

Project Team 12 - Electricity Billing System

Team Members:

- 1) Anchal Atlani
- 2) Anushka Kamalapurkar
- 3) Sagar Kulkarni
- 4) Gaurav Mahadik

Context:

The Electricity Billing System will give a single platform to customers through which they can pay and view their bills of any service provider listed in the system

Objective:

The main objective of the project is to develop a database system that will help a customer to register under a service provider, view and pay electricity bill. Apart from this a customer can also change service provider, upgrade to use solar net metering, view historic payments, peak and off peak hours and billing cycle.

Scope:

1. **Customer Registration:** Customer register under the system under a specific service provider for smart meter or solar net meter.
2. **View and Pay bills:**
 - a. Customer can pay and view historic bills.
 - b. Customer can check the unit consumption of the selected period.
 - c. Customer can view the off peak and peak hours for the area.
3. **Stop or Change the connection:** Customer can switch to other service provider and customer can deregister from the system.
4. **Admin can add Service Provider/ Customer:** Admin can add any new service provider or new customer.
5. **Admin modify the tariff rates:**
 - a. Admin can modify the tariff rates of any service provider.
 - b. Admin can modify the peak and off peak hours corresponding to any service provider in any area.

ENVIRONMENT SETUP:

We are using MySQL Database management System. For making connection to MySQL server we are using mysql-workbench-community-8.0.12-winx64 client. For user interface

and database connectivity we are using javax.swing api and jdbc.odbc 7.0 version. We will use Java 8 for our project.

HIGH LEVEL REQUIREMENTS:

Initial User Roles:

Role	Description
Customer	New customer can register under a service provider for the electricity billing system. Existing customer can view and pay electricity bills, change service provider, can see unit consumption, can see peak and off peak hours for any specific area.
Admin	System admin who can modify the tariff rate, Register/Deregister a customer, Register/Deregister a Service Provider.

Initial user story descriptions:

Story ID	Story Description
US1	As a customer I want to register under a service provider.
US2	As a registered customer I want to view bill of any selected month.
US3	As a registered customer I want to pay the bills.
US4	As a customer I want to view the tariff rates of any area offered by various service providers already listed.
US5	As a customer I want to change a service provider.
US6	As a customer I want to deregister.
US7	As an admin I want to change the tariff rates of service provider.
US8	As an admin I want to add a customer in the electricity billing system.
US9	As an admin I want to remove a customer from the electricity billing system.
US10	As an admin I want to add a service provider.

High level conceptual design:

Entities:

Customer
Admin
Service_Provider

Electricity_Bill
Tariff_Details

Relationships:

Customer has connection under a single Service_Provider.

Customer pays Electricity_Bill.

Customer changes a Service_Provider.

Admin updates Tariff_Details.

Service_Provider maintains area specific Tariff_Details.

Sprint 1

REQUIREMENTS:

Story ID	Story Description
US1	As a customer I want to register under a service provider.
US2	As a customer I want to login into the system.
US3	As a registered customer I want to view all electricity bills.
US4	As an admin I want to view all registered customers.
US5	As a customer I want to view number of units consumed.
US6	As a registered customer I want to view bill of any selected month.
US7	As a registered customer I want to pay the bills.
US8	As a customer I want to view the tariff rates of any area offered by various service providers already listed.
US9	As a customer I want to change a service provider.
US10	As a customer I want to deregister.
US11	As an admin I want to change the tariff rates of service provider.
US12	As an admin I want to add a customer in the electricity billing system.
US13	As an admin I want to remove a customer from the electricity billing system.
US14	As an admin I want to add a service provider.

CONCEPTUAL DESIGN

Entity: **Customer**

Attributes:

customer_id

login_email_id

login_password

ssn

name[composite]

first_name

last_name

address[composite]

address_line_1

address_line_2

city

state

zipcode

phone_number[multivalued]

service_provider_name

register_date

meter_type

customer_id- this customer id is generated whenever a user registers under a service provider.

Entity: **Electricity_Billing_Details**

Attributes:

electricity_bill_id

bill_generated_date

bill_due_date

billing_cycle_start_date

number_of_billing_days

payment_amount

amount_due

last_meter_reading

present_meter_reading

Note:

last_meter_reading and present_meter_reading are in kWH

Entity: **Admin**

Attributes:

login_id[Email id of Admin]

login_password

name[composite]

first_name

last_name

address[composite]

address_line_1

address_line_2

city

state

zipcode

phone_no[multivalued]

Relationship: **Customer** has **Electricity_Billing_Details**

Cardinality: One to Many

Participation:

Customer has partial participation

Electricity_Billing_Details have total participation

LOGICAL DESIGN

Table: **Customer**

Columns:

customer_id

login_email_id

login_password

ssn

first_name

last_name

address_line_1

address_line_2

city

state

country

zipcode

phone_number_primary

phone_number_secondary
service_provider_name
register_date
meter_type

Table: **Electricity_Billing_Details**

Columns:

electricity_bill_id
customer_id [foreign key;references customer_id of Customer table]
bill_generated_date
bill_due_date
billing_cycle_start_date
number_of_billing_days
payment_amount
amount_due
last_meter_reading
present_meter_reading

Note:

last_meter_reading and present_meter_reading are in kWH

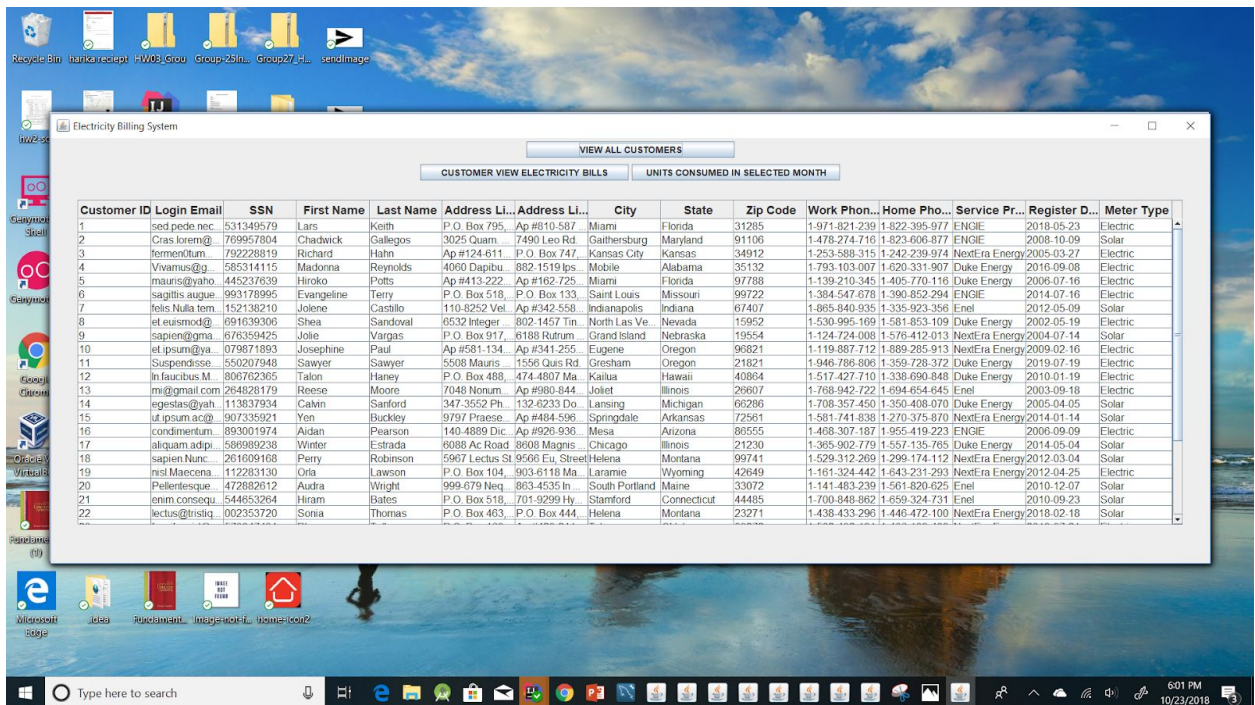
Table: **Admin**

Columns:

login_id[Email id of Admin]
login_password
first_name
last_name
address_line_1
address_line_2
city
state
zipcode
phone_no_primary
phone_no_secondary

Queries:

1. SELECT * FROM Customer;

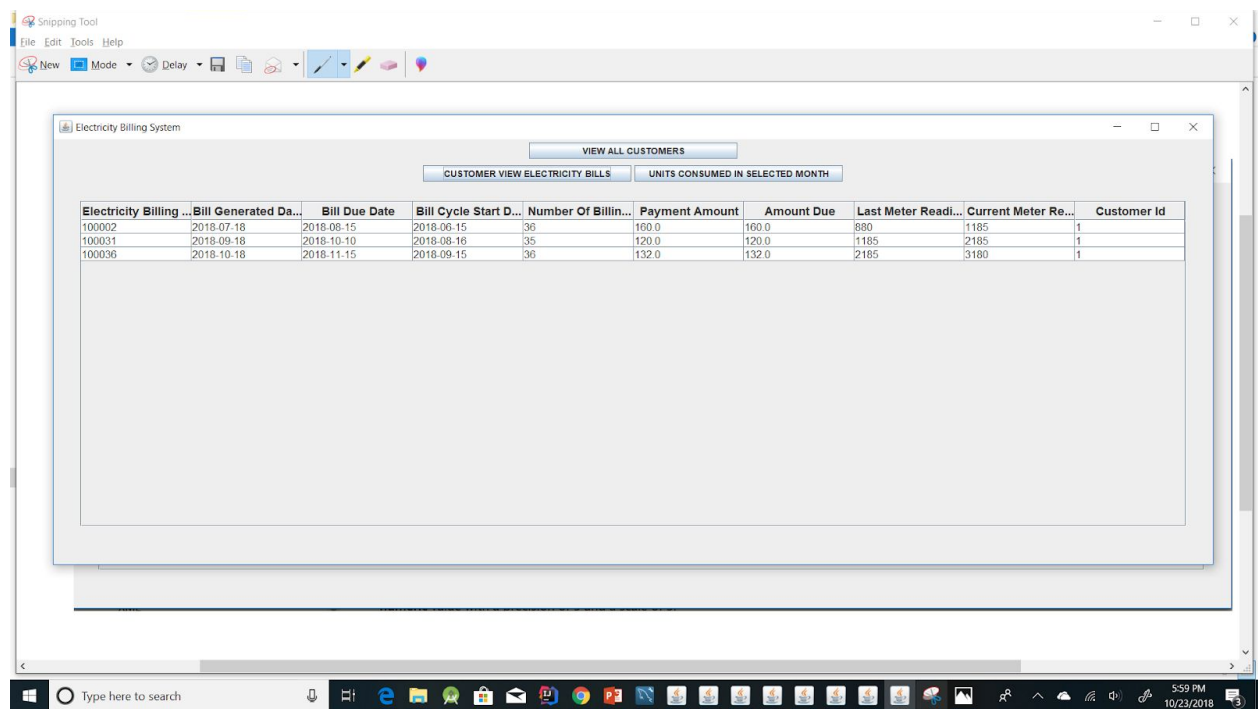


VIEW ALL CUSTOMERS

CUSTOMER VIEW ELECTRICITY BILLS UNITS CONSUMED IN SELECTED MONTH

Customer ID	Login Email	SSN	First Name	Last Name	Address Li...	Address Li...	City	State	Zip Code	Work Phon...	Home Pho...	Service Pr...	Register D...	Meter Type
1	sed.pede.nec	531349579	Lars	Keith	P.O. Box 795	Ap #810-587	Miami	Florida	31285	1.971-821-239	1.822-395-977	ENGIE	2018-05-23	Electric
2	Cras.lorem@...	769957804	Chadwick	Gallegos	3025 Quam	7490 Leo Rd	Gaithersburg	Maryland	91106	1.478-274-716	1.823-606-877	ENGIE	2008-10-09	Solar
3	fermen@um...	792228819	Richard	Hahn	Ap #124-611	P.O. Box 747	Kansas City	Kansas	34912	1.253-588-315	1.242-239-974	NextEra Energy	2005-03-27	Electric
4	Vivamus@g...	585314115	Madonna	Reynolds	4060 Dapibu	882-1519 Ips	Mobile	Alabama	35132	1.793-103-007	1.620-331-907	Duke Energy	2016-09-08	Electric
5	mauris@yaho	445237839	Hiroko	Potts	Ap #413-222	Ap #162-725	Miami	Florida	97788	1.139-210-345	1.405-770-116	Duke Energy	2006-07-16	Electric
6	sagittis augue	993178995	Evangeline	Terry	P.O. Box 518	P.O. Box 133	Saint Louis	Missouri	99722	1.384-547-678	1.390-852-294	ENGIE	2014-07-16	Electric
7	fole.Nulla.tem	152138210	Jolene	Castillo	110-3252 Vel	Ap #342-558	Indianapolis	Indiana	87407	1.865-840-935	1.335-923-356	Enel	2012-05-09	Solar
8	et.eusmod@	691636306	Shea	Sandoval	6532 Integer	802-1457 Tin	North Las Ve	Nevada	15952	1.530-995-169	1.581-853-109	Duke Energy	2002-05-19	Electric
9	sapien@gma	676359425	Jolie	Vargas	P.O. Box 917	6188 Rutum	Grand Island	Nebraska	19554	1.124-724-008	1.576-412-013	NextEra Energy	2004-07-14	Solar
10	et.ipsum@ya	079671893	Josephine	Paul	Ap #581-134	Ap #341-255	Eugene	Oregon	96821	1.119-887-712	1.889-285-913	NextEra Energy	2009-02-16	Electric
11	Suspendisse	550207948	Sawyer	Sawyer	5508 Mauris	1556 Qus Rd	Gresham	Oregon	21821	1.946-786-806	1.359-728-372	Duke Energy	2019-07-19	Electric
12	In.faucibus M	806702365	Talon	Haney	P.O. Box 488	474-4807 Ma	Kailua	Hawaii	40804	1.517-427-710	1.338-690-848	Duke Energy	2010-01-19	Electric
13	m@gmail.com	264628179	Reese	Moore	7048 Nonum	Ap #980-844	Joliet	Illinois	26607	1.768-942-722	1.694-654-645	Enel	2003-09-18	Electric
14	egestas@yah	113837994	Calvin	Sanford	347-3552 Ph	132-6233 Do	Lansing	Michigan	60286	1.708-357-450	1.350-408-070	Duke Energy	2005-04-05	Solar
15	ut.ipsum.ac@	907335921	Yen	Buckley	9797 Praese	Ap #484-596	Springdale	Arkansas	72561	1.581-741-838	1.270-375-870	NextEra Energy	2014-01-14	Solar
16	condimentum	893001974	Aidan	Pearson	140-4889 Dic	Ap #926-936	Mesa	Arizona	86555	1.468-307-187	1.955-419-223	ENGIE	2006-09-09	Electric
17	aliquam.adipi	586989238	Winter	Estrada	6088 Ac Road	8608 Magnis	Chicago	Illinois	21230	1.365-902-779	1.557-135-765	Duke Energy	2014-05-04	Solar
18	sapien.Nunc	261609168	Perry	Robinson	5967 Lectus	St 9566 Eu, Street	Helena	Montana	99741	1.529-312-269	1.299-174-112	NextEra Energy	2012-03-04	Solar
19	nsi.Maecena	112263130	Orla	Lawson	P.O. Box 104	903-6118 Ma	Laramie	Wyoming	42649	1.161-324-442	1.643-231-293	NextEra Energy	2012-04-25	Electric
20	Pelentesque	472882612	Audra	Wright	999-679 Neq	893-4535 In	South Portland	Maine	33072	1.141-483-239	1.561-820-625	Enel	2010-12-07	Solar
21	enim.consequ	444653264	Hiram	Isates	P.O. Box 518	701-9299 Hy	Stamford	Connecticut	44485	1.700-848-882	1.659-324-731	Enel	2010-09-23	Solar
22	lectus@tristiq	002353720	Sonia	Thomas	P.O. Box 463	P.O. Box 444	Helena	Montana	23271	1.438-433-296	1.446-472-100	NextEra Energy	2018-02-18	Solar

2. SELECT * FROM Electricity_Billing_Details where customer_id=1;



Electricity Billing System

VIEW ALL CUSTOMERS

CUSTOMER VIEW ELECTRICITY BILLS UNITS CONSUMED IN SELECTED MONTH

Electricity Billing ...	Bill Generated Da...	Bill Due Date	Bill Cycle Start D...	Number Of Billin...	Payment Amount	Amount Due	Last Meter Readi...	Current Meter Re...	Customer Id
100002	2018-07-18	2018-08-15	2018-06-15	36	160.0	160.0	880	1185	1
100031	2018-09-18	2018-10-10	2018-08-16	35	120.0	120.0	1185	2185	1
100036	2018-10-18	2018-11-15	2018-09-15	36	132.0	132.0	2185	3180	1

3.

```
SELECT Electricity_Billing_Details.customer_id,
Electricity_Billing_Details.electricity_bill_id,
Electricity_Billing_Details.billing_cycle_start_date, last_meter_reading,
Electricity_Billing_Details.present_meter_reading,
Electricity_Billing_Details.number_of_billing_days,
(Electricity_Billing_Details.present_meter_reading-Electricity_Billing_Details.last_meter_re
ading) as Units_Consumed FROM Electricity_Billing_Details where customer_id=1 AND
MONTH(billing_cycle_start_date)=9 AND YEAR(billing_cycle_start_date)=2018;
```

Recycle Bin

hanko redapt

HW03_Group

Group 25in...

Group27_H...

sendImage

hwPres

Galaxy Note...

Galaxy Note...

Google Chrome

Official Virtual...

Parashant (1)

Microsoft Edge

idea

PyCharm

IntelliJ

Home

Electricity Billing System

VIEW ALL CUSTOMERS

CUSTOMER VIEW ELECTRICITY BILLS

UNITS CONSUMED IN SELECTED MONTH

Customer ID	Electricity Bill Id	Bill Cycle Start Date	Last Month Meter Reading	Current Month Meter Rea...	Number Of Billing Days	Units Consumed
1	100036	2018-09-15	2185.54	3180.15	36	994.6100000000001

Type here to search

5:58 PM 10/23/2018

Sprint 2

Part 1: Refine requirements:

Story ID	Story Description
US1	As a customer I want to register under a service provider.
US2	As a customer I want to login into the system.
US3	As a registered customer I want to view all electricity bills.
US4	As an admin I want to view all registered customers.
US5	As a customer I want to view number of units consumed.
US6	As a registered customer I want to view bill of any selected month.
US7	As a customer I want to view the tariff rates of any area offered by various service providers already listed.
US8	As a customer I want to change a service provider.
US9	As a customer I want to deregister.
US10	As an admin I want to change the tariff rates of service provider.
US11	As an admin I want to add a customer in the electricity billing system.
US12	As an admin I want to remove a customer from the electricity billing system.
US13	As an admin I want to add a service provider.
US14	As a registered customer I want to pay the bills.

Part 2: Conceptual design

Entity: **Customer**

Attributes:

- customer_id
- login_email_id
- login_password
- ssn
- name[composite]
 - first_name
 - last_name
- address[composite]
 - address_line_1
 - address_line_2
 - zipcode
 - state
 - city
- phone_number[multivalued]
- service_provider_name
- register_date
- meter_type

Entity: **Electricity_Billing_Details**

Attributes:

- electricity_bill_id
- bill_generated_date
- bill_due_date
- billing_cycle_start_date
- number_of_billing_days
- payment_amount
- amount_due

Entity: **Admin**

Attributes:

- login_id[Email id of Admin]
- login_password
- employee_number
- name[composite]
 - first_name
 - last_name
- address[composite]
 - address_line_1
 - address_line_2
 - zipcode
 - state
 - city
- phone_no[multivalued]

Entity: **Service_Provider**

Attributes:

- service_provider_name
- license_number
- license_expiry
- address[composite]
 - address_line_1
 - address_line_2
 - zipcode
 - state
 - city
- head_office_number
- helpline_number
- email_contact

Entity: **Tariff_Details**

Attributes:

- zip_code
- provider_name
- state
- city
- peak_time_charge
- off_peak_charge
- peak_time
- off_peak_time

Entity: **Meter_Reading**

Attribute:

meter_number
peak_hour_reading
off_peak_hour_reading
meter_reading_date
total_current_reading
total_previous_reading

Relationship: **Customer** has **Meter_Reading**

Cardinality: One to One

Participation:

Customer has partial participation

Meter_Reading have total participation

Relationship: **Meter_Reading** generates **Electricity_Billing_Details**

Cardinality: One to One

Participation:

Customer has partial participation

Meter_Reading have total participation

Relationship: **Service_Provider** has **Tariff_Details**

Cardinality: One to many

Participation:

Service_Provider has partial Participation

Tariff_Details have partial Participation

Relationship: **Customer** views **Tariff_Details**

Cardinality: Many to Many

Participation:

Customer has partial participation

Tariff_Details have partial_participation

Relationship: **Admin** adds **Customer**

Cardinality: One to Many

Participation:

Admin has partial Participation

Customer has total participation

Part 3: Logical design

Table: **Customer**

Columns:

- customer_id
- login_email_id
- login_password
- ssn
- first_name
- last_name
- address_line_1
- address_line_2
- zip_code [foreign key; references zip_code of ZipCode_Details Table]
- phone_number_primary
- phone_number_secondary
- service_provider_name
- register_date
- meter_type

Primary Key Justification: customer_id of each customer will be unique, and it will be assigned during registration of customer in system.

Normal form:

Highest normalization level: 4NF

Table: **Admin**

Columns:

employee_id
login_id[Email id of Admin]
login_password
first_name
last_name
address_line_1
address_line_2
zip_code [foreign key; references zip_code of ZipCode_Details Table]
phone_no_primary
phone_no_secondary

Primary Key Justification: employee_id of the system admin will be unique.

Normal form:

Highest normalization level:4NF

Table: **Service_Provider**

Column:

provider_name
license_number
office_Address_line_1
office_address_line_2
zip_code[foreign key; references zip_code of ZipCode_Details Table]
head_Office_Number
helpline_Number
email_contact

Primary Key Justification: Name of Service provider is unique

Normal form:

Highest normalization level:4NF

Table: **Tariff_Details**

Columns:

service_provider_name[foreign key; references provider_name in Service_Provider]

zip_code

peak_time_charge

off_peak_charge

peak_time

off_peak_time

Foreign key mapping:

We have decided to map the primary key of Service_Provider as the foreign key in Tariff_Details as there is a one to many relationship in the conceptual design.

Primary Key Justification:

The zip_code and service_provider_name uniquely determines the tariff details of each for any area of specific service provider.

Normal Form:

Highest normalization level:4NF

Table: **Meter_Reading**

Columns:

meter_number

peak_hour_reading

off_peak_hour_reading

meter_reading_date

total_current_reading

total_previous_reading

customer_id[foreign key: references customer_id of Customer Table]

Composite Primary Key:

We have a composite primary key here as only those two columns can uniquely identify the table.

Normal Form:

Highest normalization level: 2NF

Justification: We need composite key here as this is used to record meter readings of each customer for each billing-cycle.

Table: Electricity_Billing_Details

Columns:

electricity_bill_id

meter_number [foreign key;references meter_number of Meter_Reading table]

bill_generated_date

bill_due_date

billing_cycle_start_date

number_of_billing_days

payment_amount

amount_due

Primary Key Justification: electricity_bill_id will be unique, as this will be assign whenever new bill is being added into the system.

Normal Form:

Highest normalization level:4NF

Table: Zipcode_Details

Columns:

zip_code

city

state

Primary Key Justification: zip_code will be unique for every region and that uniquely determines the state and city and here as or system is for United States so we have not included country column.

Normal Form:

Highest normalization level:4NF

Queries:

1. Register a Customer

Before

Select * from Customer;

Electricity Billing System

Customer View

Admin View

Customer View

1. Register Into System

2. View Electricity Bills

3. Look up ServiceProvider

4. View Units Consumed in a month

5. Login

6. Pay Bill

7. View Tariff Detail of Any Service Provider

Enter the Query

Select * from Customer;

Data Retrieved Successfully

Execute

customer...	login_em...	login_pas...	ssn	first_name	last_name	address_li...	address_li...	zip_code	phone_no...	phone_no...	service_p...	register_d...	meter_type
1	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	971-821-2399	1-822-395-9...	ENGIE	2018-05-23	Electric
2	Cras.lorem...	abcd	769957804	Chadwick	Gallegos	3025 Quam...	7490 Leo Rd.	91106	478-274-7169	1-823-606-8...	ENGIE	2008-10-09	Solar
3	fermen0tum...	zsjdk@65	792228819	Richard	Hahn	Ap #124-611...	P.O. Box 74...	34912	253-588-3159	1-242-239-9...	NextEra Ene...	2005-03-27	Electric
4	Vivamus@g...	zsjdk@65	585314115	Madonna	Reynolds	4060 Dapibu...	882-1519 Ips.	35132	793-103-0076	1-620-331-9...	Duke Energy	2016-09-08	Electric
5	mauris@yah...	zsjdk@65	445237639	Hiroko	Potts	Ap #413-222	Ap #162-725	97788	139-210-3455	1-405-770-1...	Duke Energy	2006-07-16	Electric
6	hdjkasd@gm...	fsjhp	458457896	kavya	poland	201 tri street		97788	234-234-1323		NextEra Ene...	2010-10-10	Solar
40	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-05-23	Electric
41	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-05-23	Electric
43	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-05-23	Electric

After insertion:

INSERT INTO Customer VALUES

```
(44,'bb.smith@gmail.com','bs@65','531349569','Bob','Smith',  
'P.O. Box 795, 3438 Consectetuer Street','Ap #810-587 Odio. St.',  
'31285','1-971-821-239','1-822-395-977','ENGIE','2018-11-11','Solar');
```

Electricity Billing System

Customer View Admin View

Customer View

Enter the Query

```
INSERT INTO Customer VALUES  
(44,'bb.smith@gmail.com','bs@65','531349569','Bob','Smith',  
'P.O. Box 795, 3438 Consectetuer Street','Ap #810-587 Odio. St.',  
'31285','1-971-821-239','1-822-395-977','ENGIE','2018-11-11','Solar');
```

Query Executed Successfully

Execute

customer...	login_em...	login_pas...	ssn	first_name	last_name	address_li...	address_li...	zip_code	phone_no...	phone_no...	service_p...	register_d...	meter_type
1	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	971-821-2399	1-822-395-9...	ENGIE	2018-05-23	Electric
2	Cras.lorem...	abcd	769957804	Chadwick	Gallegos	3025 Quam...	7490 Leo Rd...	91106	478-274-7169	1-823-606-8...	ENGIE	2008-10-09	Solar
3	fermentum...	zsjdk@65	792228819	Richard	Hahn	Ap #124-611...	P.O. Box 74...	34912	253-588-3159	1-242-239-9...	NextEra Ene...	2005-03-27	Electric
4	Vivamus@g...	zsjdk@65	585314115	Madonna	Reynolds	4060 Dapibu...	882-1519 Ips...	35132	793-103-0076	1-620-331-9...	Duke Energy	2016-09-08	Electric
5	mauris@yah...	zsjdk@65	445237639	Hiroko	Potts	Ap #413-222...	Ap #162-725...	97788	139-210-3455	1-405-770-1...	Duke Energy	2006-07-16	Electric
6	hdkasdg@gm...	fsjhb	458457896	kavya	poland	201 tri street		97788	234-234-1323		NextEra Ene...	2010-10-10	Solar
40	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-05-23	Electric
41	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-05-23	Electric
43	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-05-23	Electric
44	bb.smith@g...	bs@65	531349569	Bob	Smith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-11-11	Solar

2. Customer changes Service Provider.

Before: Here customer_id we are taking is 43.

Electricity Billing System

Customer View Admin View

Customer View

Enter the Query

Select service_provider_name from Customer where customer_id='43';

Data Retrieved Successfully

Execute

service_provider_name

ENGIE

After updating:

update Customer set service_provider_name='Duke Energy' where customer_id='43';

Electricity Billing System

Customer View Admin View

Customer View

Enter the Query

update Customer set service_provider_name='Duke Energy' where customer_id='43';

Query Executed Successfully

Execute

customer...	login_em...	login_pas...	ssn	first_name	last_name	address_li...	address_li...	zip_code	phone_no...	phone_no...	service_p...	register_d...	meter_type
1	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	971-821-2399	1-822-395-9...	ENGIE	2018-05-23	Electric
2	Cras lorem...	abcd	769957804	Chadwick	Gallegos	3025 Quam...	7490 Leo Rd	91106	478-274-7169	1-823-606-8...	ENGIE	2008-10-09	Solar
3	fermen0tum...	zsjdk@65	792228819	Richard	Hahn	Ap #124-611...	P.O. Box 74...	34912	253-588-3159	1-242-239-9...	NextEra Ene...	2005-03-27	Electric
4	Vivamus@g...	zsjdk@65	585314115	Madonna	Reynolds	4060 Dapibu...	882-1519 Ips	35132	793-103-0076	1-620-331-9...	Duke Energy	2016-09-08	Electric
5	mauris@yah...	zsjdk@65	445237639	Hiroko	Potts	Ap #413-222	Ap #162-725...	97788	139-210-3455	1-405-770-1...	Duke Energy	2006-07-16	Electric
6	hdkasdg@gm...	ifshjb	458457896	kavya	poland	201 tri street		97788	234-234-1323		NextEra Ene...	2010-10-10	Solar
40	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-05-23	Electric
41	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	ENGIE	2018-05-23	Electric
43	sed.pede.ne...	zsjdk@65	531349579	Lars	Keith	P.O. Box 79...	Ap #810-587...	31285	1-971-821-2...	1-822-395-9...	Duke Energy	2018-05-23	Electric

3. Look up Service Providers available;

Select * from Service_Provider;

Electricity Billing System

Customer View

Admin View

Customer View

1. Register Into System

2. View Electricity Bills

3. Look up ServiceProvider

4. View Units Consumed in a month

5. Login

6. Pay Bill

7. View Tariff Detail of Any Service Provider

Enter the Query

Select * from Service_Provider;

Data Retrieved Successfully

Execute

provider_name	license_number	office_address_line_1	office_address_Line_2	zip_code	head_office_number	helpline_number
Duke Energy	12345678	201 Tryon Street		35132	(526)548-4569	(526)896-1236
Engie	54513257	78 street		34912	(524)245-8968	(524)478-2145
NextEra Energy	78945612	601 Travis Street		31285	(305)246-1300	(321)457-9634

4. Customer can view Tariff_Details of any Service Provider of any specific area.

Select * from Tariff_Details where provider_name='Duke Energy' and zip_code='97788';

Electricity Billing System

Customer View Admin View

Customer View

Enter the Query

Select * from Tariff_Details where provider_name='Duke Energy' and zip_code='97788';

1. Register Into System
2. View Electricity Bills
3. Look up ServiceProvider
4. View Units Consumed in a month
5. Login
6. Pay Bill
7. View Tariff Detail of Any Service Provider

Data Retrieved Successfully

Execute

zip_code	provider_name	peak_time_charge	off_peak_time_charge	peak_time	off_peak_time
97788	Duke Energy	1.34	0.96	10:00 AM-10:PM	10:00 AM-10: PM

Sprint -3

Part 1: Refine requirements:

Story ID	Story Description
US1	As a customer I want to register under a service provider.
US2	As a customer I want to login into the system.
US3	As a registered customer I want to view all electricity bills.
US4	As an admin I want to view all registered customers.
US5	As a customer I want to view number of units consumed.
US6	As a registered customer I want to view bill of any selected month.
US7	As a customer I want to view the tariff rates of any area offered by various service providers already listed.
US8	As a customer I want to change a service provider.
US9	As a customer I want to deregister.
US10	As an admin I want to change the tariff rates of service provider.
US11	As an admin I want to add a customer in the electricity billing system.
US12	As an admin I want to remove a customer from the electricity billing system.
US13	As an admin I want to add a service provider.
US14	As a registered customer I want to pay the bills.

Part 2: Conceptual design

Entity: **Customer**

Attributes:

- login_email_id
- login_password
- ssn
- name[composite]
 - first_name
 - last_name
- address[composite]
 - address_line_1
 - address_line_2
 - zipcode
 - state
 - city
- phone_number[multivalued]
- service_provider_name
- register_date
- meter_type

Entity: **Electricity_Billing_Details**

Attributes:

- bill_generated_date
- bill_due_date
- billing_cycle_start_date
- number_of_billing_days
- payment_amount
- amount_due

Entity: **Admin**

Attributes:

- login_id[Email id of Admin]
- login_password
- employee_number
- name[composite]
 - first_name
 - last_name
- address[composite]
 - address_line_1
 - address_line_2
 - zipcode
 - state
 - city
- phone_no[multivalued]

Entity: **Service_Provider**

Attributes:

- service_provider_name
- license_number
- license_expiry
- address[composite]
 - address_line_1
 - address_line_2
 - zipcode
 - state
 - city
- head_office_number
- helpline_number
- email_contact

Entity: **Tariff_Details**

Attributes:

- zip_code
 - state
 - city
- provider_name
- peak_time_charge
- off_peak_charge
- peak_time
- off_peak_time

Entity: **Meter_Reading**

Attribute:

meter_number
peak_hour_reading
off_peak_hour_reading
meter_reading_date
total_current_reading
total_previous_reading

Entity: **Payment**

Attributes:

payment_date
amount_due
card_number
cvv
card_expiry
payment_amount

Relationship: **Customer** has **Meter_Reading**

Cardinality: One to One

Participation:

Customer has partial participation

Meter_Reading have total participation

Relationship: **Meter_Reading** generates **Electricity_Billing_Details**

Cardinality: One to One

Participation:

Customer has partial participation

Meter_Reading have total participation

Relationship: **Service_Provider** has **Tariff_Details**

Cardinality: One to many

Participation:

Service_Provider has partial Participation

Tariff_Details have partial Participation

Relationship: **Customer** views **Tariff_Details**

Cardinality: Many to Many

Participation:

Customer has partial participation

Tariff_Details have partial participation

Relationship: **Admin** adds **Customer**

Cardinality: One to Many

Participation:

Admin has partial Participation

Customer has total participation

Relationship: **Customer** makes **Payment**

Cardinality: One to Many

Participation:

Customer has partial Participation

Payment has total participation

Relationship: **Electricity_Billing_Details** has **Payment**

Cardinality: Many to One

Participation:

Payment has partial Participation

Electricity_Billing_Details has total participation

Justification : Here the cardinality is many to one because the customer is provided with the facility to pay the present bill along with any past bills and the outstanding amount is added to the present amount. So the cardinality of this relationship is Many to One.

Part 3: Logical design

Table: **Customer**

Columns:

customer_id
login_email_id
login_password
ssn
first_name
last_name
address_line_1
address_line_2
zip_code [foreign key; references zip_code of ZipCode_Details Table]
phone_number_primary
phone_number_secondary
service_provider_name[foreign key; references provider_name of Service_Provider Table]
register_date
meter_type

Primary Key Justification: customer_id of each customer will be unique, and it will be generated during registration of customer in system.

Foreign key mapping:

We have decided to map the primary key of Service_Provider as the foreign key in Customer as there is a one to one relationship in the conceptual design.

Normal form:

Highest normalization level:4NF

Table: **Admin**

Columns:

employee_id
login_id[Email id of Admin]
login_password
first_name
last_name
address_line_1
address_line_2
zip_code [foreign key; references zip_code of ZipCode_Details Table]
phone_no_primary

phone_no_secondary

Primary Key Justification: employee_id of the system admin will be unique.

Foreign key mapping:

We have decided to map the primary key of Zipcode_Details as the foreign key in Admin.

Normal form:

Highest normalization level:4NF

Table: **Service_Provider**

Column:

provider_name
license_number
office_Address_line_1
office_address_line_2
zip_code[foreign key; references zip_code of ZipCode_Details Table]
head_Office_Number
helpline_Number
email_contact

Primary Key Justification: Name of Service provider is unique

Foreign key mapping:

We have decided to map the primary key of Zipcode_Details table as the foreign key in Service_Provider as there is a one to many relationship in the conceptual design.

Normal form:

Highest normalization level:4NF

Table: **Tariff_Details**

Columns:

service_provider_name[foreign key; references provider_name in Service_Provider]

zip_code
peak_time_charge
off_peak_charge
peak_time
off_peak_time

Primary Key Justification:

The zip_code and service_provider_name uniquely determines the tariff details of each for any area of specific service provider.

Foreign key mapping:

We have decided to map the primary key of Service_Provider as the foreign key in Tariff_Details as there is a one to many relationship in the conceptual design.

Normal Form:

Highest normalization level: 4NF

Table: Meter_Reading

Columns:

meter_reading_id

meter_number

peak_hour_reading

off_peak_hour_reading

meter_reading_date

total_current_reading

total_previous_reading

customer_id[foreign key: references customer_id of Customer Table]

Primary Key:

We have meter_reading_id as primary key here as every time when admin enters the meter reading of any particular month then this will uniquely determine the entity.

Foreign key mapping:

We have decided to map the primary key of Customer table as the foreign key in Meter_Reading as there is a one to many relationship in the conceptual design.

Normal Form:

Highest normalization level: 4NF

Table: Electricity_Billing_Details

Columns:

electricity_bill_id

meter_reading_id [foreign key; references meter_reading_id of Meter_Reading table]

bill_generated_date

bill_due_date

billing_cycle_start_date

number_of_billing_days

payment_amount

amount_due

payment_id[foreign key;references payment_id of Payment table]

Primary Key Justification: electricity_bill_id will be unique, as this will be assign whenever new bill is being added into the system.

Foreign key mapping:

- a. We have decided to map the primary key of Payment table as the foreign key in Electricity_Billing_Details as there is a one to many relationship in the conceptual design.
- b. The other foreign key is meter_reading_id, which is the Primary Key of Meter_Reading table, this will uniquely determine the bill specific to each meter reading entered.

Normal Form:

Highest normalization level:4NF

Table: **Zipcode_Details**

Columns:

zip_code

city

state

Primary Key Justification: zip_code will be unique for every region and that uniquely determines the state and city and here as or system is for United States so we have not included country column.

Normal Form:

Highest normalization level:4NF

Table: **Payment**

Columns:

payment_id

payment_date

amount_due

card_number

cvv

customer_id[foreign key;references customer_id of Customer table]

card_expiry

Payment_amount

Primary Key Justification: payment_id will be the primary key as it will uniquely identify a row in the table.

Foreign key mapping:

We have decided to map the primary key of Customer table as the foreign key in Payment as there is a one to many relationship in the conceptual design.

Normal Form:

Highest normalization level:4NF

Part 4: Indexes

Indexes

Table :Admin

Indexes:

The following are the indexes of the movie table

- 1.PRIMARY Index on Column login_id which will be a clustered index.
- 2.employee_number on Column employee_number which will be a non-clustered index.
- 3.zip_code on Column zip_code which will be a non-clustered index.
- 4.admin_name Index on Columns first_name and last_name which is a non-clustered index.

Justification :

The default index has already been created as PRIMARY index made on login_id as it is the primary key in the table Admin.

The default index has already been created as employee_number index made on employee_number as it is the unique key in the table Admin.

The default foreign key index has already been created as zip_code index made on zip_code as it is the foreign key in the table Admin.

The admin_name index has been created on columns first_name and last_name as these columns has been used in many queries and this index will speed up the performance of the queries and increase the efficiency by providing the result quicker than it would have been without the admin_name index.

Table :Customer

Indexes:

The following are the indexes of the movie table.

1. PRIMARY Index on Column customer_id which will be a clustered index.
2. ssn_UNIQUE on Column ssn which will be a non-clustered index.
3. login_email_id_UNIQUE on Column login_email_id which will be a non-clustered index.
4. zip_code on Column zip_code which will be a non-clustered index.
5. service_provider_name_idx Index on Columns service_provider_name which is a non-clustered index.
6. customer_name Index on Columns first_name and last_name which is a nonclustered index.

Justification :

The default index has already been created as PRIMARY index made on customer_id as it is the primary key in the table Customer.

The default index has already been created as ssn_UNIQUE index made on ssn as it is the unique key in the table Customer.

The default index has already been created as login_email_id_UNIQUE index made on login_email_id as it is the unique key in the table Customer.

The default foreign key index has already been created as zip_code index made on zip_code as it is the foreign key in the table Customer.

The default foreign key index has already been created as service_provider_name_idx index made on service_provider_name as it is the foreign key in the table Customer.

The customer_name index has been created on columns first_name and last_name as these columns have been used in many queries and this index will speed up the performance of the queries and increase the efficiency by providing the result quicker than it would have been without the customer_name index.

Table :Electricity_Billing_Details

Indexes:

The following are the indexes of the movie table

1. PRIMARY Index on Column electricity_billing_id which will be a clustered index.
2. Meter_reading_id index on Column meter_reading_id which will be a non-clustered index.
3. payment_id index on Column payment_id which will be a non-clustered index.

Justification :

The default index has already been created as PRIMARY index made on electricity_billing_id as it is the primary key in the table Electricity_Billing_Details.

The default foreign key index has already been created as meter_reading_id index made on meter_reading_id as it is the foreign key in the table Electricity_Billing_Details.

The default foreign key index has already been created as payment_id index made on payment_id as it is the foreign key in the table Electricity_Billing_Details.

Table :Meter_Reading

Indexes:

The following are the indexes of the movie table

1. PRIMARY Index on Column meter_reading_id which will be a clustered index.
2. customer_id on Column customer_id which will be a non-clustered index.

Justification :

The default index has already been created as meter_reading_id index made on meter_reading_id as it is the primary key in the table Meter_Reading.

The default foreign key index has already been created as customer_id index made on customer_id as it is the foreign key in the table Meter_Reading.

Table :Service_Provider

Indexes:

The following are the indexes of the movie table

- 1.PRIMARY Index on Column provider_name which will be a clustered index.
2. license_number_UNIQUE Index on Column license_number which will be a non-clustered index.
- 3.zip_code_idx on Column zip_code which will be a non-clustered index.

Justification :

The default index has already been created as provider_name index made on provider_name as it is the primary key in the table Service_Provider.

The default index has already been created as license_number_UNIQUE index made on license_number as it is the unique key in the table Service_Provider.

The default foreign key index has already been created as zip_code_idx index made on zip_code as it is the foreign key in the table Service_Provider.

Table :Tariff_Details

Indexes:

The following are the indexes of the movie table

- 1.PRIMARY Index on Column zip_code and provider_name which will be a clustered index.
2. provider_name Index on Column provider_name which will be a clustered index.

Justification :

The default index has already been created as PRIMARY Index made on zip_code as it is the primary key in the table Tariff_Details.

The foreign key index has already been created as provider_name Index made on provider_name as it is the foreign key in the table Tariff_Details.

Table :Zipcode_Details

Indexes:

The following are the indexes of the movie table

1. PRIMARY Index on Column zip_code which will be a clustered index.

Justification :

The default index has already been created as PRIMARY Index made on zip_code as it is the primary key in the table Zipcode_Details.

Table :Payment

Indexes:

The following are the indexes of the movie table

- 1.PRIMARY Index on Column payment_id which will be a clustered index.
2. customer_id index on Column customer_id which will be a non-clustered index.

Justification :

The default index has already been created as PRIMARY Index made on payment_id as it is the primary key in the table Payment.

The default foreign key index has already been created as customer_id Index made on customer_id as it is the foreign key in the table Payment.

Part 5: Stored programs and views

STORED PROCEDURE:

a.

Procedure Name: generate_electricity_bill

Input Parameter:

1. meter_num int(10)- This is the meter_number of a particular customer.
2. peak_hr_reading double- This is the reading during peak hours.
3. off_peak_hr_reading double -This is the reading during off peak hours.
4. customer_id - This is the customer_id of the customer for which the admin will enter the details.

Description:

Whenever admin will enter the meter_reading of current month specific to any customer, electricity bill should get generated and payment amount should also get calculated using tariff rates of the service provider which we will get from Tariff_Details table.

Assumption: Here whenever we are adding entry in Meter Reading we are assuming that meter reading of the current month i.e month in current date is being added.

Before Procedure Run:

Meter_Reading table view for customer_id=5:

MySQL Workbench

Electricity Billing System

Customer View Admin View Procedure Calls

Admin View

Enter the Query

Select * from Meter_Reading where customer_id=5;

Execute

meter_number	peak_hour_reading	off_peak_hour_reading	meter_reading_date	total_current_reading	total_previous_reading	customer_id	meter_reading_id
1005	98.0	61.0	2018-09-20	369.0	230.0	5	9
1005	75.0	35.0	2018-10-22	494.0	389.0	5	10
1005	112.0	64.0	2018-11-21	670.0	494.0	5	11

PERFORMANCE

Dashboard

Performance Reports

Performance Schema

SCHEMAS

Filter objects

classicmode

Electricity_Billing_De

Tables

Admin

Customer

Electricity_Billing

Meter_Reading

Payment

Service_Provider

Tariff_Details

Zipcode_Details

Views

Stored Procedures

Information

No object selected

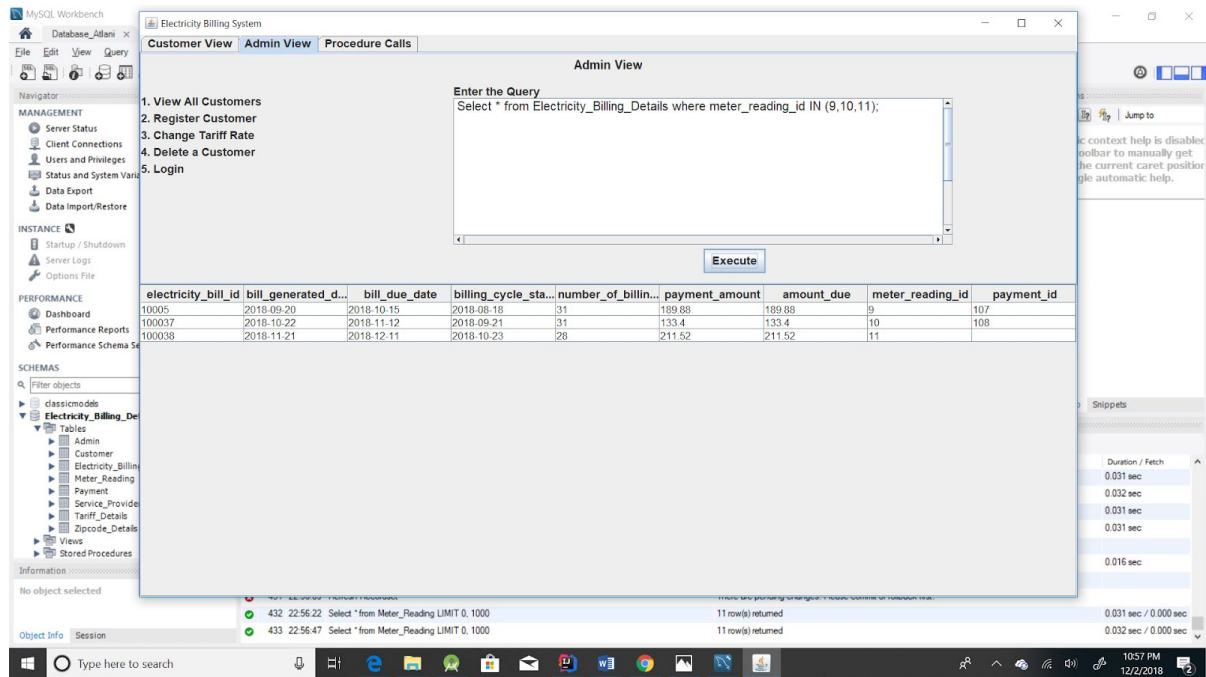
Object Info Session

432 22:56:22 Select * from Meter_Reading LIMIT 0, 1000 11 row(s) returned 0.031 sec / 0.000 sec

433 22:56:47 Select * from Meter_Reading LIMIT 0, 1000 11 row(s) returned 0.032 sec / 0.000 sec

10:57 PM 12/2/2018

View of Electricity_Billing_Details:



MySQL Workbench - Electricity Billing System

Customer View | Admin View | Procedure Calls

Admin View

Enter the Query

```
Select * from Electricity_Billing_Details where meter_reading_id IN (9,10,11);
```

Execute

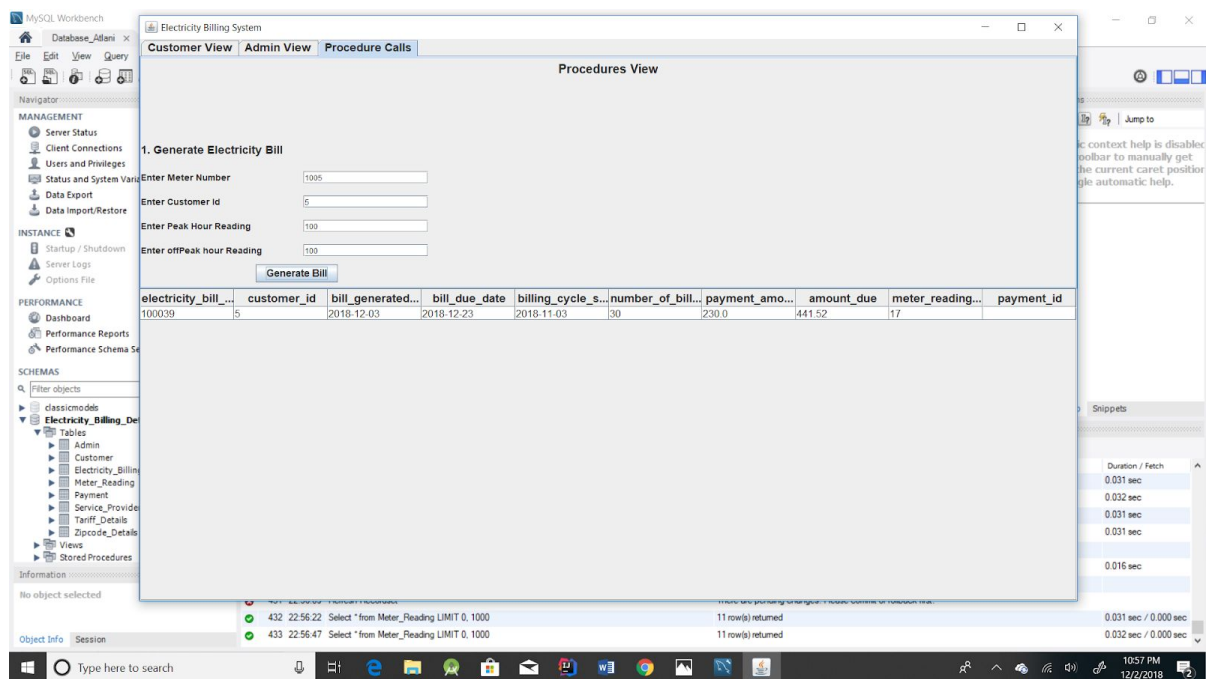
electricity_bill_id	bill_generated_d...	bill_due_date	billing_cycle_sta...	number_of_billin...	payment_amount	amount_due	meter_reading_id	payment_id
10005	2018-09-20	2018-10-15	2018-08-18	31	189.88	189.88	9	107
100037	2018-10-22	2018-11-12	2018-09-21	31	133.4	133.4	10	108
100038	2018-11-21	2018-12-11	2018-10-23	28	211.52	211.52	11	

Object Info: No object selected

Session: Type here to search

1057 PM 12/2/2018

After Procedure run, new bill generated when we click generate bill



MySQL Workbench - Electricity Billing System

Customer View | Admin View | Procedure Calls

Procedures View

1. Generate Electricity Bill

Enter Meter Number: 1005

Enter Customer Id: 5

Enter Peak Hour Reading: 100

Enter offPeak hour Reading: 100

Generate Bill

electricity_bill_id	customer_id	bill_generated...	bill_due_date	billing_cycle_s...	number_of_bill...	payment_amo...	amount_due	meter_reading...	payment_id
100039	5	2018-12-03	2018-12-23	2018-11-03	30	230.0	441.52	17	

Object Info: No object selected

Session: Type here to search

1057 PM 12/2/2018

View of Meter_Reading table for customer id 5 after procedure run:

MySQL Workbench - Electricity Billing System

Admin View

Enter the Query

Select * from Meter_Reading where customer_id =5;

Execute

meter_number	peak_hour_reading	off_peak_hour_reading	meter_reading_date	total_current_reading	total_previous_reading	customer_id	meter_reading_id
1005	98.0	61.0	2018-09-20	389.0	230.0	5	9
1005	75.0	35.0	2018-10-22	494.0	389.0	5	10
1005	112.0	64.0	2018-11-21	670.0	494.0	5	11
1005	100.0	100.0	2018-12-03	870.0	670.0	5	17

Object Info Session

Object here to search

10:58 PM 12/2/2018

b.

Procedure Name: get_amount_due

Input Parameter:

cust_id bigint(20)-- This is the customer id of customer for which we need to find the amount due.

Output Parameter:

amountDue double - This is return parameter as this will be used in above procedure for calculation of cumulative amount due.

Description:

Procedure for getting amount not paid in previous bill. This procedure is used to get the amount due of any customer id who has not paid the bill of the last month. This is being used inside the above procedure.

Above procedure called inside the above procedure.

TRIGGER:

a.

Trigger Name:after_payment_insert

Description: This trigger is there because as soon as Customer Pays the Bill the payment id should get updated in Electricity_Billing_Details table.

VIEWS:

a.

View Name: customer_electricity_bills

Description: This view is used by admin whenever he wants to view all bills of any particular customer.

Create View customer_electricity_bills As

Select e.electricity_bill_id,m.customer_id, e.bill_generated_date,

e.bill_due_date,e.billing_cycle_start_date,

e.number_of_billing_days,e.payment_amount,e.amount_due,e.meter_reading_id,e.payment_id

from Electricity_Billing_Details e INNER JOIN Meter_Reading m using(meter_reading_id)

INNER JOIN

Customer c using(customer_id);

The screenshot displays the MySQL Workbench interface for the 'Electricity Billing System'. The 'Admin View' is active, showing a menu with five options: '1. View All Customers', '2. Register Customer', '3. Change Tariff Rate', '4. Delete a Customer', and '5. Login'. Below the menu is a text area for entering queries, containing the SQL statement: 'Select * from customer_count_under_service_providers;'. An 'Execute' button is located below the query area. At the bottom, a table displays the results of the query:

Provider Name	Total Number of Customers
Duke Energy	3
ENGIE	2
NextEra Energy	2

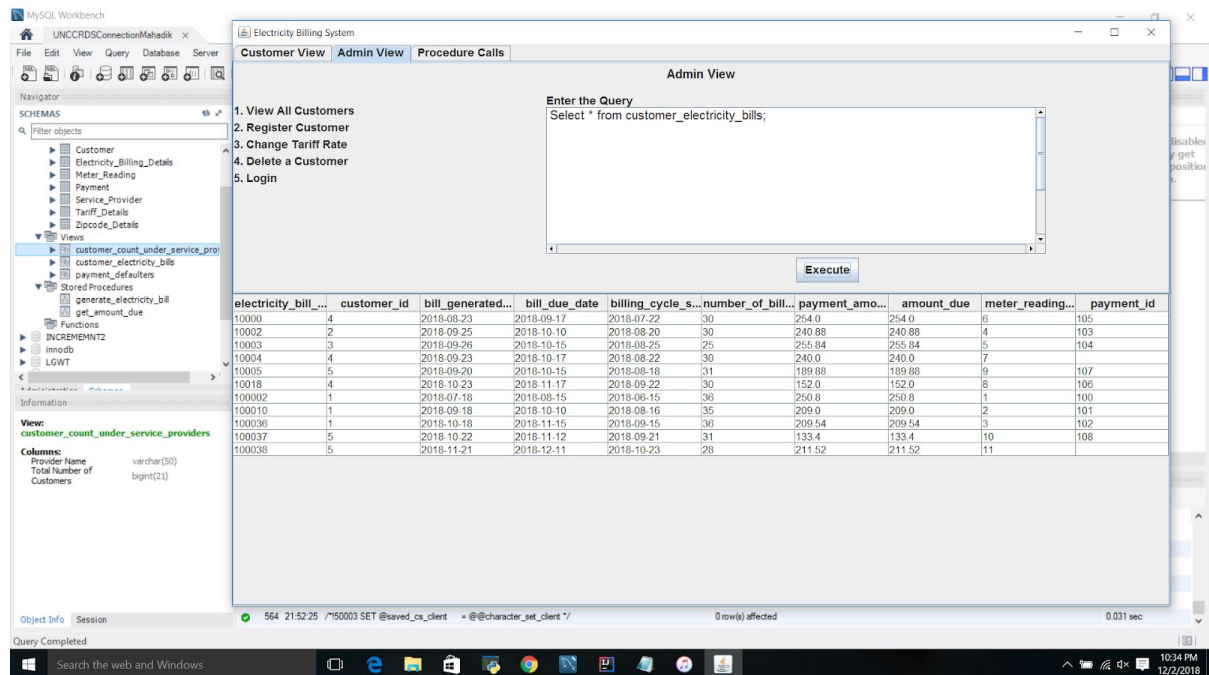
The sidebar on the left shows the 'SCHEMAS' tree with various database objects like 'Customer', 'Electricity_Billing_Details', 'Meter_Reading', 'Payment', 'Service_Provider', 'Tariff_Details', 'Zipcode_Details', 'Views', 'Stored Procedures', 'Functions', 'INCREMENNT2', 'innodb', and 'LGWT'. The 'Views' section is expanded, showing 'customer_count_under_service_providers'.

b.

View Name: customer_count_under_service_providers

Description: Admin View where the admin can view the number of customer under each service provider.

create view customer_count_under_service_providers as
select c.service_provider_name as 'Provider Name',count(customer_id) 'Total Number of Customers' from Customer c
inner join Service_Provider sp on
sp.provider_name=c.service_provider_name
group by sp.provider_name;



MySQL Workbench

UNCRCDSConnectionMahadik

Electricity Billing System

Customer View Admin View Procedure Calls

Admin View

1. View All Customers
2. Register Customer
3. Change Tariff Rate
4. Delete a Customer
5. Login

Enter the Query

Select * from customer_electricity_bills;

Execute

electricity_bill...	customer_id	bill_generated...	bill_due_date	billing_cycle_s...	number_of_bill...	payment_amo...	amount_due	meter_reading...	payment_id
10000	4	2018-08-23	2018-09-17	2018-07-22	30	254.0	254.0	6	105
10002	2	2018-09-25	2018-10-10	2018-08-20	30	240.88	240.88	4	103
10003	3	2018-09-26	2018-10-15	2018-08-25	25	255.84	255.84	5	104
10004	4	2018-09-23	2018-10-17	2018-08-22	30	240.0	240.0	7	106
10005	5	2018-09-20	2018-10-15	2018-08-18	31	189.88	189.88	9	107
10018	4	2018-10-23	2018-11-17	2018-09-22	30	152.0	152.0	8	108
100002	1	2018-07-18	2018-08-15	2018-06-15	36	250.8	250.8	1	100
100010	1	2018-09-18	2018-10-10	2018-08-16	35	209.0	209.0	2	101
100036	1	2018-10-18	2018-11-15	2018-09-15	36	209.54	209.54	3	102
100037	5	2018-10-22	2018-11-12	2018-09-21	31	133.4	133.4	10	108
100038	5	2018-11-21	2018-12-11	2018-10-23	28	211.52	211.52	11	108

Object Info Session

564 21:52:25 /150003 SET @saved_cs_client = @@character_set_client /

0 row(s) affected 0.031 sec

Query Completed

Search the web and Windows

10:34 PM 12/2/2018

c.

View Name: payment_defaulters

Description: To find all customers who are defaulters i.e they did not paid the bill and due date is also crossed.

create view payment_defaulters as
select c.first_name,c.last_name, e.amount_due,e.bill_due_date
from Customer c
inner join Meter_Reading mr on
mr.customer_id=c.customer_id
inner join Electricity_Billing_Details e
on e.meter_reading_id=mr.meter_reading_id
where e.bill_due_date<curdate() and e.payment_id is null;

MySQL Workbench

UNCORRDSConnectionMahadik x

Electricity Billing System

Customer View Admin View Procedure Calls

Admin View

1. View All Customers
2. Register Customer
3. Change Tariff Rate
4. Delete a Customer
5. Login

Enter the Query

Select * from payment_defaulters;

Execute

first_name	last_name	amount_due	bill_due_date
Madonna	Reynolds	240.0	2018-10-17

Object Info Session

564 21:52:25 /150003 SET @saved_cs_client = @@character_set_client */

0 rows affected 0.031 sec

Query Completed

Search the web and Windows

10:35 PM 12/2/2018