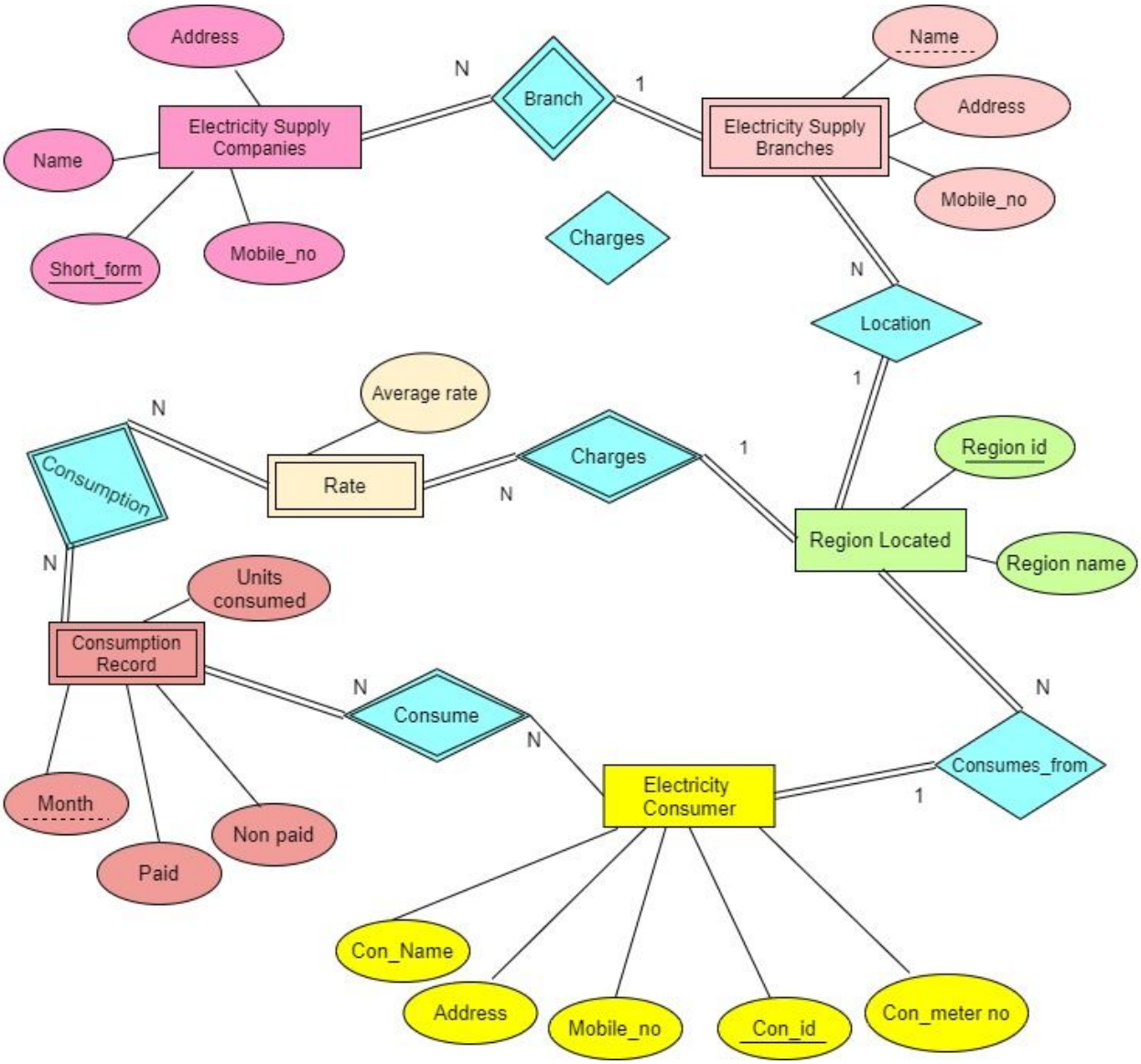
The consumption record contains unit consumed ,paid and unpaid amounts of Electricity consumer,

Electricity Consumer contains name ,meter no,and many personal information of consumers.

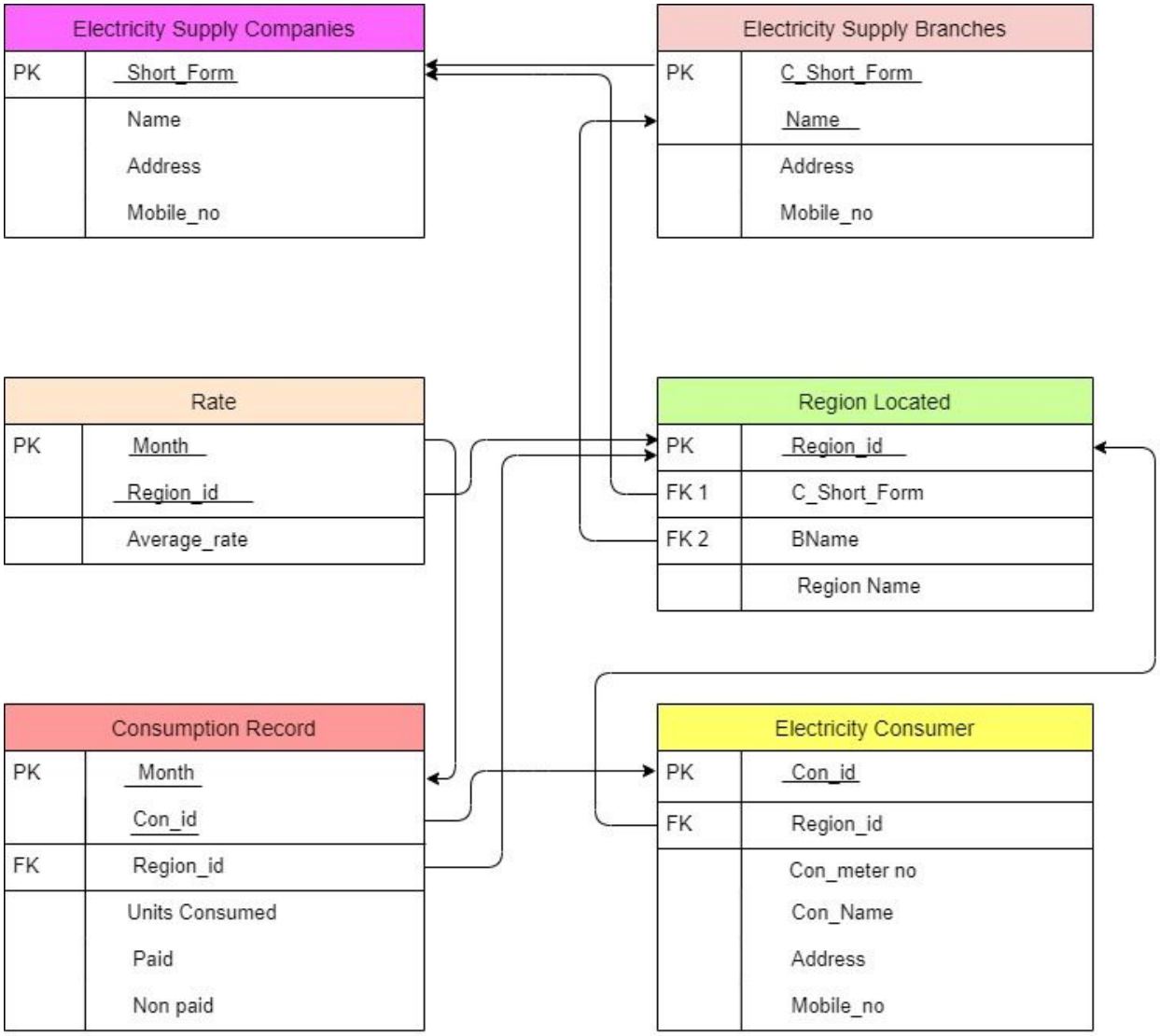
Transaction of system is to update consumption record of consumers on the basis of units consumed and average rate of the electricity given by that particular region supplier.

Data Model

ER Diagram



RELATIONAL SCHEMA



FD and Normalization

Functional Dependencies:

1) Short_Form -> Name , Address , Mobile_no
   Name -> Short_Form , Address , Mobile_no
2) (C_Short_Form , Name) -> Address , Mobile_no
3) Region_id -> Region Name , BName , C_Short_Form

4) (Month , Region_id) -> Average rate
5) (Month,Con_id) -> Region_id , Units Consumed , Paid , Non paid
6) Con_id -> Con_meter no , Region_id , Com_Name , Address , Mobile_no
(Con_meter no , Region_id) -> Con_id , Com_Name , Address , Mobile_no

Candidate Keys:

1) Short_Form (Primary Key)
Name
It can be used to uniquely identify Short_Form , Address and Mobile_no as Name->Short_Form , Address , Mobile_no (its attribute closure is the set of all attributes of the table).
2) (C_Short_Form , Name) (Primary Key)
3) Region_id (Primary Key)
4) (Month , Region_id)
Before adding region_id as foreign key it was okay taking Month and Average rate as primary key but now these combination can repeat for different region_id . so we take candidate key as (Month , Region_id) since this combination can derive all the attributes of the table .
5) (Month,Con_id)
Taking foreign key with primary key(before adding FK) we can derive all the attributes of the table
6) Con_id (Primary Key)
(Con_meter no , Region_id)
Distinct meter no is given to every consumer in a Region.

Normal forms of Relations:
Here all relations are their in 3rd or more than 3rd Normal Form.

2nd Normal Forms Violations:

- If we add company name in Electricity Supply Branches Table it will result in 2nd normal form violation as Company name is functionally dependent on partial key C_Short_Form.
- If we add Region Name in Rate table it will be 2nd normal form violation as Region Name is functionally dependent on partial key Region_id.
- If we add Con_Name or Con_meter_no in Consumption Record table it will be 2nd normal form violation as Con_Name and Con_meter_no are functionally dependent on partial key Con_id.

3rd Normal Forms Violations:

- If we add Company Name in Region Located table it will be 3rd Normal Form Violation because C_Short_Form -> company Name and Region_id -> C_Short_Form . it becomes transitive.
- If we add Region Name in Consumption Record table it is 3rd Normal Form Violation as Region_id -> Region Name and (Month , Con_id) -> Region_id it is transitive.
- If we add Region Name in Electricity Consumer table it is 3rd Normal Form violation as

- Region_id -> Region Name and Con_id -> Region_id . it becomes transitive.

For Lossless join property checking purpose we are adding columns and decomposing Electricity Consumer(R) table into
Electricity Consumer1(R1) and meter table(R2)

Electricity Consumer (R)

Con_id	Con_Name	Address	Mobile_no	Con_meter_no	Region_id	Meter_power

Electricity Consumer1 (R1)

Con_id	Con_Name	Address	Mobile_no	Con_meter_no	Region_id

Meter_Table (R2)

Con_meter_no	Region_id	Meter_power

Checking lossless join

- Joining R1 AND R2 we should get R
- All attributes in R should be there in R1 or R2

R1 U R2 = R (We can see that all attributes present in R are there in R1 or R2)

R1 ∩ R2 = { Con_meter_no , Regio_id }

In R1 { Con_meter_no , Regio_id } -> { Meter_power }

{ Con_meter_no , Regio_id }$^{+}$ -> { Con_meter_no , Regio_id , Meter_power}

So closure of this gives all attributes in R2 , satisfying in one of the Relation table proves That R1 and R2 are Lossless Join Decomposition.

DDL

CREATE QUERIES

```
CREATE TABLE Electricity_Supply_Companies (
   Short_Form CHAR(6) NOT NULL,
   Name VARCHAR(40) NOT NULL,
   Address TEXT NOT NULL,
   Mobile_no DECIMAL(10) NOT NULL,
   PRIMARY KEY (Short_Form),
   UNIQUE (Name)
);
CREATE TABLE Electricity_Supply_Branches (
   Name VARCHAR(20) NOT NULL,
        C_Short_Form CHAR(6) NOT NULL,
   Address TEXT NOT NULL,
   Mobile_no DECIMAL(10) NOT NULL,
   PRIMARY KEY (Name , C_Short_Form),
   FOREIGN KEY (C_Short_Form)
      REFERENCES Electricity_Supply_Companies (Short_Form)
);
CREATE TABLE Region_Located (
   Region_id INT NOT NULL,
   Region_Name VARCHAR(20) NOT NULL,
   C_Short_Form CHAR(6) NOT NULL,
   BName VARCHAR(20) NOT NULL,
   PRIMARY KEY (Region_id),
   FOREIGN KEY (C_Short_Form)
      REFERENCES Electricity_Supply_Companies (Short_Form),
   FOREIGN KEY (BName)
      REFERENCES Electricity_Supply_Branches (Name)
);
CREATE TABLE Rate (
```

```
    Month VARCHAR(10) NOT NULL,
    Region_id INT NOT NULL,
    Average_rate INT NOT NULL,
    PRIMARY KEY (Month , Region_id),
    FOREIGN KEY (Region_id)
        REFERENCES Region_Located (Region_id)
);
CREATE TABLE Electricity_Consumer (
    Con_id INT NOT NULL,
    Con_meter_no INT NOT NULL,
    Region_id INT NOT NULL,
    Con_Name VARCHAR(20) NOT NULL,
    Address TEXT NOT NULL,
    Mobile_no DECIMAL(12) NOT NULL,
    PRIMARY KEY (Con_id),
    FOREIGN KEY (Region_id)
        REFERENCES Region_Located (Region_id),
    UNIQUE (Con_meter_no , Region_id)
);
CREATE TABLE Consumption_Record (
    Month VARCHAR(10) NOT NULL,
    Con_id INT NOT NULL,
    Region_id INT NOT NULL,
    Units_Consumed INT NOT NULL,
    Paid FLOAT NOT NULL,
    Non_paid FLOAT NOT NULL,
    PRIMARY KEY (Month , Con_id),
    FOREIGN KEY (Con_id)
        REFERENCES Electricity_Consumer (Con_id)
        ON DELETE CASCADE,
    FOREIGN KEY (Region_id)
        REFERENCES Region_Located (Region_id)
);
ALTER TABLE Rate
ADD FOREIGN KEY (Month) REFERENCES Consumption_Record(Month);

ALTER TABLE Rate
ADD CHECK (Average_rate>=2);
```

Some sql queries for inserting

```
insert into Electricity_Supply_Companies
values ('MESCOM', 'Mangalore Electricity Supply Company', 'MESCOM Bhavan, Kavoor Cross
Road, Bejai, Mangalore, Karnataka 575004', 1234567890);

insert into Electricity_Supply_Companies
values ('HESCOM', 'Hubali Electricity Supply Company', 'Ganesh Path, Durgad Bail, New Hubli,
Hubli, Karnataka 580028', '1234567892'
);

insert into Electricity_Supply_Branches
values ('shimoga','MESCOM','shimoga road',1234567895);


insert into Electricity_Supply_Branches
values ('hubballi','HESCOM','hubballi road',1234567898);


insert into Region_Located
values (131,'sagard','MESCOM','shimoga');


insert into Region_Located
values (132,'shikaripurad','MESCOM','shimoga');


insert into Region_Located
values (221,'bagalkot','HESCOM','belagavi');


insert into Electricity_Consumer
values (2211,1,221,'adarsha','ada_road',919876543218);


insert into Electricity_Consumer
values (2221,1,222,'kumara','kua_road',919876543219);


insert into Consumption_Record
values ('feb',1312,131,0,0,0);


insert into Consumption_Record
```

```
values ('jan',1111,111,0,0,0);


insert into Rate
values ('feb',222,3);

insert into Rate
values ('feb',212,5);
```

Triggers

```
create trigger units_paid_change
before update on Consumption_Record
for each row
        set
new.Non_paid=old.Non_paid-new.Paid+old.Paid,new.Non_paid=new.Units_Consumed*(select
Average_rate
        from Rate
        where Rate.Region_id=old.Region_id and Rate.Month=old.Month)-new.Paid;

create trigger rate_change
after update on Rate
for each row
update Consumption_Record
set
Consumption_Record.Paid=Consumption_Record.Paid*new.Average_rate/old.Average_rate,Co
nsumption_Record.Non_paid=Consumption_Record.Non_paid*new.Average_rate/old.Average_
rate;
```

SQL Queries

Simple queries

- Extracting Branch Name in ascending order along with Company Name

  ```
  select b.Name,a.Short_Form
  from electricity_supply_companies a,electricity_supply_branches b
  ```

```
where a.Short_Form=b.C_Short_Form
order by Name;
```

- Extracting Region Name which are ending with ‘d’

```
select a.Region_name
from Region_Located a
where Region_name like '%d'
order by Region_name;
```

- Extracting all consumers Name ,their region and rate of those regions in the month of JAN in ascending order of consumers name.

```
select b.Con_Name,a.Region_name,c.Month,c.Average_rate
from Region_Located a,electricity_consumer b,Rate c
where a.Region_id=b.Region_id and b.Region_id=c.Region_id and c.Month='jan'
order by Con_Name ;
```

Complex Queries

- Extracting consumer who has units_consumed>20 and belonging to MESCOM

```
select Con_id ,Con_Name ,Con_meter_no,Region_id
from Electricity_Consumer
where Con_id in(
select Con_id
from Consumption_Record
inner join region_located on
Consumption_Record.Region_id=region_located.Region_id
where Units_Consumed>20 and region_located.Region_id in (
select Region_id
from region_located
where region_located.C_Short_Form='MESCOM'));
```

- Finding companies which have particular name in their database

```
select Short_Form,Name,Mobile_no
from electricity_supply_companies
where Short_Form in(
select electricity_supply_branches.C_Short_Form
from electricity_supply_branches
inner join region_located on
electricity_supply_branches.Name=region_located.BName
```

```
where region_located.BName in (
select BName
from region_located
inner join electricity_consumer on
electricity_consumer.Region_id=region_located.Region_id
where electricity_consumer.Region_id in(
select  electricity_consumer.Region_id
from electricity_consumer
where Con_Name='ravi')));
```

- Extracting Consumers and their regions who are paying particular rate

```
select Con_id ,Con_Name ,Con_meter_no,Region_id
from Electricity_Consumer
where Con_id in(
select Con_id
from Consumption_Record
inner join region_located on
Consumption_Record.Region_id=region_located.Region_id
where region_located.Region_id  in (
select region_located.Region_id
from region_located
inner join Rate on
region_located.Region_id=Rate.Region_id
where Rate.Region_id in (
select Rate.Region_id
from Rate
where Average_rate=6)));
```

- Extract total units consumed by each Company

```
select sum(Units_Consumed),region_located.C_Short_Form
from consumption_record,region_located
where consumption_record.Region_id=region_located.Region_id
 group by C_Short_Form;
```

- Extract total units consumed by each Region

```
 select consumption_record.Region_id,sum(Units_Consumed)
from consumption_record,region_located
where consumption_record.Region_id=region_located.Region_id
 group by Region_id
```

Conclusion

This System can store consumers billing ,units_consumed ,regions linked etc and can manage it

This system reduces the manual work by ⅔ rd so increases automation

This system has the capacity to automatically change it's paid and unpaid amount on changing units_consumed and Rate change and also balances amount to be given upon payment

Some limitations are it will not calculate previous unpaid amount due and extra charges.

Future work can be done upon increasing attributes related to billing and more dynamic automation.