AIM:

To create an online package delivery system to make it easy for the customer to place and track orders and for the administrator to schedule deliveries and foresee them using java and operating system related concepts like scheduling,semaphores,bounded buffer,etc.

OBJECTIVE:

- To schedule all the packages in the order of priority which is in this case speed delivery (high priority)

  and normal delivery (low priority). We do this using priority scheduling algorithm.

- We use bounded buffer concept to find number of packages that can be filled in a truck as its capacity is fixed.

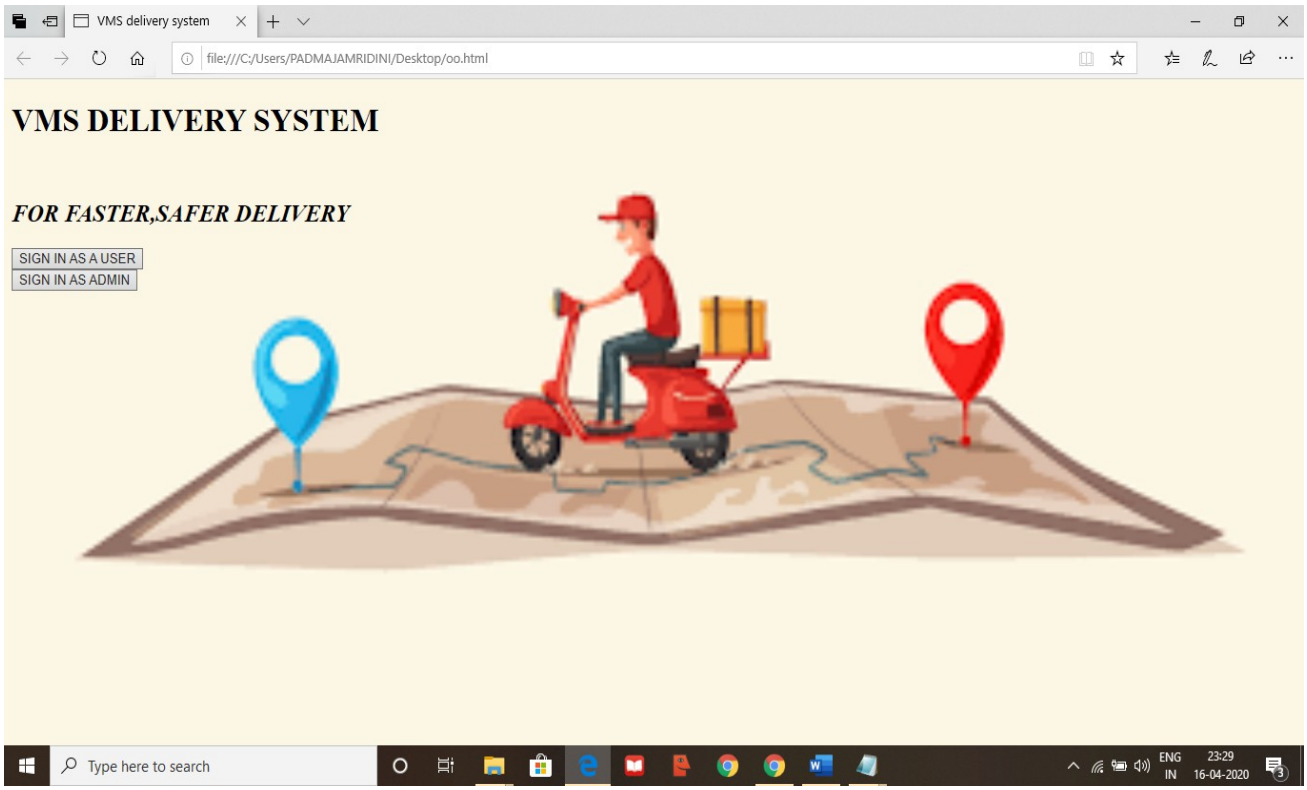
- We also use semaphore concept to find the number of available trucks so that its easy to keep track of the delivery status.

Here we have used java, html and MySQL for our project. We accomplished this by using NetBeans platform where we created html forms in jsp pages and downloaded MySQL JDBC connector for establishing the connection between MySQL and java.

In our project we take the information from the user store it in MySQL and prioritize it and then display the final output consisting of amount to be paid,to the users for their respective orders. Whereas the administrator can see the delivery dates of all the packages and also the information related to the orders loaded into trucks and the availability of trucks.

Home page:

This is the page where the person gets access as a user or admin according to their need.



Source code:

```
<!DOCTYPE html>
<html>
  <head>
 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
 <title>VMS delivery system</title>
 </head>
 <body>
 <h1><b>VMS DELIVERY SYSTEM</b></h1>
 <br> <h2><b><i>FOR FASTER,SAFER DELIVERY</i></b></h2>
 <form  name="page1" action="" method="post">
 <div class="form-group">
 <button type="submit" name="action" onClick="user1.jsp" value="signuser" class="btn">SIGN IN AS A USER</button>
 <a href="user1.jsp">sign in a user</a>
  </div>
```

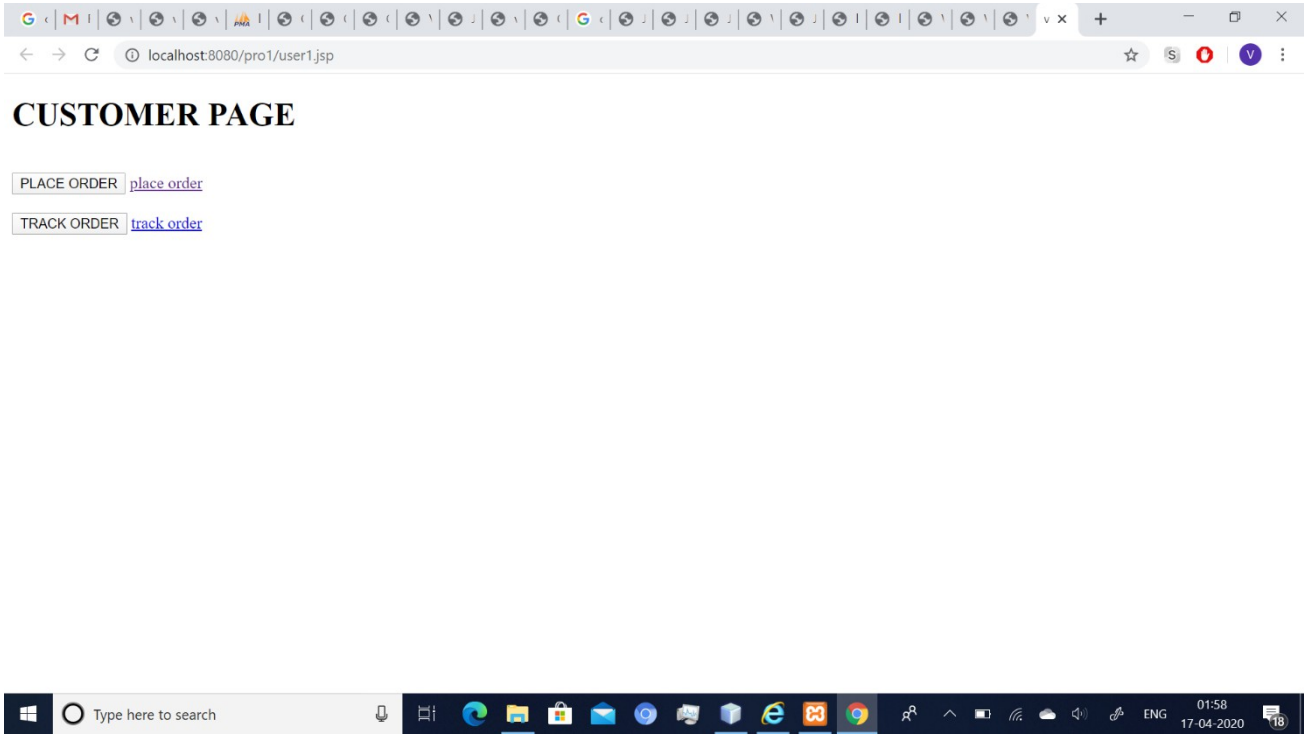
```
 <div>
  <button type="submit" name="action" onClick="admin.jsp"
value="signadmin" class="btn">SIGN IN AS ADMIN</button>
<a href="admin.jsp">admin</a>
 </div>
 </form>
 </body>
</html>
```

USER:

This page appears if the person chooses the user option which means he customer.

Here customer gets two options:

1) Place orders (places new orders)
2) Track orders (gets the info about already placed orders)

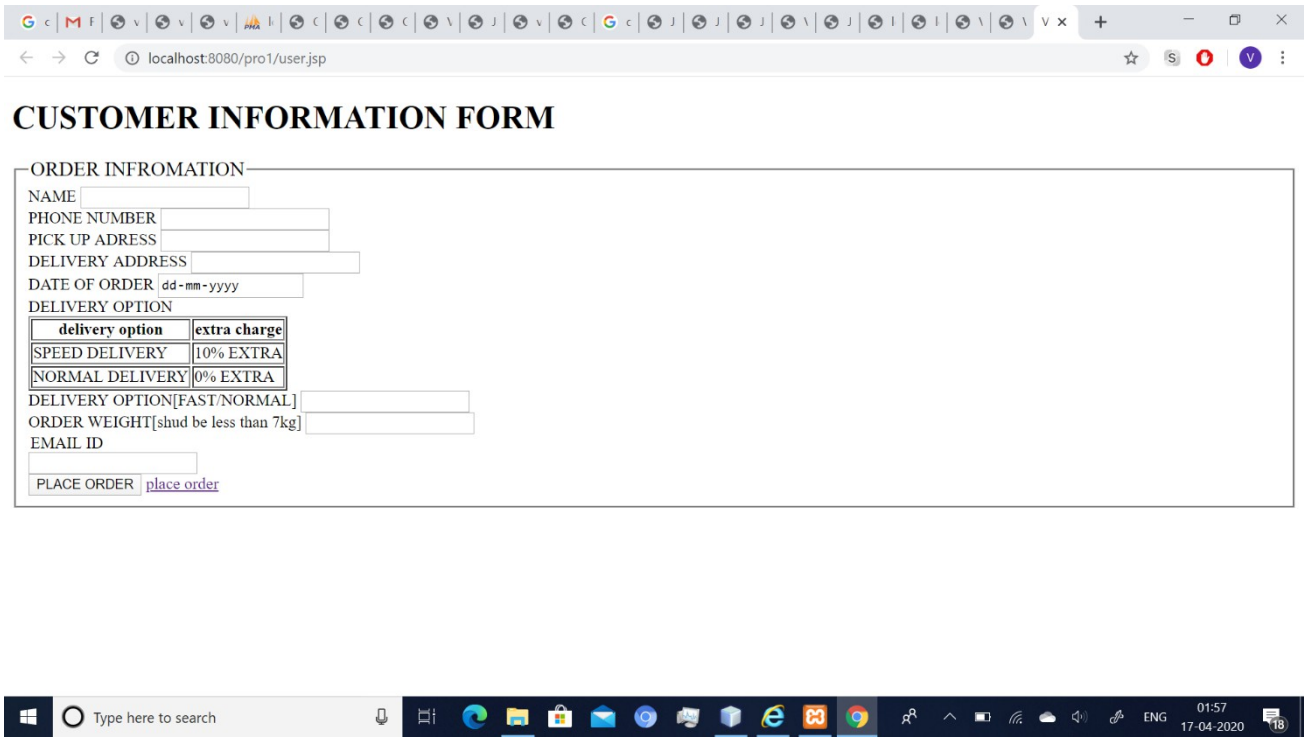


Source code:

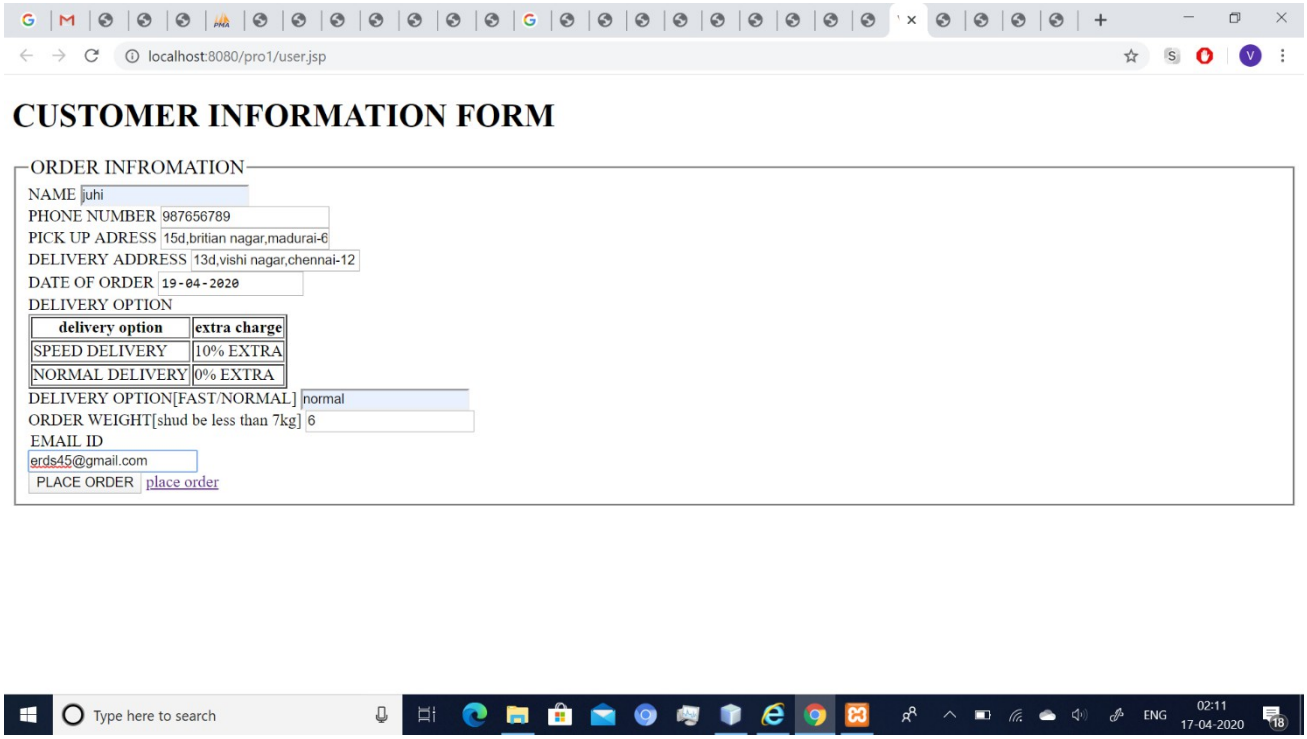
```
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>vms</title>
   </head>
   <body>
      <form name="form3" method="post">
         <h1><b> CUSTOMER PAGE</b></h1>
         <div>
            <br>
            <button type="submit" name="user" onclick="user.jsp">PLACE ORDER</button>
            <a href="user.jsp">place order</a>
         </div>
         <div>
            <br>
            <button type="submit" name="track" onclick="track.jsp">TRACK
ORDER</button>
            <a href="track.jsp">track order</a>
         </div>

      </form>
   </body>
</html>
```

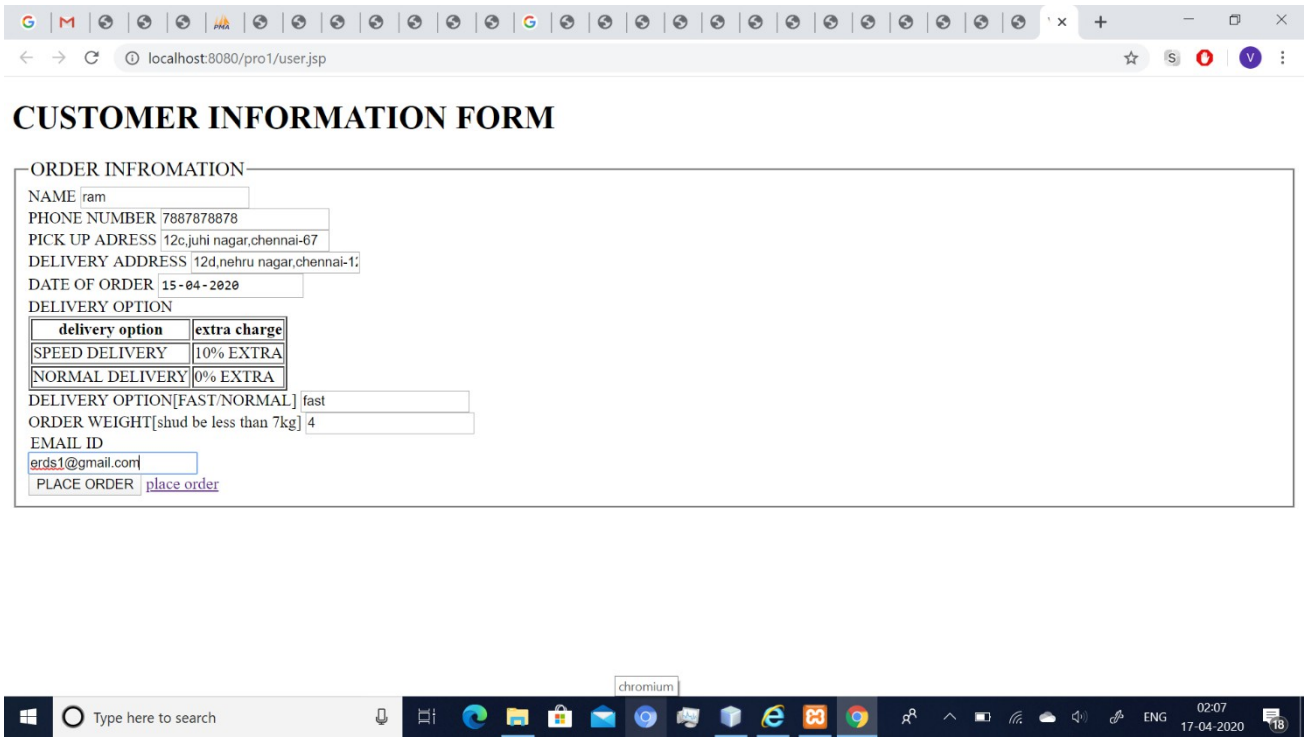
PLACE ORDER:

- This page appears if the customer clicks place order option. This means he wants to place new orders.
- In this page customer must fill in all the details and mainly choose delivery option whether fast or normal, because based on it priorities will be allocated.
- The weight must also be entered correctly as based on that we fill our trucks.

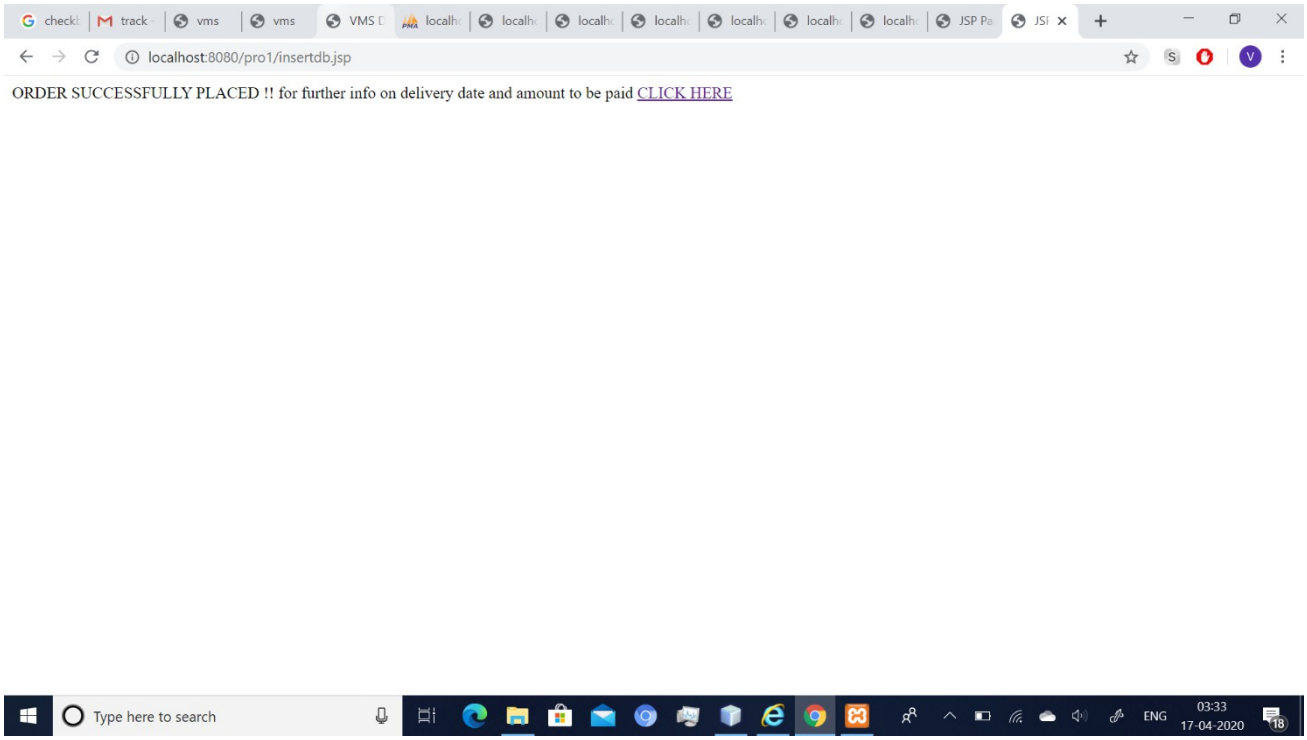


- THERE ARE THE EXAMPLE OF TWO CUSTOMERS WHO HAVE FILLED THEIR DETAILS





IF THE PLACE ORDER BUTTON IS CLICKED THEN THE BELOW PAGE APPEARS:



The click here button again joins with track order page.

Source code:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>VMS DELIVERY SYSTEM</title>
    </head>
    <body>
        <h1>CUSTOMER INFORMATION FORM</h1>
        <form name="custform" method="post" action="insertdb.jsp" >
            <fieldset>
                <legend style="padding:20px,0 ;font-size: 20px">ORDER
INFROMATION</legend>
                <div> <label> NAME</label>
                    <input type="text" value="name" class="btn"></div>
                <div><label>PHONE NUMBER</label>
                    <input type="number" name="phno" class="btn"></div>
                <div><label>PICK UP ADRESS</label>
                    <input type="textbox" name="pad" class="btn"></div>
                <div><label>DELIVERY ADDRESS</label>
                    <input type="textbox" name="dad" class="btn"></div>
                <div><label>DATE OF ORDER</label>
                    <input type="date" name="date" class="btn"></div>
                <div><label>DELIVERY OPTION</label>
                    <table id="deloption" border="1" cellpadding="1%">
                        <tr>
                            <th>delivery option</th>
                            <th>extra charge</th>
                        </tr>
                        <tr>
                            <td>SPEED DELIVERY</td>
                            <td>10% EXTRA</td>
                        </tr>
                        <tr>
                            <td>NORMAL DELIVERY</td>
                            <td>0% EXTRA</td>
                        </tr>
                    </table>
                </div>
                <div>
                    <input type="checkbox" id="deloption1" name="deloption1" value="speed">
                    <label for="deloption1">SPEED DELIVERY</label><br>
```

```
                <input type="checkbox" id="deloption2" name="deloption1" value="normal">
                <label for="deloption1">NORMAL DELIVERY</label><br>
            </div>
            <div>
                <label>ORDER WEIGHT[shud be less than 7kg]</label>
                <input type="number" name="weight" class="btn">
            </div>
            <div>
                <legend>EMAIL ID</legend>
                <input type="email" name="email" class="btn">
            </div>
            <div>
                <button type="submit" name="action"  value="orderplace">PLACE
ORDER</button>
                <a href url="insertdb.jsp">place order</a>
            </div>
        </fieldset>

    </form>
  </body
</html>
```

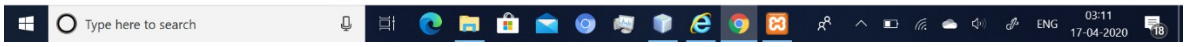
TRACK ORDER:

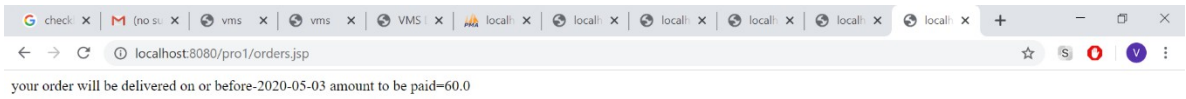
This page appears if the customer chooses to see when their order is delivered and the amount which they must pay.

PRICE CALCULATION:

The standard price for an order is calculated by multiplying the each kg by 15.
Incase of fast delivery,10% of the standard price is added extra.
For normal delivery,the standard price is the final amount to be paid by the customer.

The below picture is given with an example

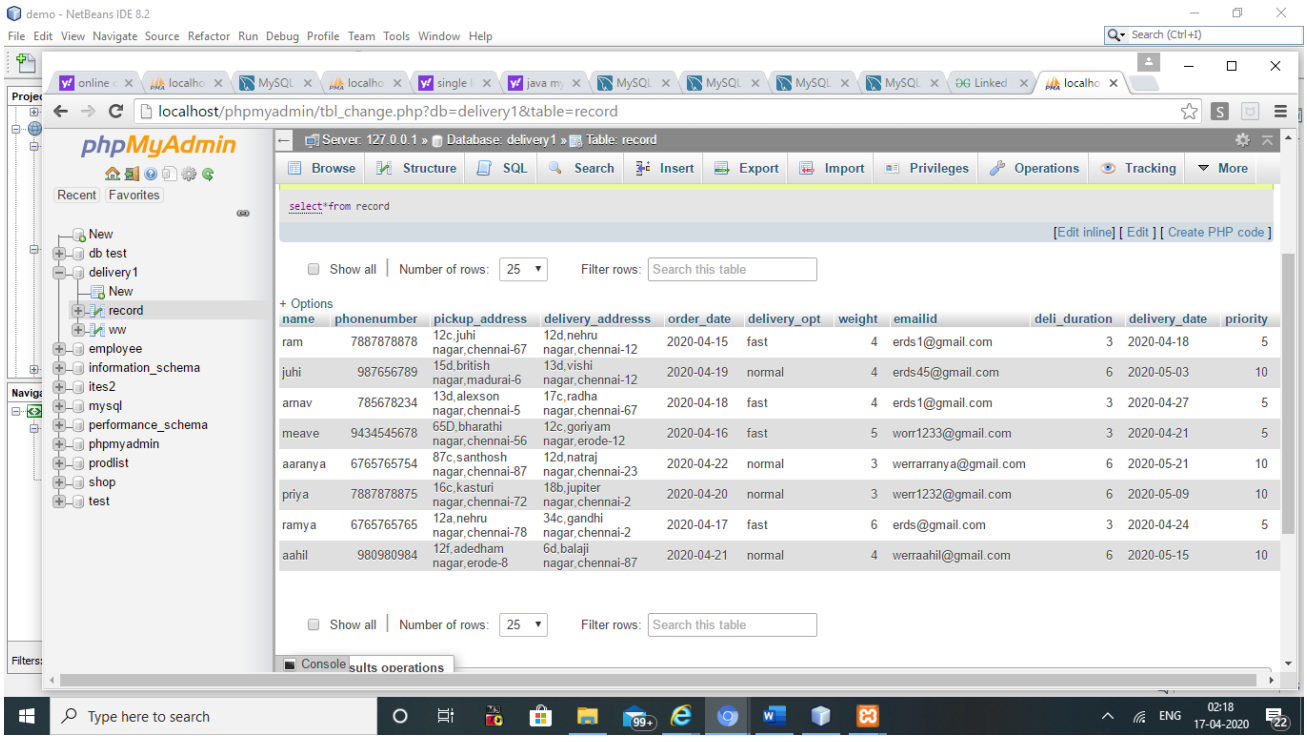




Source code:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>VMS DELIVERY SYSTEM</title>
    </head>
    <body>
        <form name="trackform" action="track2.jsp" method="post">
            <fieldset>
                <legend>track info</legend>
                <div>
                    <label>NAME</label>
                    <input type="text" name="name">
                </div>
                <div>
                    <label>PHONE NUMBER</label>
                    <input type="number" name="phonenumber">
                </div>
                <div>
                    <button type="submit" onclick="" name="sigin">GO</button>
                </div>
            </fieldset>
                    </form>
    </body>
</html>
```

THE FINAL UNORDERED DATA:



SOURCE CODE FOR INSERTING FROM FORM INTO DATABASE

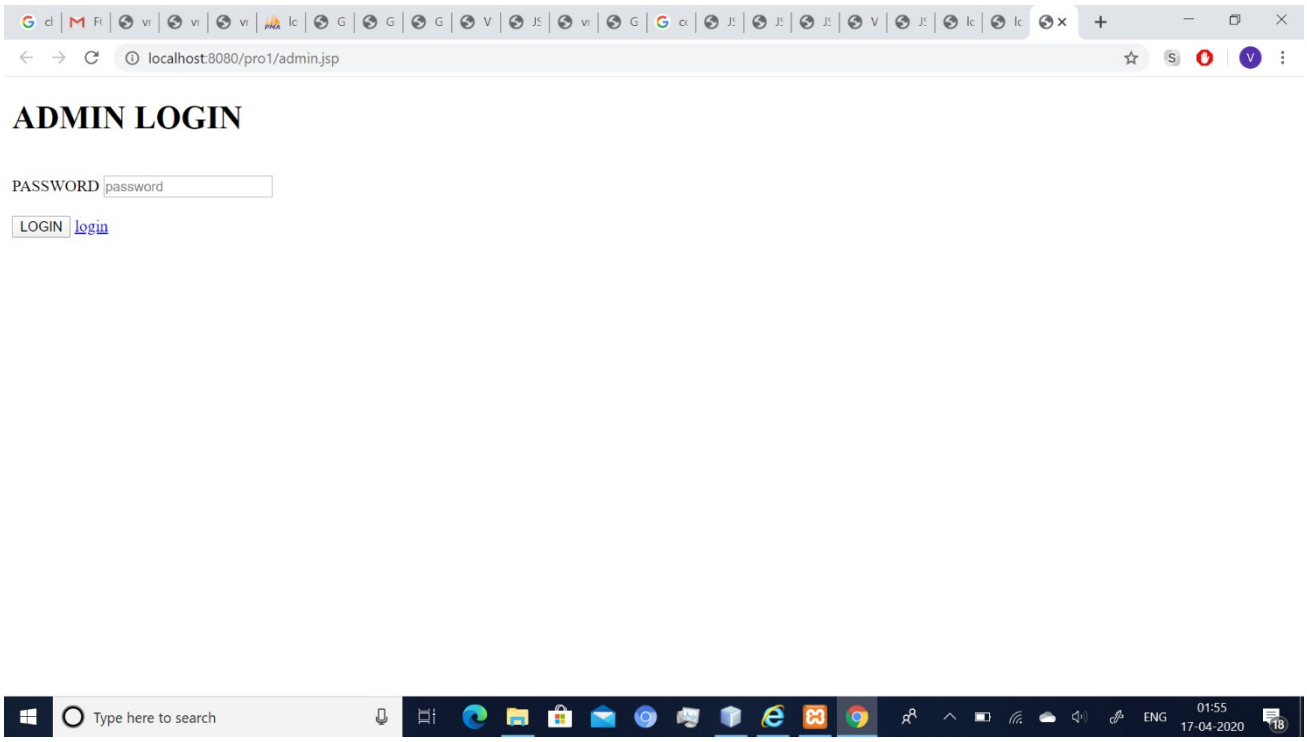
```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <%@page import="java.sql.*" %>
        <%@page import="java.util.Date" %>
       <%@page import="java.text.SimpleDateFormat" %>
         <%
         String name1=request.getParameter("nm");
         String phoneno=request.getParameter("phno");
         //int pn=Integer.parseInt(phoneno);


         String paddr=request.getParameter("pad");
         String daddr=request.getParameter("dad");
         String od=request.getParameter("date");
        //SimpleDateFormat f1=new SimpleDateFormat("dd-mm-yyyy");
        //Date d=f1.parse(od);


         String delmode=request.getParameter("deloption1");




         String wt=request.getParameter("weight");
         //int weight=Integer.parseInt(wt);

         String email=request.getParameter("email");
```

```
        try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/delivery1","ro
ot","");
        Statement st=con.createStatement();
       st.executeUpdate("insert into
record(name,phonenumber,pickup_address,delivery_address,order_date,deliv
ery_opt,weight,emailid)values('"+name1+"','"+phoneno+"','"+paddr+"','"+daddr
+"','"+od+"','"+delmode+"','"+wt+"','"+email+"')");


       out.println(" ORDER SUCCESSFULLY PLACED !!");
       out.println("for further info on delivery date and amount to be paid");

         }catch(Exception e){out.println(e);}%>
         <a href url="track.jsp">CLICK HERE</a>
    </body>
</html>
```

ADMIN:

This page appears if the person chooses admin option from the home page.

Where the administrator enters the password and validation occurs



Source code:

```
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>VMS DELIVERY SYSTEM</title>
   </head>
   <body>
      <h1>ADMIN LOGIN</h1>
      <BR>
      <form name="form2" action="admin2.jsp" method="post">
         <div>
            <label>PASSWORD</label>
            <input type="password" class="form-control" name="password"
placeholder="password">
         </div>
         <div>
            <br>
            <button name="submit" onclick="admin2.jsp" value="admin"
class="btn">LOGIN</button>
            <a href="admin2.jsp">login</a>
         </div>
      </form>
```

```
    </body>
</html>
```

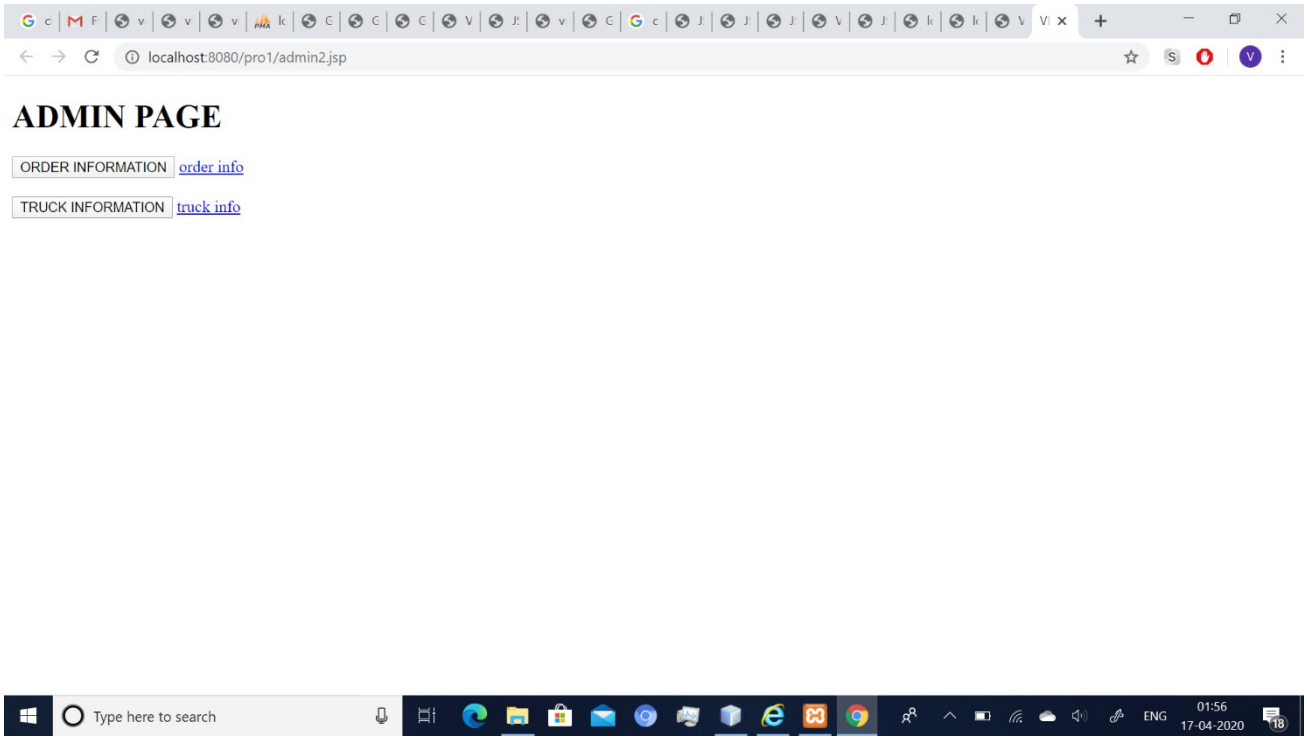
ADMIN PAGE:

This page appears if the password given by the admin matches the systems password.

In this page the admin has two options to choose:

1) Order information
2) Truck information

Admin can view all the users scheduled date for delivery of the packages and the truck management .



Source code:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>VMS DELIVERY SYSTEM</title>
    </head>
    <body>
        <h1>ADMIN PAGE</h1>
        <form name="form3" method="post">
            <div>
                <button type="submit" name="button1" onclick="orders.jsp" value="order"
class="btn">ORDER INFORMATION</button>
                <a href="orders.jsp">order info</a>
            </div>
            <div>
                <br>
                <button type="submit" name="button2" onclick="truck.jsp" value="order"
class="btn">TRUCK INFORMATION</button>
                <a href="truck.jsp">truck info</a>
                </br>
            </div>
        </form>
    </body>
</html>
```

ORDER INFORMATION:

This page appears if admin chooses to see scheduled delivery details based on the priority that the users selected.

name	phonenumber	pickup_address	delivery_address	order_date	delivery_opt	weight	emailid	deli_duration	delivery_date	priority
ram	788787887	12c,juhi nagar,chennai-67	12d,nerhu nagar,chennai-12	2020-04-15	fast	4	erds1@gmail.com	3	2020-04-18	5
maeve	943454567	65D,bharathi nagar,chennai-56	12c,goriyam nagar,erode-12	2020-04-16	fast	5	werr1233@gmail.com	3	2020-04-21	5
ramya	676576576	12a,nerhu nagar,chennai-78	34c,gandhi nagar,chennai-2	2020-04-17	fast	6	erds@gmail.com	3	2020-04-24	5
amav	785678234	13d,alexson nagar,chennai-5	17c,radha nagar,chennai-67	2020-04-18	fast	4	erds1@gmail.com	3	2020-04-27	5
juhi	987656789	15d,british nagar,madurai-6	13d,vishi nagar,chennai-12	2020-04-19	normal	4	erds45@gmail.com	6	2020-05-3	10
priya	788787887	16c,kasthuri nagar,chennai-72	18d,jupiter nagar,chennai-2	2020-04-20	normal	3	werr1232@gmail.com	6	2020-05-9	10
aahil	980980984	12f,adedham nagar,erode-8	6d,balaji nagar,chennai-87	2020-04-21	normal	4	werraahil@gmail.com	6	2020-05-16	10
aarayna	967657657	87c,santhosh nagar,chennai-87	12d,natraj nagar,chennai-23	2020-04-22	normal	3	werraarayna@gmail.com	6	2020-05-21	10

OS CONCEPT USED:

The scheduling algorithm implemented in our project is PRIORITY SCHEDULING(non-preemptive) .Based on the delivery option (fast/normal)opted by the customer the priority is set respectively.(i.e) in our database the orders with delivery option "fast" are given priority 5 and those with option "normal" are given priority ->10 .The date of placement of order is also mentioned in the form which is equivalent to the arrival time in the standard algorithm. The delivery duration is set based on the delivery option opted ,for fast delivery it is set as 3 ,and for normal delivery it is set as 6.These delivery duration are backend data and are accessed only by the system admin for the internal processing and scheduling. The delivery duration is equivalent to the burst time in the algorithm. The database is sorted based on the priority and the arrival time by which the orders with lower priority (i.e) 5=fast delivery are pushed to the top of the table. Based on the arrival time the orders are scheduled respectively. The approximate delivery date is hence manipulated by adding the delivery duration to the arrival time .

SOURCE CODE:

```
<%@page import="java.sql.*" %>
<%@page import="java.util.ArrayList" %>
<%@page import="java.util.Date" %>
<%@page import="java1.arrlist" %>
<%@page import="java.util.Calendar" %>
<% ArrayList<arrlist> mylist =new ArrayList<arrlist>();

    try{
        Class.forName("com.mysql.jdbc.Driver");
```

```
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/delivery1","root","");
        Statement st=con.createStatement();
        st.executeUpdate("update record set priority=5,deli_duration=3 where
delivery_opt='fast'");
        st.executeUpdate("update record set priority=10,deli_duration=6 where
delivery_opt='normal'");
        ResultSet rs=st.executeQuery("select*from record order by priority,order_date");
        Date t;
        int count=1;
       while(rs.next())
       {arrlist a=new arrlist();
       a.setname(rs.getString(1));
       a.setphno(rs.getString(2));
       a.setpad(rs.getString(3));
       a.setdad(rs.getString(4));
       a.setod(rs.getDate(5));
       a.setdelop(rs.getString(6));
       a.setwt(rs.getInt(7));
       a.setemail(rs.getString(8));
       a.setdd(rs.getInt(9));
       a.setdeld(rs.getDate(10));
       Date d=a.getdeld();
       a.setpri(rs.getInt(11));
       Calendar c=Calendar.getInstance();
       if(count==1)
       {
       c.setTime(a.getod());
       }
       else
           c.setTime(d);

       c.add(Calendar.DAY_OF_MONTH,a.getdd());

       a.setdeld(c.getTime());
       d=a.getdeld();
       count=count+1;




       mylist.add(a);
       }
       int i;

        out.println("NAME          PHONE NUMBER            PICKUP_ADD
 DELIVERY_ADD          ORDER_DATE            DELIVERY_OPT            WEIGHT
 EMAILID             DELI_DURATION             DERLIVERY_DATE             PRIORITY");

       for(arrlist s:mylist)
```

```
    {
        out.println(s.name+"
                "+s.phonenumber+"
                "+s.padd+"
                "+s.dadd+"
                "+s.orderd+"
         "s.delopt+"
        "+s.weight+"
        "+s.email+"
        "+s.deldur+"
        "+s.deldate+"
        "+s.pri);


    }

}
catch(Exception e)
{
    out.println(e);}
%>
```

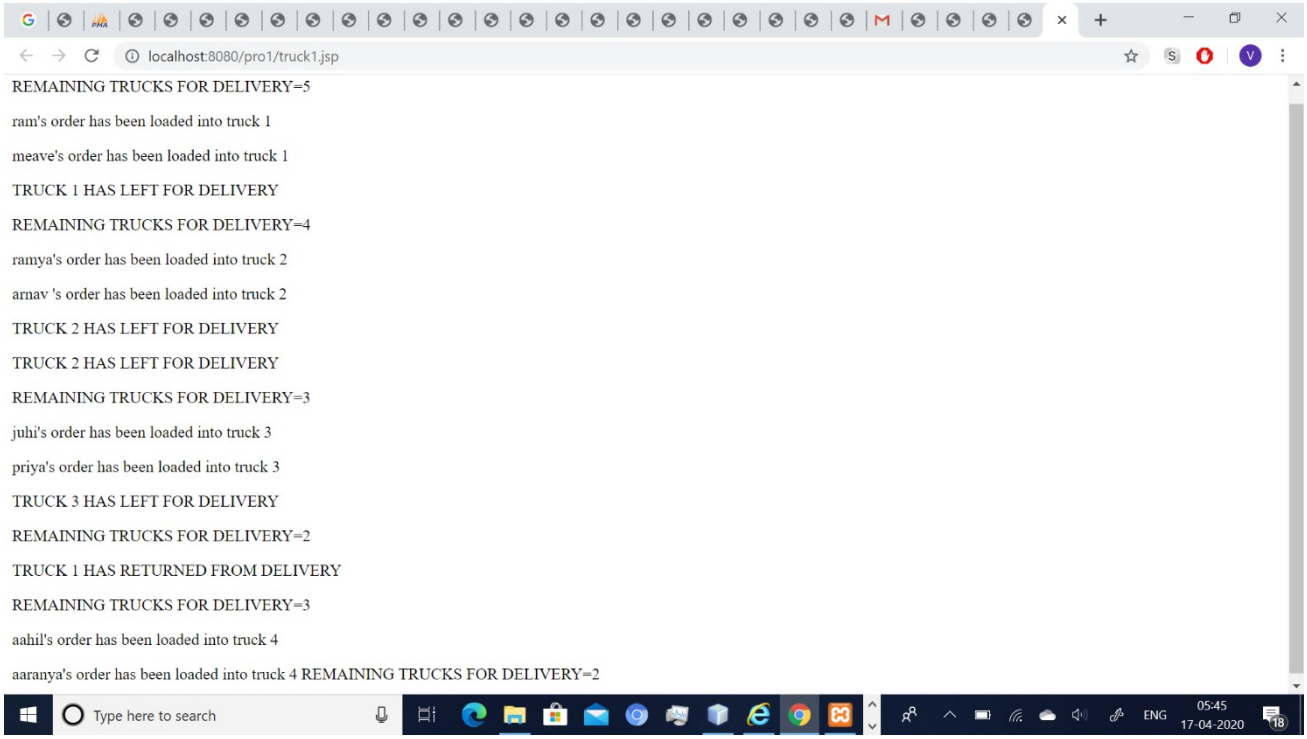
TRUCK INFORMATION:

Truck information provides details about the orders currently being shipped by the respective trucks and the return status of the trucks and number of trucks waiting to be loaded.

OS CONCEPTS USED:

1.We have implemented **BOUNDED BUFFER CONCEPT** for the truck capacity which is manipulated by the weight of the order. The bound is the capacity of each truck in our case. Every truck has a capacity of 10kg.As and when the orders are filled, the truck capacity decreases. When the truck is full, no more orders can be assigned to it. When the truck is empty, it cannot be sent out for delivery.

2.It also includes **SEMAPHORE CONCEPT** where the availability of trucks for the delivery is checked. Here, the trucks are equivalent to resources available. Every time a truck is sent out for delivery, the number of trucks available is decremented by1.Every time a truck comes back after delivering all orders assigned to it, the number of trucks available is incremented by 1.



Source code:

```
 <%@page import="java.sql.*" %>
      <%@page import="java.util.Date" %>
     <%@page import="java.text.SimpleDateFormat" %>

   <%   try{
       Class.forName("com.mysql.jdbc.Driver");
       Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/delivery1","root","");
       Statement st=con.createStatement();
       String sname=request.getParameter("lname");
       String phoneno=request.getParameter("lphonenumber");

      ResultSet rs=st.executeQuery("select *from record");
      int no_truck=5;
      int t1=10,t2=10,t3=10,t4=10,t5=10;
      int arr[]=new int[8];
      String a[]=new String[8];
      int i=0,j=0;
      while(rs.next())
```

```
{
    arr[i]=rs.getInt(7);
    a[i]=rs.getString(1);
    i++;
}
for(j=0;j<arr.length;j++)
{
    out.println(arr[j]+" "+a[j]);
}
i=0;
*while((t1-arr[i]>=0) && i<arr.length)
{
    t1=t1-arr[i];
    out.println(a[i]+"order has been loaded in truck_1");
    i++;
}
out.println("TRUCK 1 HAS LEFT FOR DELIVERY");
if((t1-arr[i]<0) && no_truck>=0)
    no_truck--;
out.println("REMAINING TRUCKS FOR DELIVERY="+no_truck);
while(t2-arr[i]>=0 && i<arr.length)
{
    t2=t2-arr[i];
    out.println(a[i]+"order has been loaded in truck_2");
    i++;
}
out.println("TRUCK 2 HAS LEFT FOR DELIVERY");
if(t2-arr[i]<0 && no_truck>=0)
    no_truck--;
out.println("REMAINING TRUCKS FOR DELIVERY="+no_truck);
    while(t3-arr[i]>=0 && i<arr.length)
{
    t3=t3-arr[i];
    out.println(a[i]+"order has been loaded in truck_3");
    i++;
}
 out.println("TRUCK 3 HAS LEFT FOR DELIVERY");
 out.println("TRUCK 1 HAS RETURNED FROM DELIVERY");
 no_truck++;
if(t3-arr[i]<0 && no_truck>=0)
    no_truck--;
out.println("REMAINING TRUCKS FOR DELIVERY="+no_truck);


while(t4-arr[i]>=0 && i<arr.length)
{
```

```
        t4=t4-arr[i];
        out.println(a[i]+"order has been loaded in truck_4");
        i++;
        if(i==8)
            break;
    }
    out.println("TRUCK 4 HAS LEFT FOR DELIVERY");
    if(i<arr.length && t4-arr[i]<0 && no_truck>=0)
        no_truck--;
    out.println("REMAINING TRUCKS FOR DELIVERY="+no_truck);

}
catch(Exception e)
        {
            out.println(e);
        }
%>
```

CONCLUSION:

The project successfully implements the Operating system concepts namely Priority scheduling for the proper arrangement of the orders based on the delivery option preferred by the user ,bounded buffer concept is implemented in the truck capacity which corresponds to the weight of the orders and finally the semaphore concept is used the availability of the trucks for delivery and the price is henceforth calculated by manipulating the weight of the order + delivery option charges.