

Visvesvaraya Technological University
Belagavi-590 018,
Karnataka



A Mini Project Report on

“Cable Management System”

Mini Project Report submitted in partial fulfilment of the requirement for the
DBMS Laboratory with Mini Project [18CSL58]

Bachelor of Engineering
in
Computer Science and Engineering

Submitted by
Harsha K S [1JT18CS020]
Gowhar Ahmad Reshi [1JT18CS018]



Department of Computer Science and Engineering
Jyothy Institute of Technology
Tataguni, Bengaluru-560082



Jyothy Institute of Technology
Tataguni, Bengaluru-560082
Department of Computer Science and Engineering



CERTIFICATE

Certified that the mini project work entitled **“Cable Management System”** carried out by **Harsha K S[1JT18CS020]** and **Gowhar Ahmad Reshi [1JT18CS018]** bonafide students of Jyothy Institute of Technology, in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** department of the **Visvesvaraya Technological University, Belagavi** during the year **2019-2020**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

Mrs. Nikitha S

Guide, Asst. Professor

Dept. of CSE

Dept. of CSE

Dr. Prabhanjan S

Professor & HOD

External Viva Examiner

Signature with Date :

- 1.
- 2.

ACKNOWLEDGEMENT

Firstly, we are very grateful to this esteemed institution “**Jyothy Institute of Technology**” for providing us an opportunity to complete our project.

We express our sincere thanks to our **Principal Dr. Gopalakrishna K** for providing us with adequate facilities to undertake this project.

We would like to thank **Dr. Prabhanjan S, Professor and Head of Computer Science** and Engineering Department for providing for his valuable support.

We would like to thank our guides **Mrs.Nikitha S, Assistant Professor** for their keen interest and guidance in preparing this work.

Finally, we would thank all our friends who have helped us directly or indirectly in this project.

Harsha K S [1JT18CS020]

Gowhar Ahmad Reshi [1JT18CS018]

ABSTRACT

There are staffs who are working in the dth service providing companies. Where staffs belongs to different departments. As they belong to different department each of them will be provided with their unique id in those companies. There are operation staffs who keeps track of all the activities that are going on in their respective all departments. And the department is also the important one that categorizes the staffs based on the fields they work. So each of the department will be provided with the unique number and their respective name. The operators are the key persons in the **dth** service providing companies. Where each company will be having many of the operators involved. There are some details such as name, phone number and address that are associated with each of the operators including their id. The amount collector is a person whose main job is to collect the amount regularly after every month. He is also identified with his uniquely specified id. He also takes part in managing the offline customers, and does the registration part for them in the application and also entering the status of payment of each customer. Based on the status of payment the operator will decide to continue the subscription or to end it for a customer. Then comes the service staff, he is also given with an id. His main job is to solve the technical issues that the customer come across and he also is involved to make a note in the application that which customer had the issue and is that been solved. Then comes the customer, where the customer will be having their choice to do their registration and subscription to the channels or to the package which ever they are interested in through the application in online. But also provided with staff to do that as mentioned above. The details that the customer enters i.e. his name, phone number, address and others while they register will be added automatically to the database.

TABLE OF CONTENT

SL No	Description	Page No.
1	INTRODUCTION	6-8
2	DESIGN	9-12
3	IMPLEMENTATION	13-19
4	RESULTS AND SNAPSHOTS	20-33
5	CONCLUSION	34

CHAPTER 1

INTRODUCTION

1 INTRODUCTION

1.1 INTRODUCTION TO DBMS:

A database management system refers to technology for creating and managing databases. DBMS is a software tool to organize (create, retrieve, update and manage) data in a database.

The main aim of DBMS is to supply a way to store up and retrieve database information that is both convenient and efficient.

Advantages of databases:

- To develop software applications in less time.
- Data independence and efficient use of data.
- For uniform data administration.
- For data integrity and security.
- To use user-friendly declarative query language.

1.2 INTRODUCTION TO SQL:

SQL is an abbreviation of structured query language, is a language to request data from a database, to add, update, remove data within a database, or to manipulate the metadata of the database.

SQL is a declarative language in which the expected result or operation is given without the specific details about how to accomplish the task. The steps required to execute SQL statement are handled transparently by the SQL database. Sometimes SQL is characterised as non-procedural because procedural language generally require the details of the operations to be specified, such as opening and closing tables, loading and searching indexes, or flushing buffers and writing data to file system. Therefore, SQL is considered to be designed at a higher conceptual level of operation than procedural languages because the lower level logical and physical operation aren't specified and are determined by the SQL engine or server process that executes it.

1.3 INTRODUCTION TO CABLE MANAGEMENT SYSTEM:

The main objective of cable management system project is to create an application for the cable operator to reduce their paperwork. And to have the information to be stored digitally. So that the cable operator can see the details of his customer and also his staff. The workers under the operator are going to get the details of the customer and the details can also be given directly by the customer through the application via online. And this will be updated continuously. And retrieving the data of individual customer will become very easier for both the operator and the other staffs. This data for them is very important. when the customer needs any service from the operators then customer gets the respective staff details through the application, so that they will contact the staff directly, And also provide the customers to choose their packages by themselves in the application, Or even that will be done by the staff.

1.4 SCOPE AND IMPORTANCE OF WORK:

The scope of the project is clear to give a simple and attractive application to simplify the work as well as to reduce the efforts while doing it offline or we can say by doing with old methods.

In this project we are able to store the information of the customer, the subscription he has purchased and the payment details. This will mainly help the staffs to easily view the details of the customer. The main role is played by the operator who will go through the information entered by the other staffs and then decide whether to continue the subscription chosen by the customer or to end it. This application is surely be very useful and comfortable for even cable operation staffs and also to the customers.

CHAPTER 2

DESIGN

2 THEORY OF ER DIAGRAM

An entity relationship diagram shows the relationships of entity sets stored in a database. It mainly describes the structure of a database with the help of a diagram, which is so called the entity relationship diagram. An ER model is the design or blueprint of the database that can later be implemented as a database. The main components of ER model are as said above entity set and relationship set.

2.1 ENTITIES:

An entity is an object that exists. It doesn't have to do anything, it just has to exist. In database administration, an entity can be a single thing, person, place, or an object. Data can be stored about such entities. A design tool that allows database administration to view the relationships between several entities is basically called as an ER diagram.

2.2 RELATIONSHIPS:

A relational database collects different types of data sets that use tables, records, and columns. It is used to create a well defined relationship between database tables so that relational database can be easily stored. For example say we need to have a connection between the two entities such as staff and customer we can connect them using the relationship say staff serves customer where serves is the relation that exists between them.

2.3 ATTRIBUTES:

In general, an attribute is a characteristic. In a database management system, an

attribute refers to a database component, such as a table.

It also may refer to a database field. Attributes describe the instances in the column of a database.

2.4 ER DIAGRAM

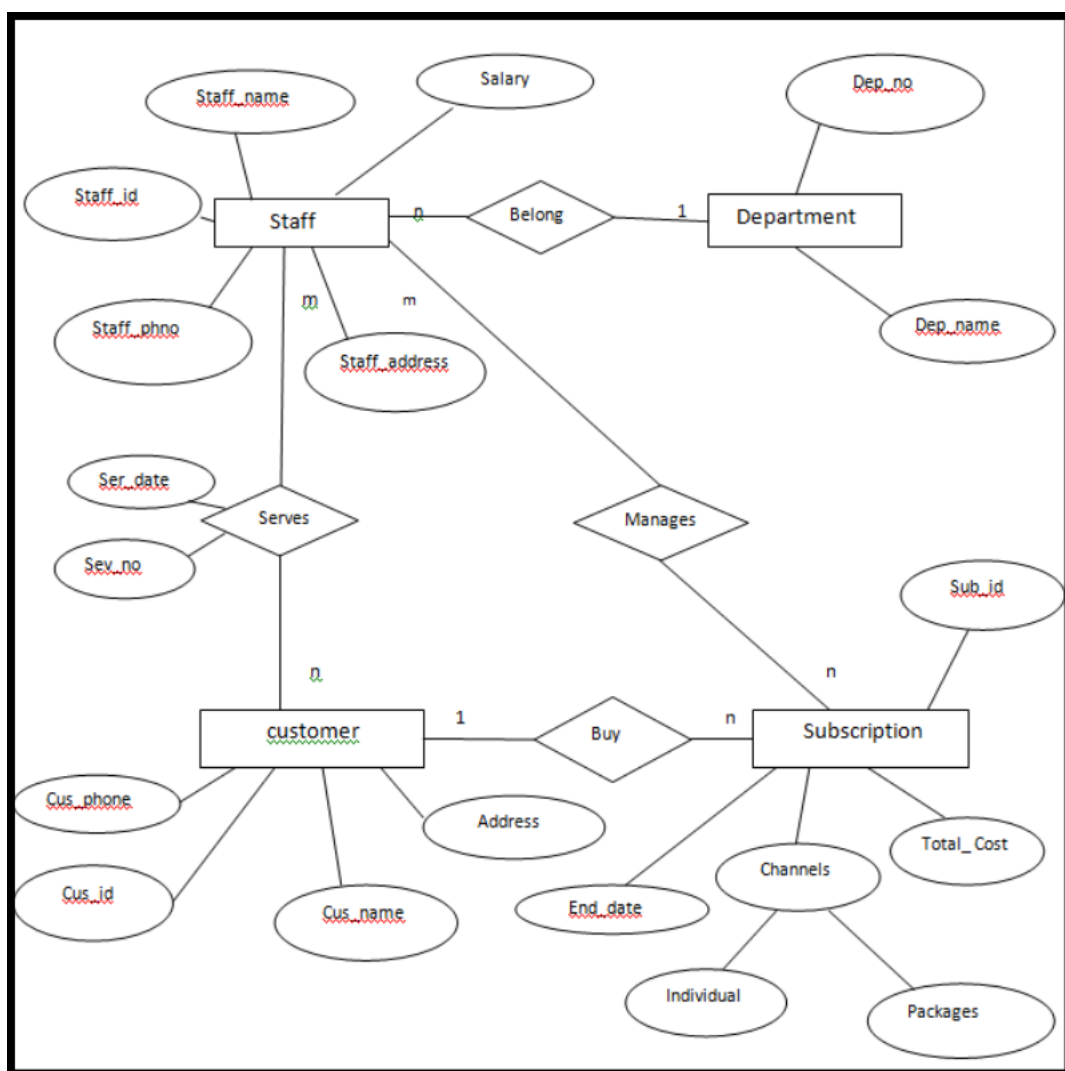


Figure 2.4 ER-diagram for cable management system

2.5 SCHEMA DIAGRAM

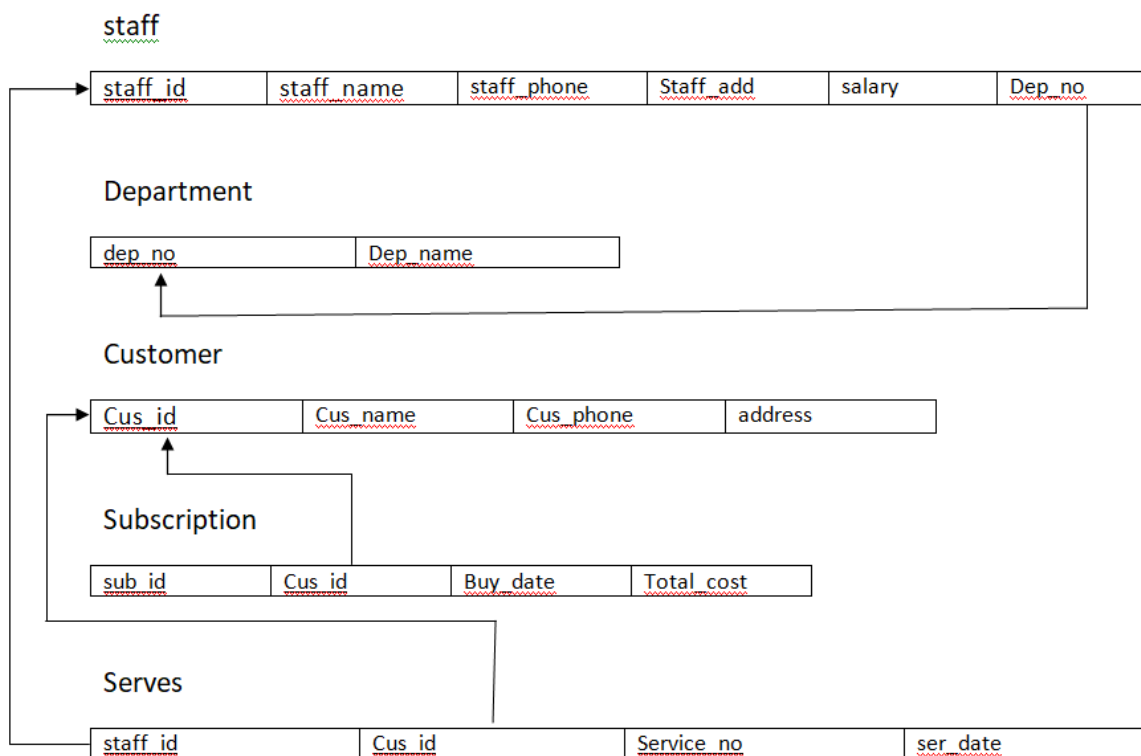


Figure 2.5 schema diagram for cable management system

2.6 LIST OF TABLES:

1. STAFF
2. DEPARTMENT
3. CUSTOMER
4. SUBSCRIPTION

5. SERVICE

CHAPTER 3 IMPLEMENTATION

3.1 CREATE TABLE STATEMENTS

```
create table department (dep_no varchar(20),dep_name varchar(20),primary
key(dep_no));
```

```
create table staff( staff_id varchar(20), staff_name varchar(20),address varchar(20),
ph_no varchar(20),gender char(10),dep_no varchar(20),salary varchar(20),
primary key(staff_id),
foreign key(dep_no) references department (dep_no) on update cascade on delete
cascade);
```

```
create table customer(customer_ID varchar(20),customer_name varchar(20),
phone_no varchar(20),address varchar(20),
primary key(customer_id));
```

```
create table subscription(sub_id varchar(20),cus_id varchar(20),buy_date
date,total_cost int,primary key(sub_id,cus_id),
foreign key(cus_id) references customer(customer_ID) on delete cascade on update
cascade);
```

```
create table service(service_no varchar(20),cus_id varchar(20),staff_id
varchar(20),service_date date,
primary key(service_no,cus_id,staff_id),
foreign key(cus_id) references customer(customer_ID) on delete cascade on
update cascade,
foreign key(staff_id) references staff(staff_id) on delete cascade on update
```

cascade);

3.2 INSERTING VALUES TO TABLES

INSERTION INTO DEPARTMENT TABLE:

insert into department values("1","operation");

insert into department values("2","finance");

insert into department values("3","service");

INSERTION INTO STAFF TABLE:

insert into staff values("op1","ganesh","srinagar","9556734322","male","1","25000");

insert into staff values("op2","vignesh","girinagar","9556734456","male","1","20000");

insert into staff
values("op3","gowri","vijayanagar","9336734456","female","1","23000");

insert into staff
values("op4","parvathi","jayanagar","9236734456","female","1","25000");

insert into staff values("op5","shiva","kilas nagar","8806734456","male","1","25000");

insert into staff
values("fin2","mahesh","balaginagar","9906734456","male","2","16000");

insert into staff
values("fin4","prajwal","basavangudi","9903330056","male","2","18000");


```
insert into staff values("fin5","bhavana","chikpet","9943330056","female","2","16000");
```

```
insert into staff values("ser1","gagan","vilsan  
garden","9945530056","male","3","20000");
```

```
insert into staff values("ser2","deepu","srinivas  
nagar","8845530056","male","3","18000");
```

```
insert into staff values("ser3","deleep","srinagar","8845530098","male","3","20000");
```

```
insert into staff values("ser4","sunil","basavangudi","8345530098","male","3","18000");
```

```
insert into staff values("ser5","rudresh","girinagar","8345530001","male","3","20000");
```

INSERTION INTO CUSTOMER TABLE:

```
insert into customer values("1","srinivas","9740245873","girinagar");
```

```
insert into customer values("2","vijayalaksmi","9449182246","srinagar");
```

```
insert into customer values("3","shivam","8050018571","vijayanagar");
```

```
insert into customer values("4","bargav","7975520193","srinivas nagar");
```

```
insert into customer values("5","omakar","7975520193","basavanagudi");
```

```
insert into customer values("6","akash","7349788637","jayaprakash nagar");
```

```
insert into customer values("7","akshay","9731460483","nagarabhavi");
```

```
insert into customer values("8","bhavani","9449182246","chikpet");
```

```
insert into customer values("9","bhagvan","8059918571","vilsan garden");
```

```
insert into customer values("10","bhvana","8059918534","srinagar");
```

INSERTION INTO SUBSCRIPTION TABLE:

```
insert into subscription values("incs1","1","2017-01-01 ",39);
```

```
insert into subscription values("incs2","2","2017-06-04 ",42);
```

```
insert into subscription values("incs3","4","2017-02-18 ",56);
```

```
insert into subscription values("incs4","3","2017-02-08",42);
```

```
insert into subscription values("pac1","1","2017-01-01 ",290);
```

```
insert into subscription values("pac2","3","2017-01-07 ",300);
```

```
insert into subscription values("pac3","4","2017-02-18 ",250);
```

INSERTION INTO SERVICE TABLE:

```
insert into service values("1","1","ser5","2017-02-03");
```

```
insert into service values("2","5","ser4","2017-05-19");
```

```
insert into service values("3","4","ser2","2017-03-06");
```

```
insert into service values("4","3","ser5","2017-02-23");
```

```
insert into service values("5","2","ser3","2017-07-09");
```

3.3 GUI IMPLIMENTATION

FOR INSERTION (CODE SNIPPET):

```
String sql="insert into customer values(?,?,?,?)";
    try{
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/database_project",
"root","root123*");
        PreparedStatement pre=con.prepareStatement(sql);
        pre.setString(1,jTextField2.getText());
        pre.setString(2,jTextField1.getText());
        pre.setString(3,jTextField5.getText());
        pre.setString(4,jTextField4.getText());

        int x=pre.executeUpdate();
        if(x==1){
            JOptionPane.showMessageDialog(null,"added
```

```
successful","insertion",JOptionPane.PLAIN_MESSAGE);
```

```
    }  
}  
    catch(Exception e){
```

```
JOptionPane.showMessageDialog(null,e,"error",JOptionPane.PLAIN_MESSAGE);  
}
```

This part of code basically takes the input from the user through the text fields that are provided, and then will make that data entered by the user to be stored in the respective tables in the back end.

FOR UPDATION (CODE SNIPPET):

```
String sql="update customer set customer_name=?,phone_no=?,address=? where  
customer_ID=?";
```

```
    try{
```

```
        Connection
```

```
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/database_project",  
"root","root123*");
```

```
        PreparedStatement pre=con.prepareStatement(sql);
```

```
        pre.setString(1,jTextField1.getText());
```

```
        pre.setString(4,jTextField2.getText());
```

```
        pre.setString(2,jTextField5.getText());
```

```
        pre.setString(3,jTextField4.getText());
```

```
        int x= pre.executeUpdate();
```

```
        if(x==1){
```

```
            JOptionPane.showMessageDialog(null,"update  
done","updation",JOptionPane.PLAIN_MESSAGE);
```

```

        }

    }catch(Exception e){

JOptionPane.showMessageDialog(null,e,"error",JOptionPane.PLAIN_MESSAGE);

    }

```

This part of code will basically help in updating the values/ or the data that are to be modified. In this project updating plays a very important role, for example if the customer hasn't paid the amount for the subscription that should be updated immediately, so the operator will deny the service for that user.

FOR CLEARING CONTENT (CODE SNIPPET):

```

jTextField1.setText("");
jTextField2.setText("");
jTextField3.setText("");

```

This part of code will mainly clear the content in the text fields.

VIEWING DETAILS IN TABLES (CODE SNIPPET):

```

DefaultTableModel tb1=(DefaultTableModel)jTable1.getModel();
    tb1.setRowCount(0);
    try{
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/database_project",
"root","root123*");
        Statement sta=con.createStatement();
        ResultSet res=sta.executeQuery("select * from customer");
        while(res.next()){

```

```

        String customer_ID=res.getString(1);
        String customer_name=res.getString(2);
        String phone_no=res.getString(3);
        String address=res.getString(4);

        String record[]={customer_ID,customer_name,phone_no,address};
        tb1.addRow(record);
    }
}
catch(Exception e){

JOptionPane.showMessageDialog(null,e,"error",JOptionPane.PLAIN_MESSAGE);
}

```

The above part of code will mainly make the data stored in the tables in back end to be displayed in the front end. That too inside the tables, so that the data will be organized.

CHAPTER

4

RESULTS AND SNAPSHOTS

SELECT * FROM TABLES

SELECT * FROM STAFF;

```
mysql> select * from staff;
```

staff_id	staff_name	address	ph_no	gender	dep_no	salary
fin1	bramha	jayaprakashnagar	8906734456	male	2	20500
fin2	mahesh	balaginagar	9906734456	male	2	16000
fin3	pavithra	kormangala	9906730056	female	2	15000
fin4	prajwal	basavangudi	9903330056	male	2	18000
fin5	bhavana	chikpet	9943330056	female	2	16000
op1	ganesh	srinagar	9556734322	male	1	25850
op2	vignesh	girinagar	9556734456	male	1	18000
op3	gowri	vijayanagar	9336734456	female	1	23000
op4	parvathi	jayanagar	9236734456	female	1	25000
op5	shiva	kilas nagar	8806734456	male	1	25000
ser1	gagan	vilsan garden	9945530056	male	3	19850
ser2	deepu	srinivas nagar	8845530056	male	3	18000
ser3	deleep	srinagar	8845530098	male	3	20000
ser4	sunil	basavangudi	8345530098	male	3	18000
ser5	rudresh	girinagar	8345530001	male	3	20000

```
15 rows in set (0.23 sec)
```

Figure 4.1 staff table

SELECT * FROM CUSTOMER;

```
mysql> select * from customer;
```

customer_ID	customer_name	phone_no	address
1	srinivas	9740245873	girinagar
10	bhvana	8059918534	srinagar
2	vijayalaksmi	9449182246	srinagar
3	shivam	8050018571	vijayanagar
4	bargav	7975520193	srinivas nagar
5	omakar	7975520193	basavanagudi
6	akash	7349788637	jayaprakash nagar
7	akshay	9731460483	nagarabhavi
8	bhavani	9449182246	chikpet
9	bhagvan	8059918571	vilsan garden

```
10 rows in set (0.07 sec)
```

Figure 4.2 customer table

SELECT * FROM DEPARTMENT;

```
mysql> select * from department;
+-----+-----+
| dep_no | dep_name |
+-----+-----+
| 1      | operation |
| 2      | finance  |
| 3      | service  |
+-----+-----+
3 rows in set (0.07 sec)
```

Figure 4.3 department table

SELECT * FROM SUBSCRIPTION;

```
mysql> select * from subscription;
+-----+-----+-----+-----+
| sub_id | cus_id | buy_date | total_cost |
+-----+-----+-----+-----+
| incs1  | 1      | 2017-01-01 | 39 |
| incs2  | 2      | 2017-06-04 | 42 |
| incs3  | 4      | 2017-02-18 | 56 |
| incs4  | 3      | 2017-02-08 | 42 |
| pac1   | 1      | 2017-01-01 | 290 |
| pac2   | 3      | 2017-01-07 | 300 |
| pac3   | 4      | 2017-02-18 | 250 |
+-----+-----+-----+-----+
7 rows in set (0.05 sec)
```

Figure 4.4 subscription table

SELECT * FROM SERVICE;

```
mysql> select * from service;
+-----+-----+-----+-----+
| service_no | cus_id | staff_id | service_date |
+-----+-----+-----+-----+
| 1          | 1      | ser5     | 2017-02-03   |
| 2          | 5      | ser4     | 2017-05-19   |
| 3          | 4      | ser2     | 2017-03-06   |
| 4          | 3      | ser5     | 2017-02-23   |
| 5          | 2      | ser3     | 2017-07-09   |
+-----+-----+-----+-----+
5 rows in set (0.03 sec)
```


Figure 4.5 service table

QUERIES SNAPSHOT

QUERY -1:

Delimiter \$\$

Create procedure query1(in sub_id, in cus_id);

Begin

Select c.customer_name,s.buy_date,s.total_cost from customer c,subscription s
where c.customer_ID=s.cus_id and sub_id =sub_id and cus_id =cus_id;

End \$\$

“ Call query1(?,?)”

Output:

The screenshot shows a web application interface with a dark blue background. At the top, there is a header with the word "Subscription" in white. Below the header, there are two buttons: "Individual selection" and "package selection". Under "Individual selection", there are two input fields: "subscription ID" with the value "incs1" and "customer ID" with the value "1". Below these fields are three buttons: "Search", "view complete", and "back". To the right of the input fields, there is a table with the following data:

Customer name	Buy date	total cost
srinivas	2017-01-01	39

Figure 4.6 query-1 output

QUERY - 2:

Select cus_id from subscription where buy_date between ? and ? group by cus_id;

Output:

query 1

Retrive the customer ID who has purchased the subscription between specified dates

start date

2017-01-01

end date

2017-03-01

execute

back

customer_id

1

3

4

Figure 4.7 query-2 output

QUERY - 3:

Select max(s.salary),min(s.salary), sum(s.salary),d.dep_no from staff f,department d where s.dep_no=d.dep_no and d.dep_no=?;

Output:

query 2

Retrive the maximum, minimum,sum of salaries
for the specified department

department number

max_salary	min_salary	sum_salary	dep_name
20500	15000	85500	finance

Figure 4.8 query-3 output

QUERY – 4:

Select s.staff_id, ser.service_no, c.customer_name from staff s, service ser,
customer c where s.staff_id=ser.staff_id and c.customer_ID=ser.cus_id and
s.address <> c.address;

Output:

query3

Retrive the Id of staff and the name of customer where the staff had serviced the customer who is not in the area of service.

staff_id	service_no	customer_name
ser5	4	shivam
ser4	2	omakar

view result

back

Figure 4.9 query-4 output

QUERY – 5:

Select s.staff_id, s.staff_name,s.salary, d.dep_name from staff s, department d where s.dep_no=d.dep_no and salary in (select max(salary) from staff group by dep_no);

Output:

query 4

Retrive the staff details who has highest salary in their respective department.

sattf_id	staff_name	salary	dep_name
op1	ganesh	25850	operation
fin1	bramha	20500	finance
ser3	deleep	20000	service
ser5	rudresh	20000	service

view result

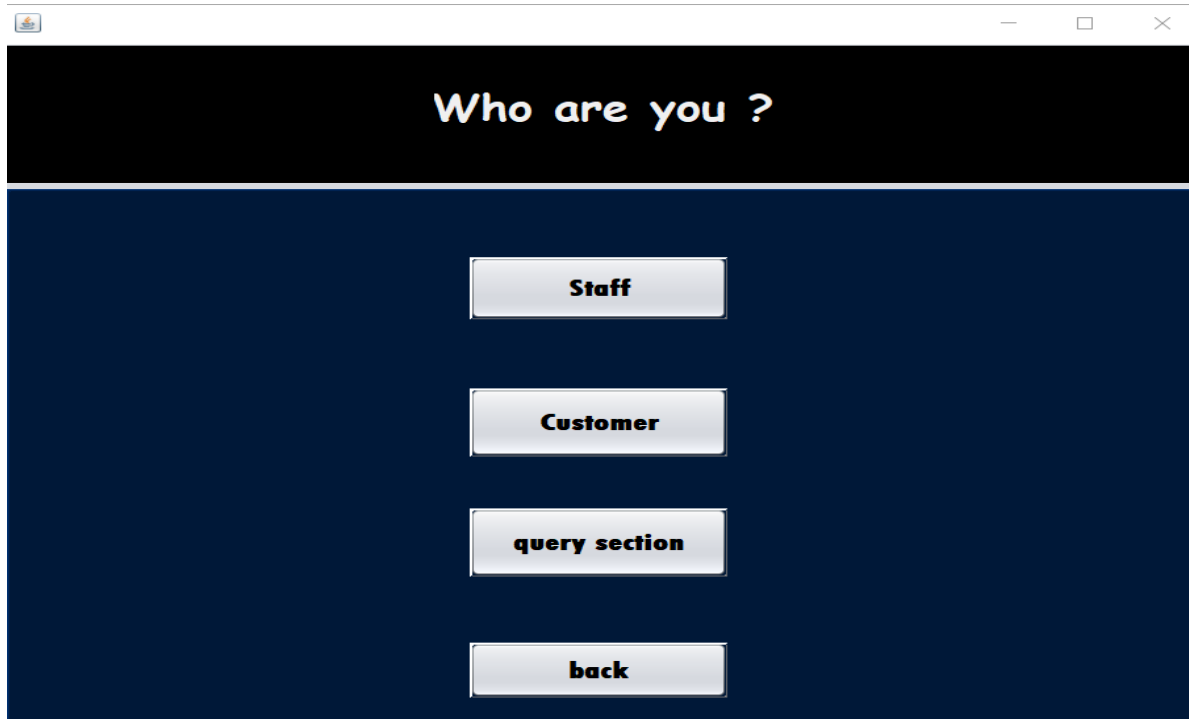
back

Figure 4.10 query-5 output

FRONT END SNAPSHOT

LOGIN FORMS:

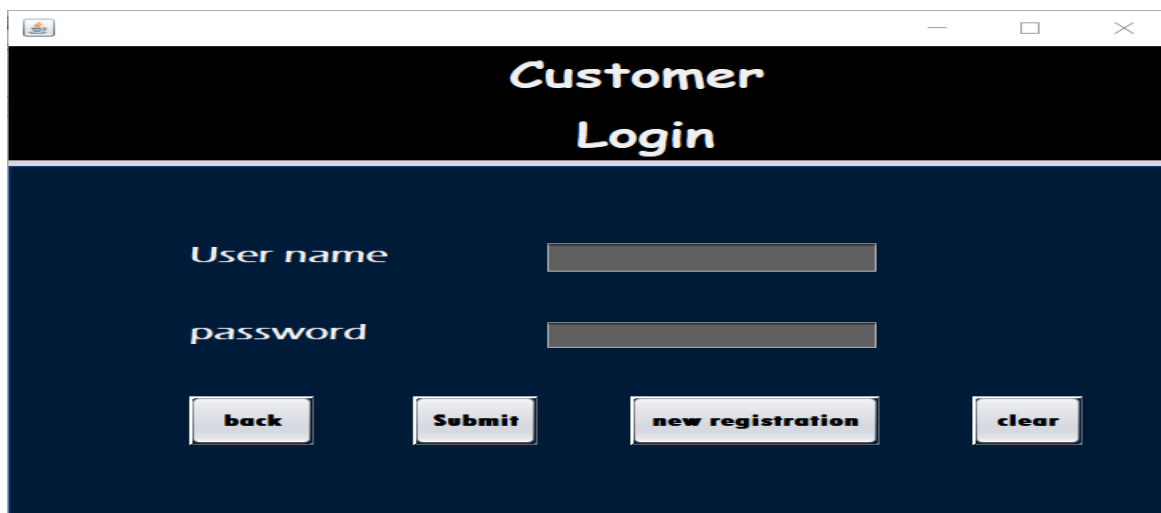
HOME PAGE:



The screenshot shows a web browser window with a dark blue background. At the top, a black header bar contains the text "Who are you ?" in white. Below the header, four light gray buttons with black text are arranged vertically in the center: "Staff", "Customer", "query section", and "back".

Figure 4.11 home page

CUSTOMER LOGIN PAGE:



The screenshot shows a web browser window with a dark blue background. At the top, a black header bar contains the text "Customer Login" in white. Below the header, the form includes two labels, "User name" and "password", each followed by a light gray input field. At the bottom, four light gray buttons with black text are arranged horizontally: "back", "Submit", "new registration", and "clear".

Figure 4.12

customer login

[note: similarly other three forms for staff login is created]

STAFF PAGE:

OPERATION STAFF PAGE:

Details Entry By Operator

customer ID

service status

staff ID

start date

end date

customer ID	staff ID	service status	start date	end date
-------------	----------	----------------	------------	----------

Add **view payment details** **view service details** **update** **select** **back**

Figure 4.13 customer details by operator staff

FINANCIAL STAFF PAGE:

Details Entry By Financial Staff

customer ID

customer name

payment status

subscription ID

customer ID	customer name	pay_status	subscription ID
-------------	---------------	------------	-----------------

Add

update

Add customer

view

clear

back

Figure 4.14 customer details by financial staff

SERVICE STAFF PAGE:

— □ ×

Details Entry By Service Staff

customer ID

staff ID

service number

service date

customer ID	staff ID	service number	service date
-------------	----------	----------------	--------------

submit

update

view

clear

back

STAFF REGISTRATION:

Figure 4.16 staff registration form

CUSTOMERS PAGE:

33

Figure 4.17 customer page/ subscription page
CUSTOMERS REGISTRATION PAGE:

Customer Registration

customer name

customer ID

phone number

address

customer ID	customer name	phone number	address
-------------	---------------	--------------	---------

Figure 4.18 customer registration form

INDIVIDUAL CHANNEL SUBSCRIPTION PAGE:

Individual channel selection

customer ID

subscription ID

buy date

total cost

channel ID	channel name	cost	select
------------	--------------	------	--------

Figure 4.19 individual channel selection

PACKAGE SUBSCRIPTION PAGE:

The screenshot shows a web application window titled "Package selection". The interface has a dark blue background. On the left, there are four input fields labeled "customer ID", "subscription ID", "buy date", and "total cost". To the right of these fields is a table with four columns: "package ID", "package name", "cost", and "select". The table is currently empty. At the bottom of the page, there are four buttons: "back", "buy", "update", and "show packages".

customer ID	subscription ID	buy date	total cost	package ID	package name	cost	select
-------------	-----------------	----------	------------	------------	--------------	------	--------

Figure 4.20 package selection

5 CONCLUSION

We have successfully completed the implementation of cable management system, where the staffs in the DTH company who were struggling for maintaining the records in a book which was not so better for storing the information, and not only about storing it also matters about the retrieving of the specified information was also very difficult. Now by using this application one can easily enter the information of the customer, view the details and also update the details if required. And this application not only has the staff interaction it is also provided for customers too..

Where the customers can login through their id specified and can themselves subscribe the channels or the packages they want and if the customer doesn't want to do so then this part will be handled by the staff, because this data is very important for them.

So finally, we are succeeded in building an application by using the concept of java and sql, and we mainly used NetBeans for our front end development and mysql for the backend implementation as the platform.

6 REFERENCES

6.1 JAVA DATABASE CONNECTIVITY:

["https://www.javapoint.com/java-jdbc"](https://www.javapoint.com/java-jdbc)

["https://www.udemy.com/course/how-to-connect-java-jdbc-to-mysql/"](https://www.udemy.com/course/how-to-connect-java-jdbc-to-mysql/)

6.2 TEXTBOOK:

Database management system by ramakrishnan.

Fundamentals of database system by Ramez Elmarsi, Navathe.

6.3 NETBEANS:

["https://netbeans.org/kb/docs/java/quickstart.html"](https://netbeans.org/kb/docs/java/quickstart.html)

["https://netbeans.org/kb/docs/java/quickstart-gui.html"](https://netbeans.org/kb/docs/java/quickstart-gui.html)

6.4 MYSQL:

["https://www.w3schools.com/sql/sql_syntax.asp"](https://www.w3schools.com/sql/sql_syntax.asp)