College Commute System

A PROJECT REPORT
Submitted by

Sanjana S 19MIS1141

in partial fulfillment for the award of the degree of

Master of Technology

in

Software Engineering (5 Year Integrated Programme)

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Abstract :

Usually colleges provide transport which operates at scheduled timings. This may cause a problem for some students, faculties, any workers in case of any personal emergencies. The other reason may be that students and faculties may not have classes for the whole day but since the college bus operates at scheduled time they have to wait for it. The system which we are developing might be helpful for these people to utilise a cab or auto or to reach to their exact or nearby destinations. In this project the user can book tickets in case they choose a cab or in auto. It provides a simple interface which can be easily used and accessible to all without any guidance. It is a very user-friendly software by which people can book necessary vehicles with comfort from college to home and vice versa.

Introduction :

In this paper, we are going to provide an simple way for the online cab booking system. Our application can be used by customers. It is mandatory for the customer to register in the application. The driver should register with their first name , last name , date of birth, phone number, setting login password and confirming it. etc. If customer wants a ride it is mandatory for customer to send customer's details and destination. After that customer can view distance between source and destination on the map Their data is stored in encrypted format in database so no other person or administrator can see their information. When particular user wants to use their own data then data will be decrypted. Main advantage of that application is that customer has to pay for only one way trip, so that customers can save their money and one more advantage is that, it is time efficient and always available for customers.

Problem Statement :

"The transport problems students or any other member in college faces due to uneven college timing or due to ambivalent dispersal of classes or any other uncertainty in the usual schedule ."

Existing System :

We have seen that most of these kind of applications need internet connection and signal tower for receiving One Time Password or payment through online mode. At some locations there wont be these signal availabilities.

- While we use any cab booking or auto booking system application we see that sometimes the cab might be available at very far distance or sometimes cabs won't be available to some destinations .
- Even today in some places there are no facilities for booking cab through phones or laptops . They still search for drivers from some driving agencies.
- In peak hours like school, office, railway stations, airports pickup and drop of customers is high and the drivers in such case look for profits and earn more money by increasing the fare of the cab.
- There might be some fake registrations access in the website or app which is harmful for the application .

Literature Review :

- EZcab is a proof-of-concept ubiquitous computing application that allows people to book nearby cabs using their cell phones or PDA's equipped with short-range wireless network interfaces. EZcab discovers and books free cabs using mobile ad hoc networks of vehicles. We have implemented an EZcab prototype on top of smart Messages, a middleware architecture based on execution migration, which we had developed to provide a common execution environment for outdoor ubiquitous computing applications. The experimental and simulation results have demonstrated the feasibility of Ezcab.

- Taxi industry is growing very rapidly. Everyone uses taxies for long or short distance journey instead of travelling by own car. In this paper an online recommendation model is presented to help the driver to decide about the best stand to head in each moment , minimizing the waiting time. The proposed approach uses time series advance techniques to predict the solution of real-time problem faced by taxi driver every day. We combine some information with the live current network status to produce our output. Also we are implementing taxi management system based on gps which has become an important tool for the sake of management, but also to provide useful information for taxi drivers to earn more profit by mining the historical GPS trajectories. In this paper, we propose a taxi recommender system for next cruising location which could be a value added module of the management system.

- The taxi dispatch problem involves assigning taxies to requester waiting at different locations. A dispatch system currently in use by a major cab service provider divides the city into regional dispatch areas. Each area has fixed assigned adjacent areas hand-coded by human experts . however , such fixed, hand coded adjacency of areas cannot be a prudent indicator because it does not take into consideration frequent changes in traffic patterns and road structure. The proposed system dynamically modifies the adjacency of dispatch areas. The proposed technique will decrease the total waiting time, is less in comparison with the results of the simulation without self-organization.

Proposed System :

- This is a simple user interface which can be used without any special guidance. It is a very user friendly software to use.
- In this system we have included the login page and creating a new login .
- After the login page we have give the vehicle selection page which is used to select auto or cab option.
- With respect to the vehicle selected we enter the details asked in that particular page and depending on the drop location and type of vehicle we have produced cost and availability of vehicle.
- After that page we get the confirmation page where we get the details for the cab or auto that we booked along with driver detail.

Objectives :

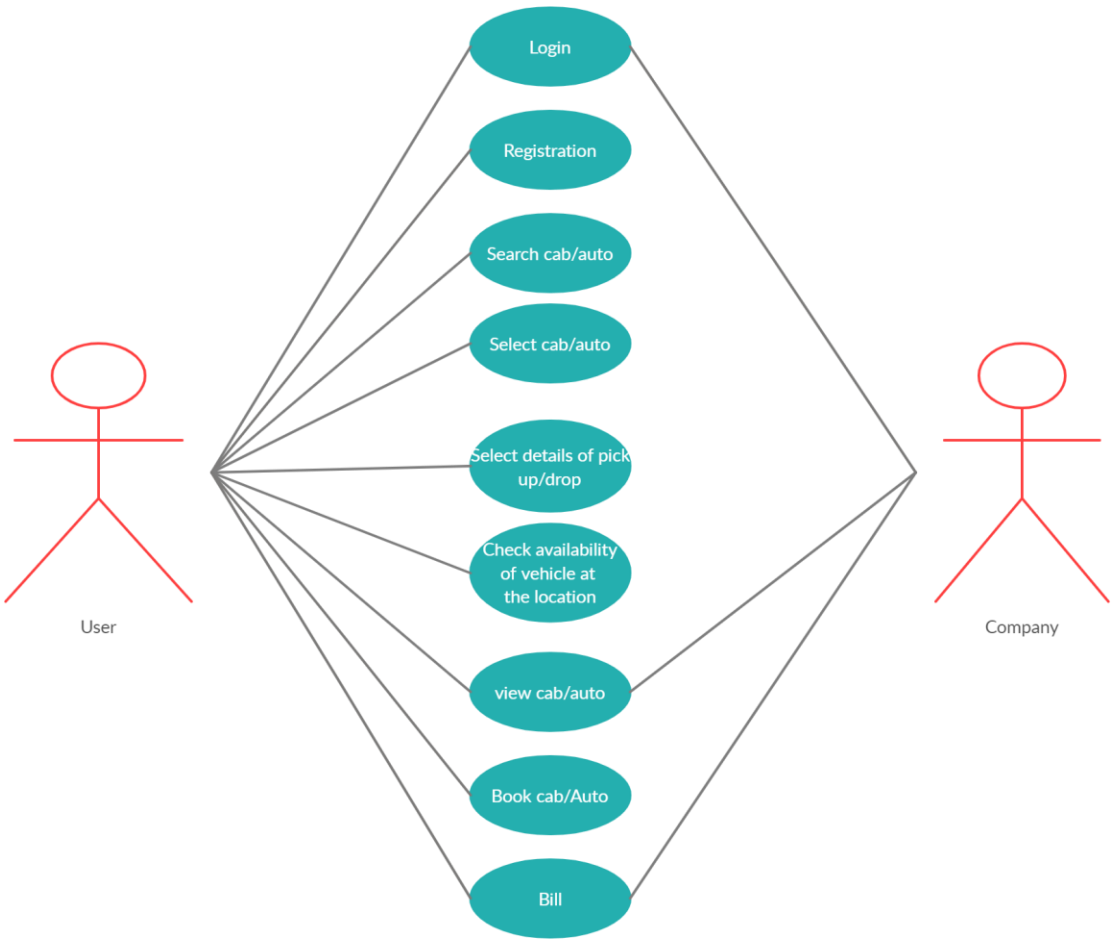
- The main objective of this application we have developed is the college class timings can be different for different faculties as well as students. The college provides transport only in fixed timings . So in case of any emergencies we need a mode of transportation to reach the nearby their destination.
- Sometimes we wont understand the steps to book the vehicle so the application we have developed is very easy to use without any guidance.
- The transport problems students or any other member in college faces due to uneven college timing or due to ambivalent dispersal of classes or any other uncertainty in the usual schedule .

Methadology :

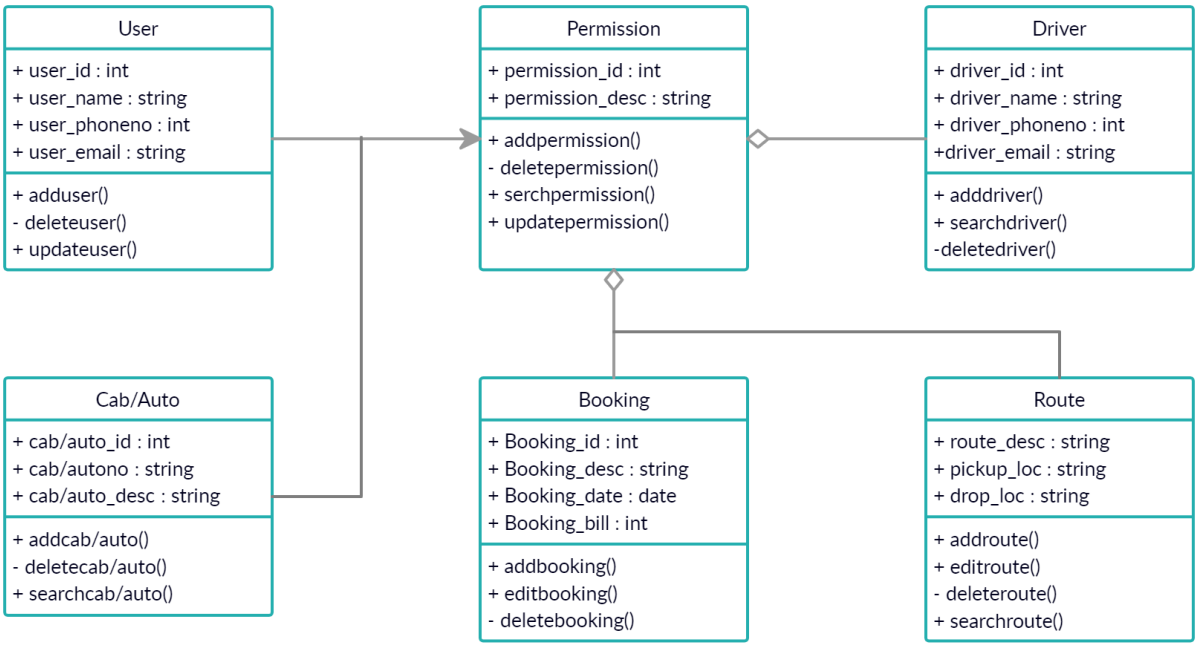
We have used netbeans software for developing our application using Jswing . For a better understanding we have drawn some basic diagrams on the application which we have developed.

Architectural Designs :

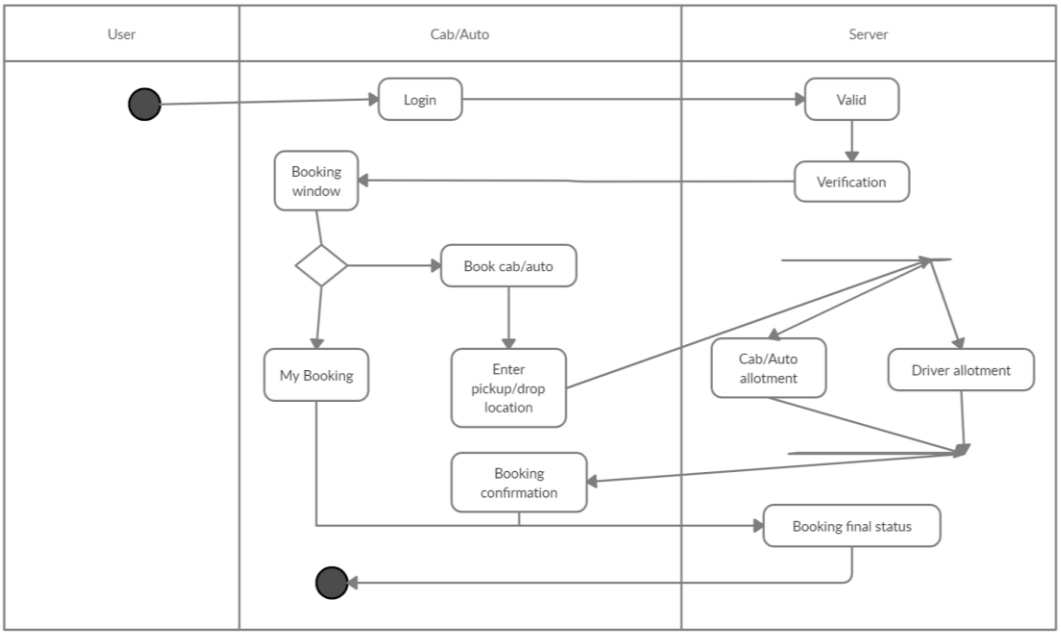
1. Use Case Diagram :



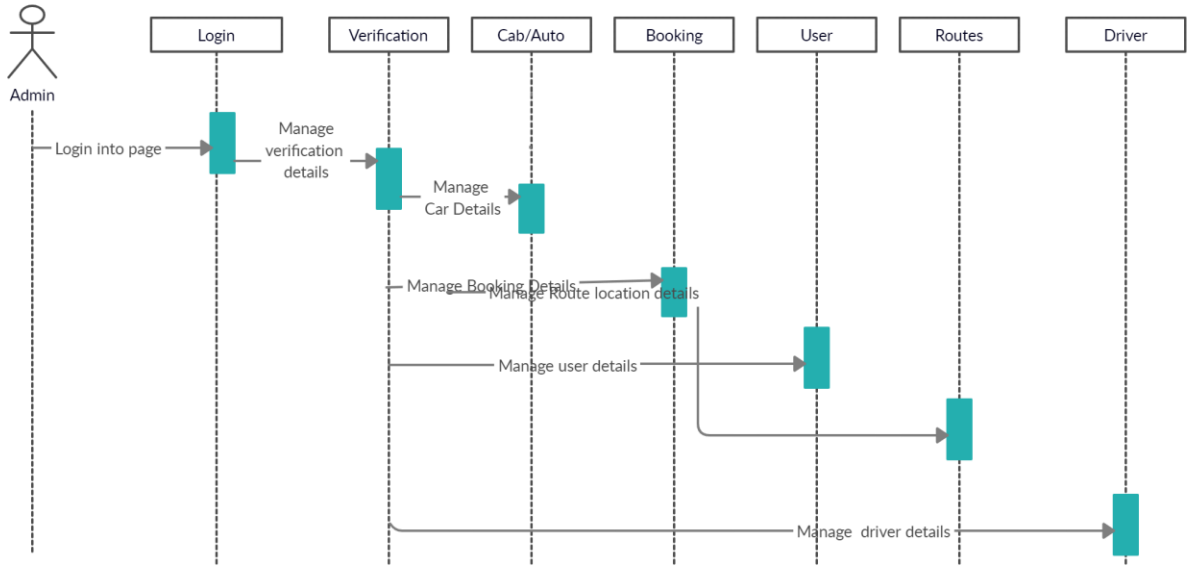
2.Class Diagram :



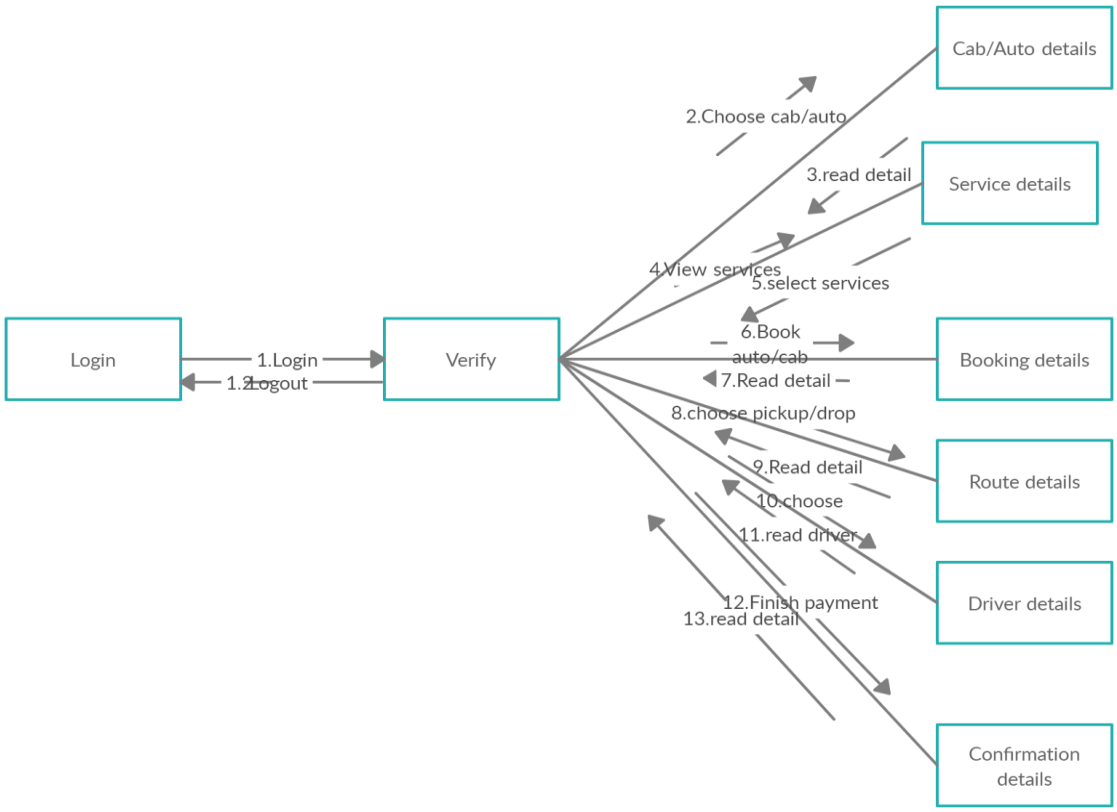
3.Activity diagram :



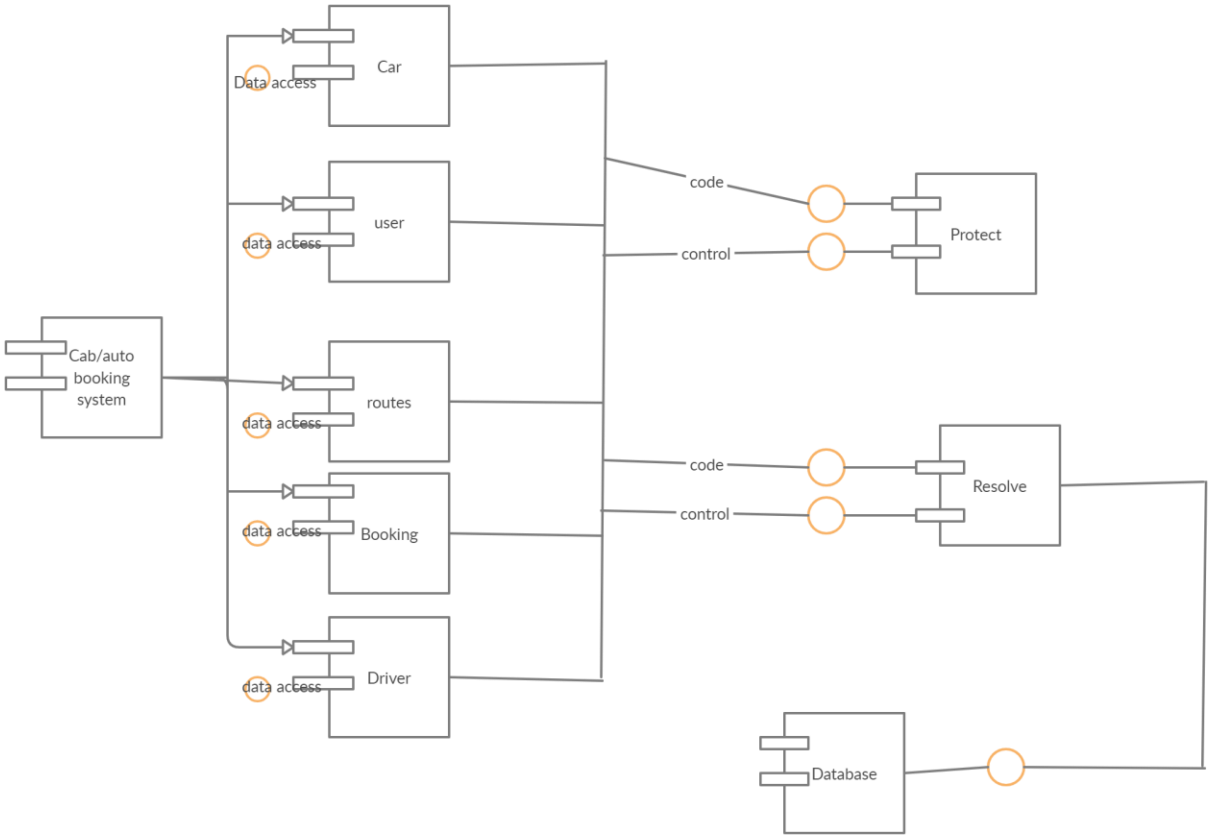
4.Sequence diagram :



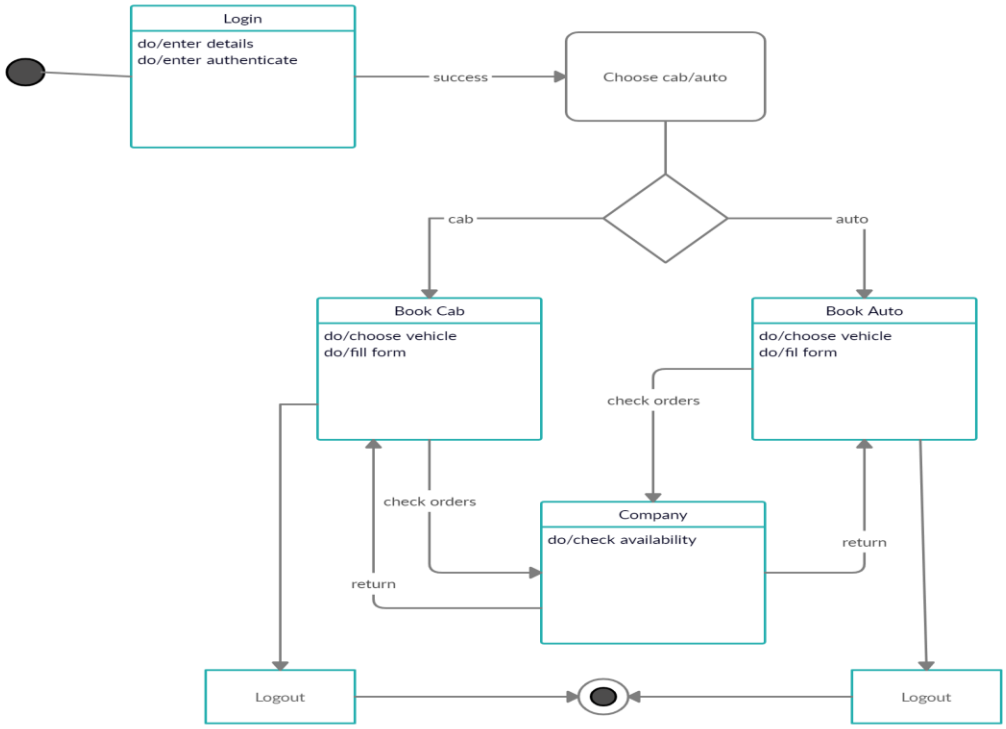
5.Collaboration diagram :



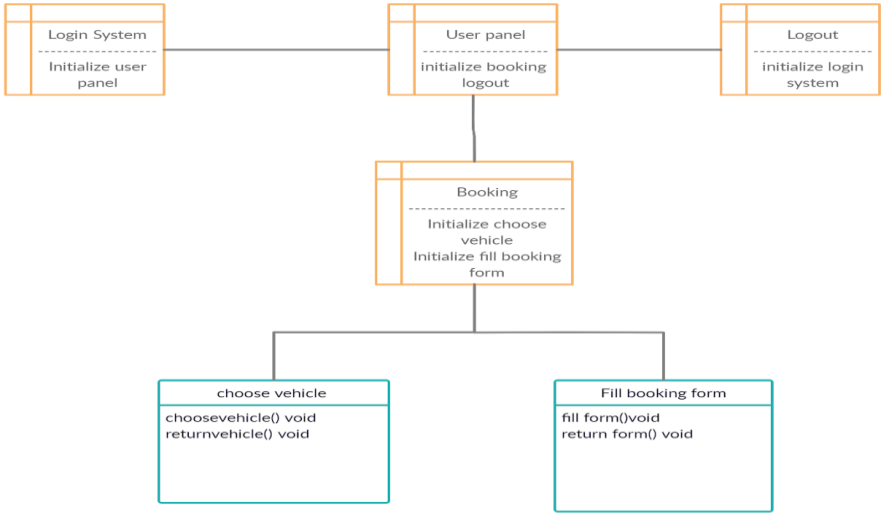
6.Component Diagram :



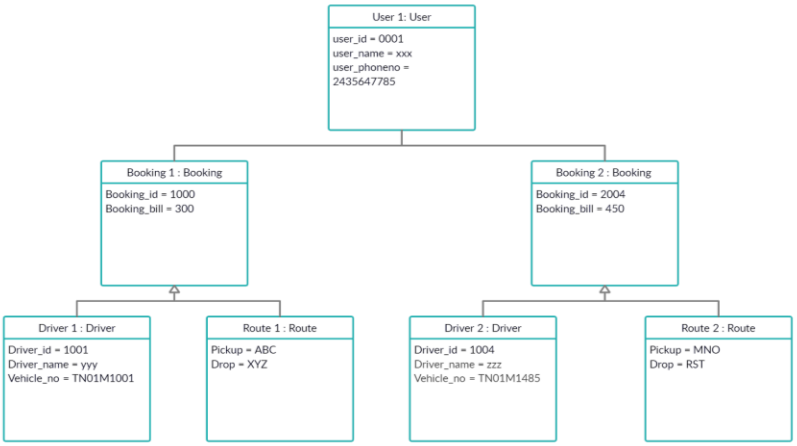
7.State chart :



8.Deployment diagram :

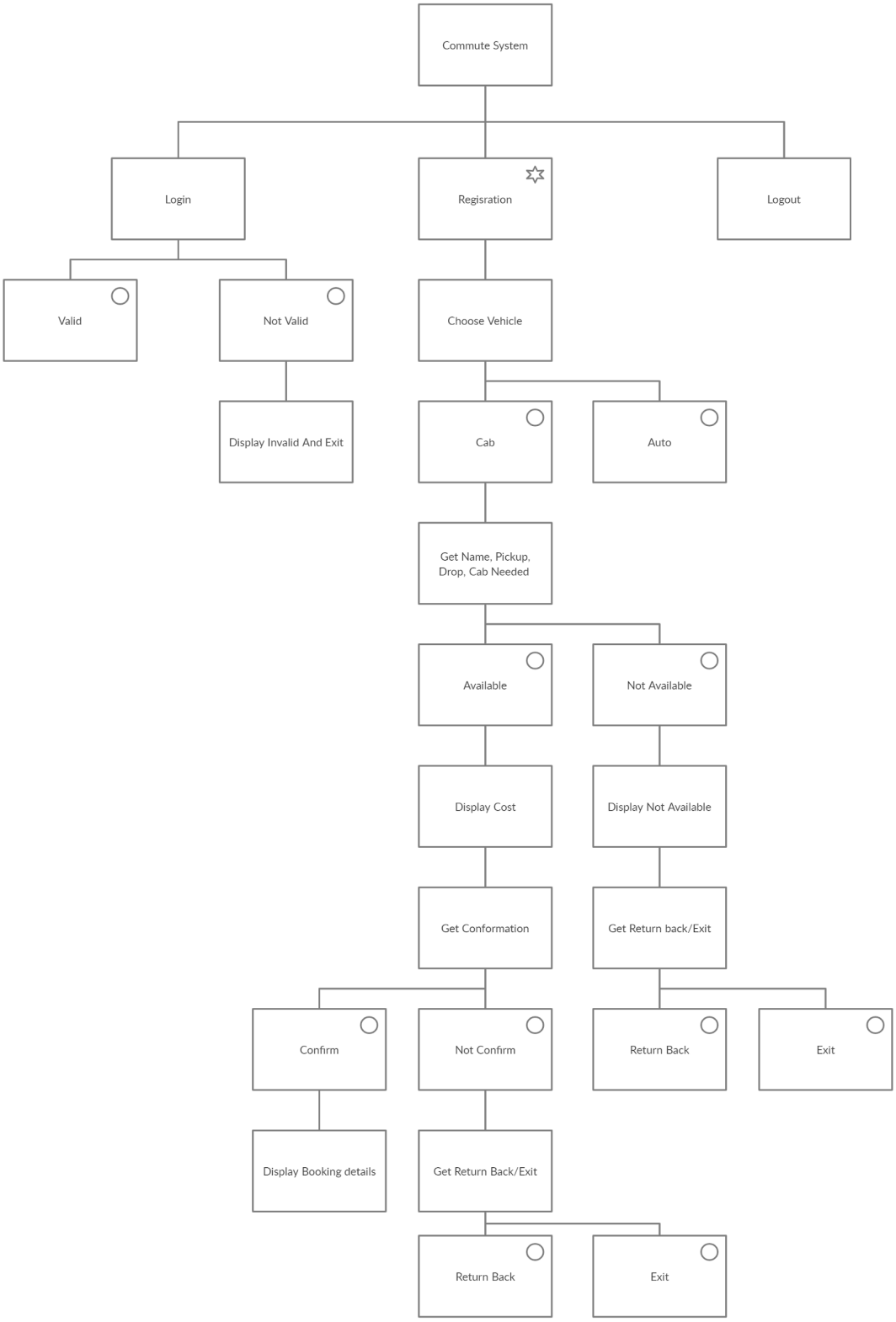


9.Object Diagram :

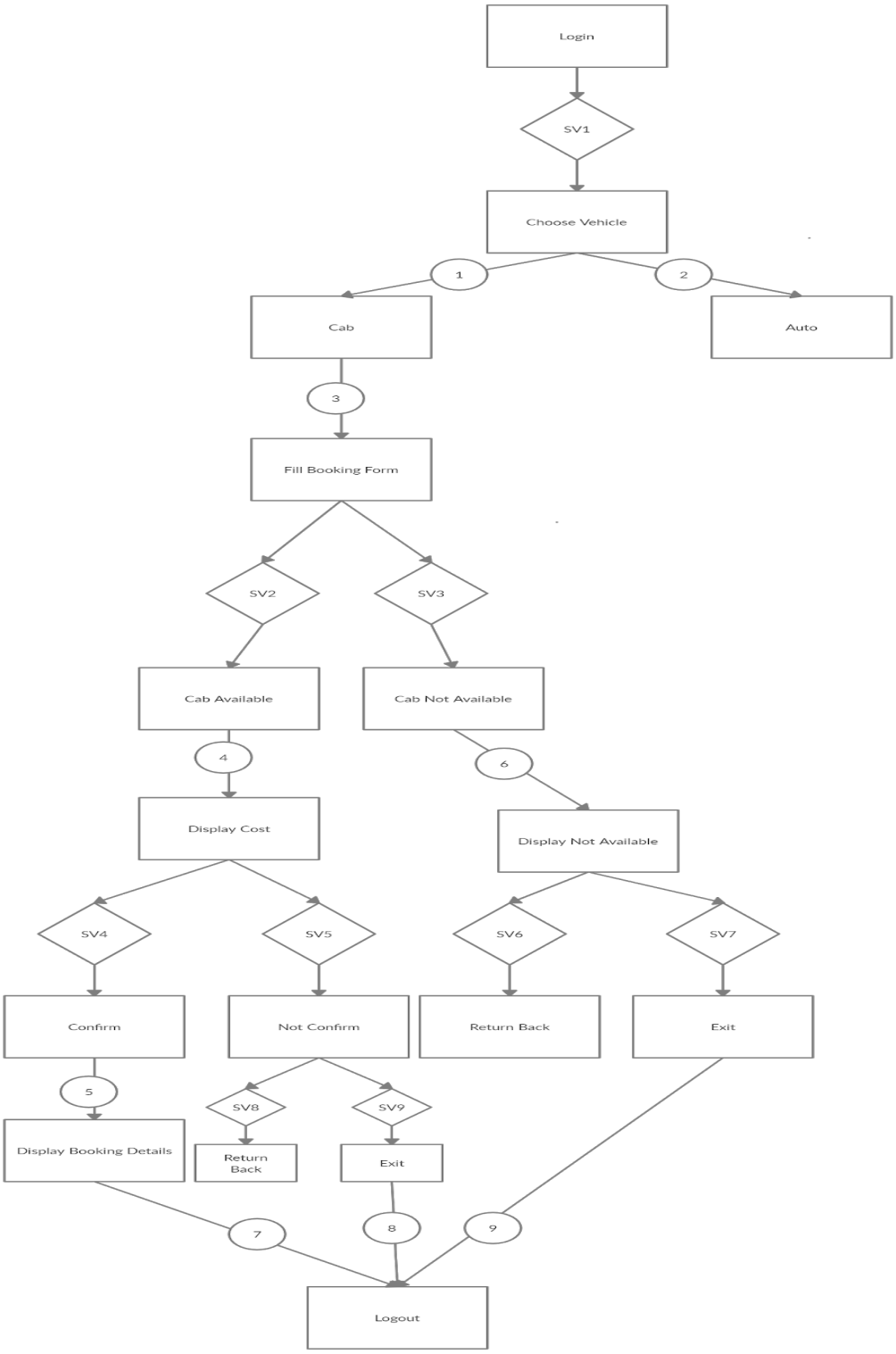


10. JSD :

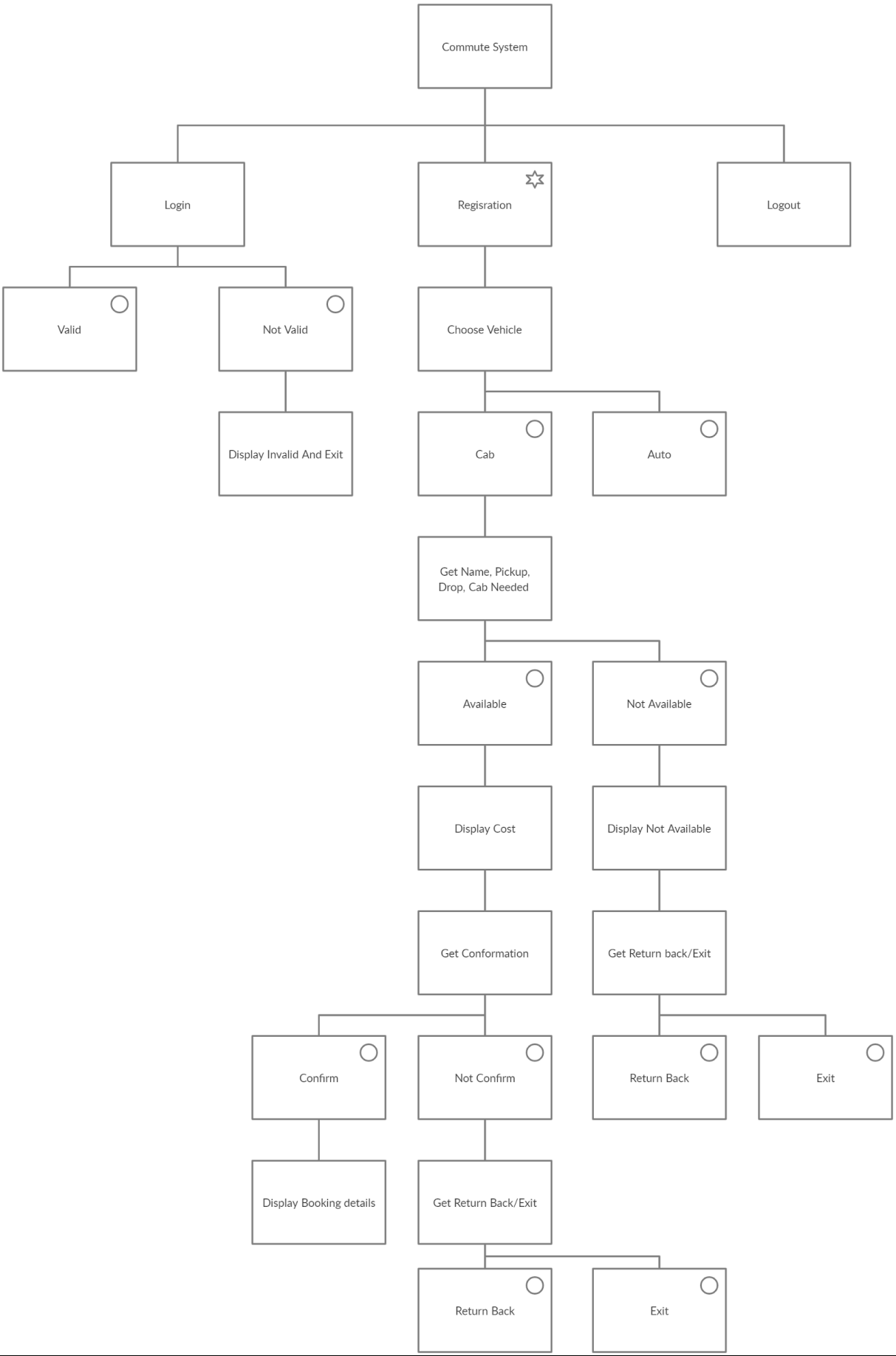
a ESD:


Commute System
Login
Regisration
Logout
Valid
Not Valid
Choose Vehicle
Display Invalid And Exit
Cab
Auto
Get Name, Pickup, Drop, Cab Needed
Available
Not Available
Display Cost
Display Not Available
Get Conformation
Get Return back/Exit
Confirm
Not Confirm
Return Back
Exit
Display Booking details
Get Return Back/Exit
Return Back
Exit

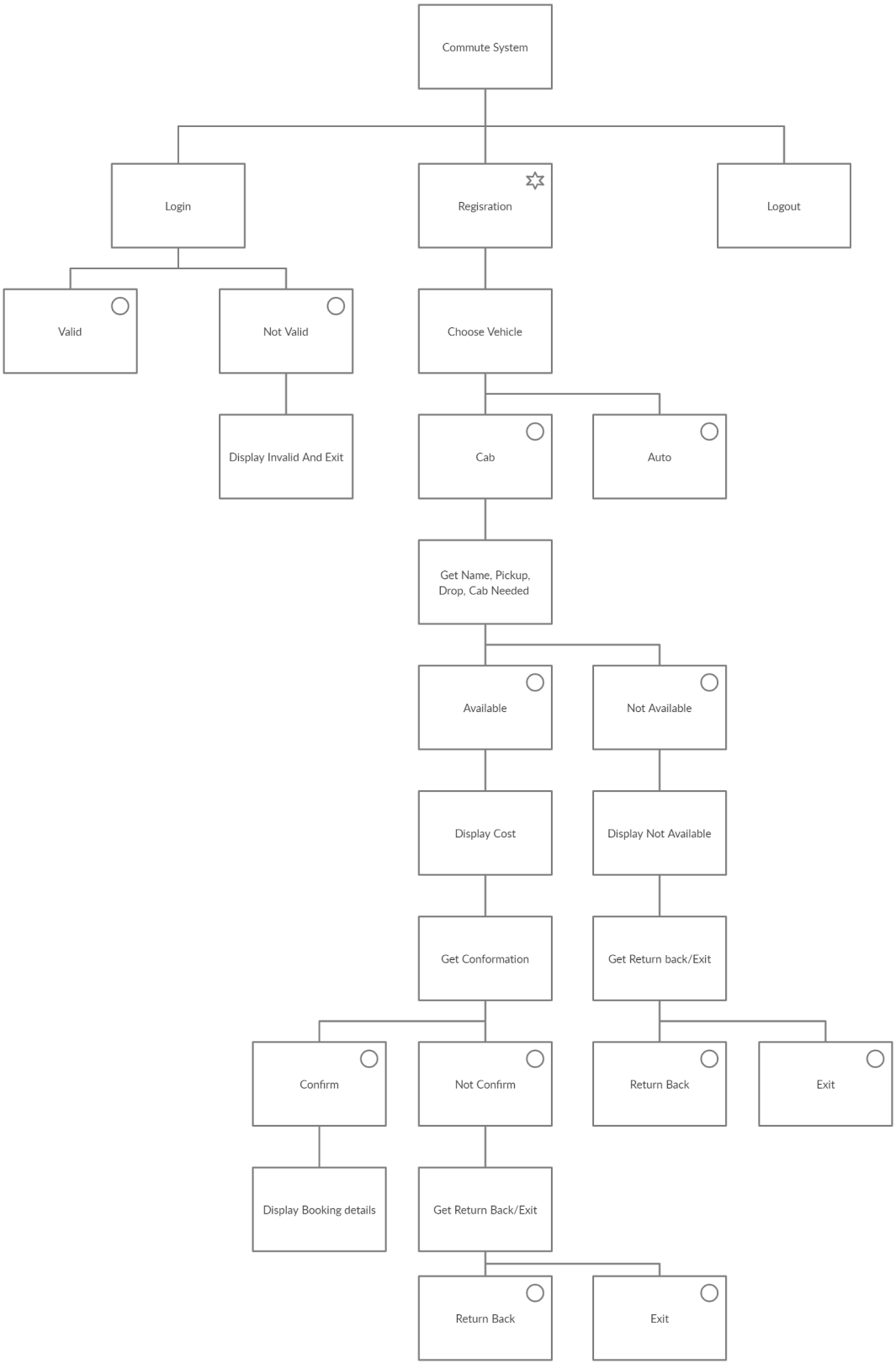
b SSD:


Login
SV1
Choose Vehicle
1
2
Cab
Auto
3
Fill Booking Form
SV2
SV3
Cab Available
Cab Not Available
4
6
Display Cost
Display Not Available
SV4
SV5
SV6
SV7
Confirm
Not Confirm
Return Back
Exit
5
SV8
SV9
Display Booking Details
Return Back
Exit
7
8
9
Logout

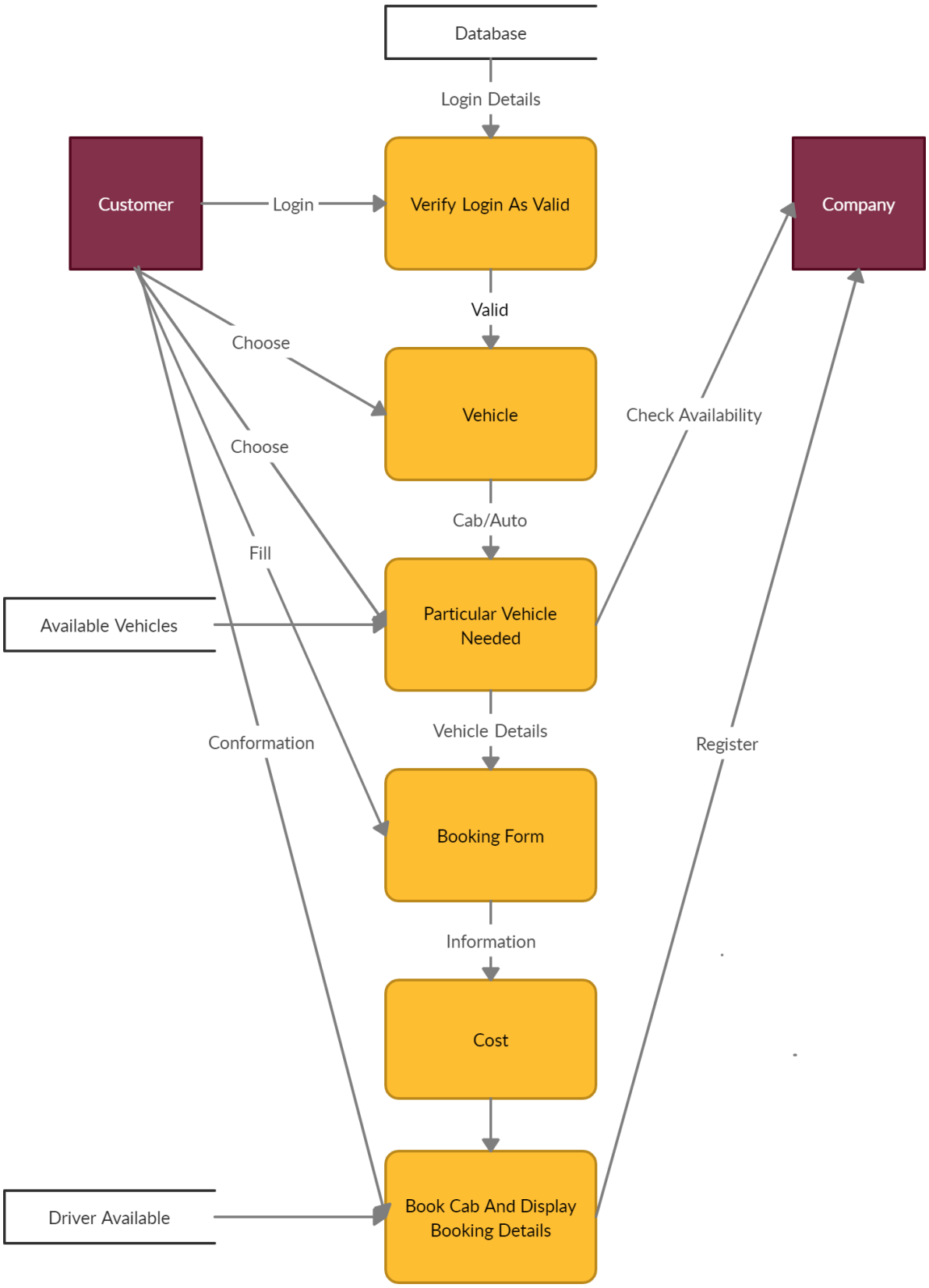
11 JSP :


Commute System
Login
Regisration
Logout
Valid
Not Valid
Choose Vehicle
Display Invalid And Exit
Cab
Auto
Get Name, Pickup, Drop, Cab Needed
Available
Not Available
Display Cost
Display Not Available
Get Conformation
Get Return back/Exit
Confirm
Not Confirm
Return Back
Exit
Display Booking details
Get Return Back/Exit
Return Back
Exit

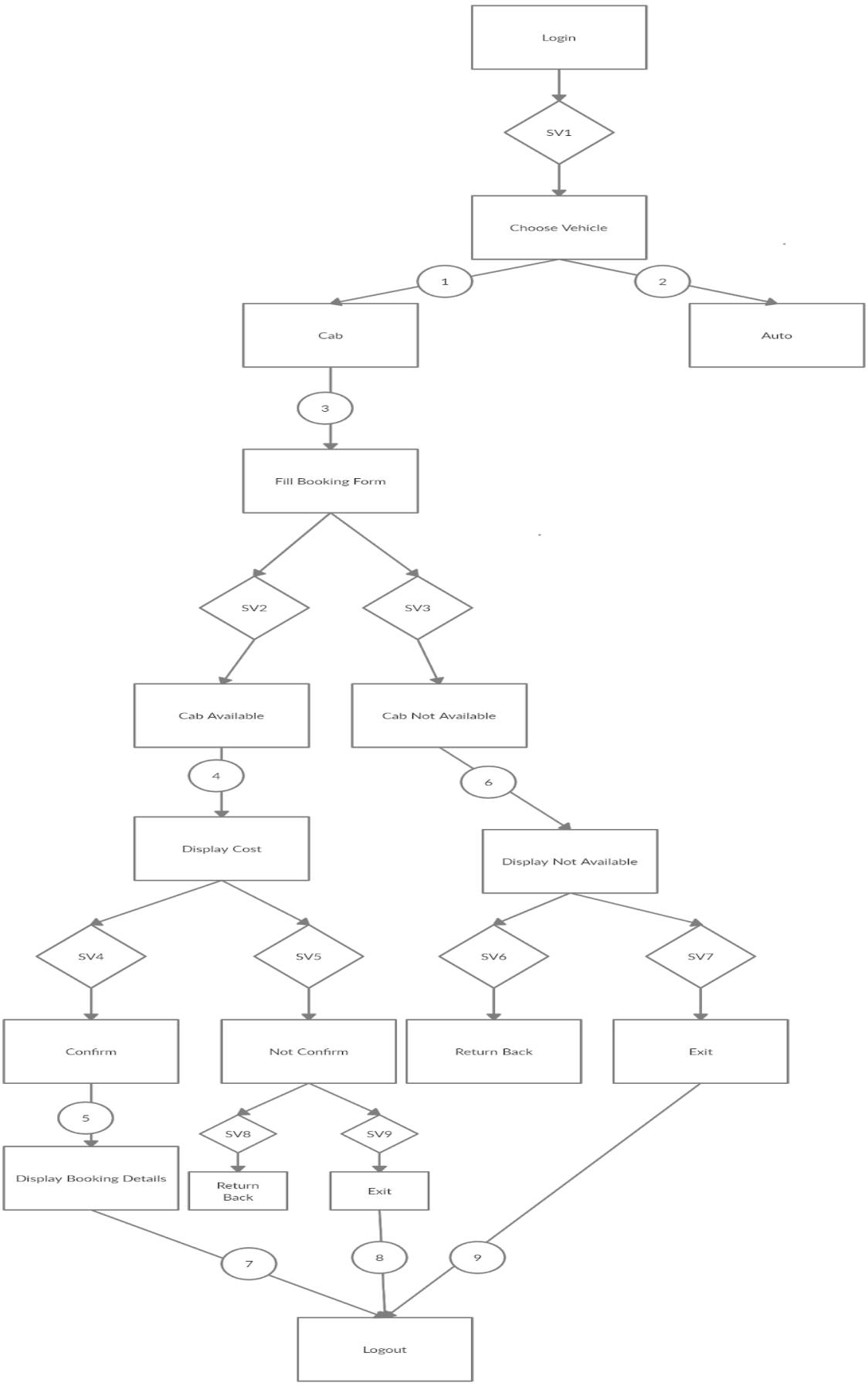
12 ESD :


Commute System
Login
Regisration
Logout
Valid
Not Valid
Choose Vehicle
Display Invalid And Exit
Cab
Auto
Get Name, Pickup, Drop, Cab Needed
Available
Not Available
Display Cost
Display Not Available
Get Conformation
Get Return back/Exit
Confirm
Not Confirm
Return Back
Exit
Display Booking details
Get Return Back/Exit
Return Back
Exit

13 DFD :

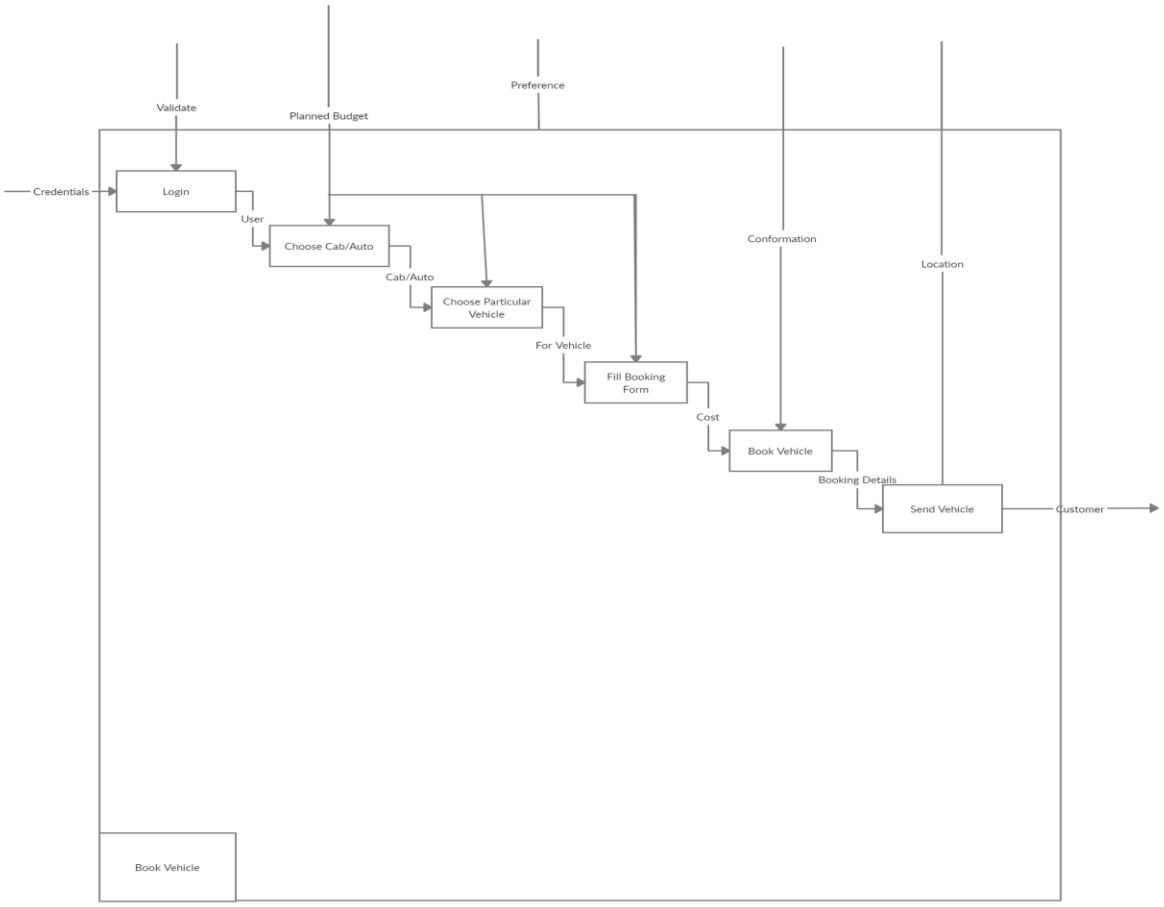

Database
Login Details
Customer
Login
Verify Login As Valid
Company
Valid
Choose
Vehicle
Check Availability
Choose
Cab/Auto
Fill
Available Vehicles
Particular Vehicle Needed
Vehicle Details
Conformation
Register
Booking Form
Information
Cost
Driver Available
Book Cab And Display Booking Details

14 SSD :

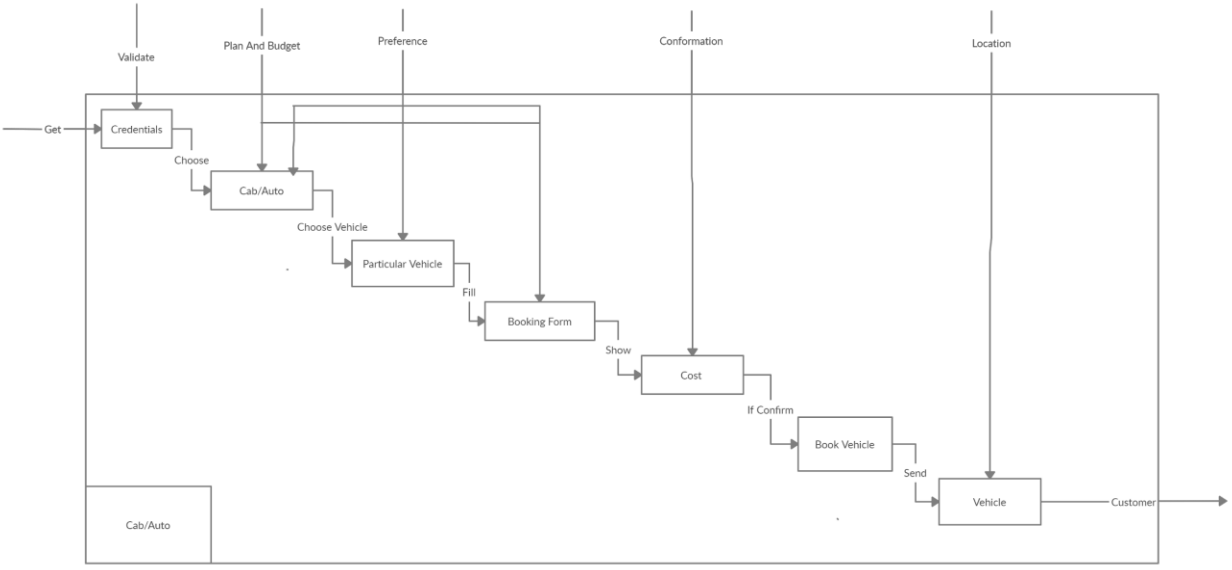

Login
SV1
Choose Vehicle
1
2
Cab
Auto
3
Fill Booking Form
SV2
SV3
Cab Available
Cab Not Available
4
6
Display Cost
Display Not Available
SV4
SV5
SV6
SV7
Confirm
Not Confirm
Return Back
Exit
5
SV8
SV9
Display Booking Details
Return Back
Exit
7
8
9
Logout

15 SADT :

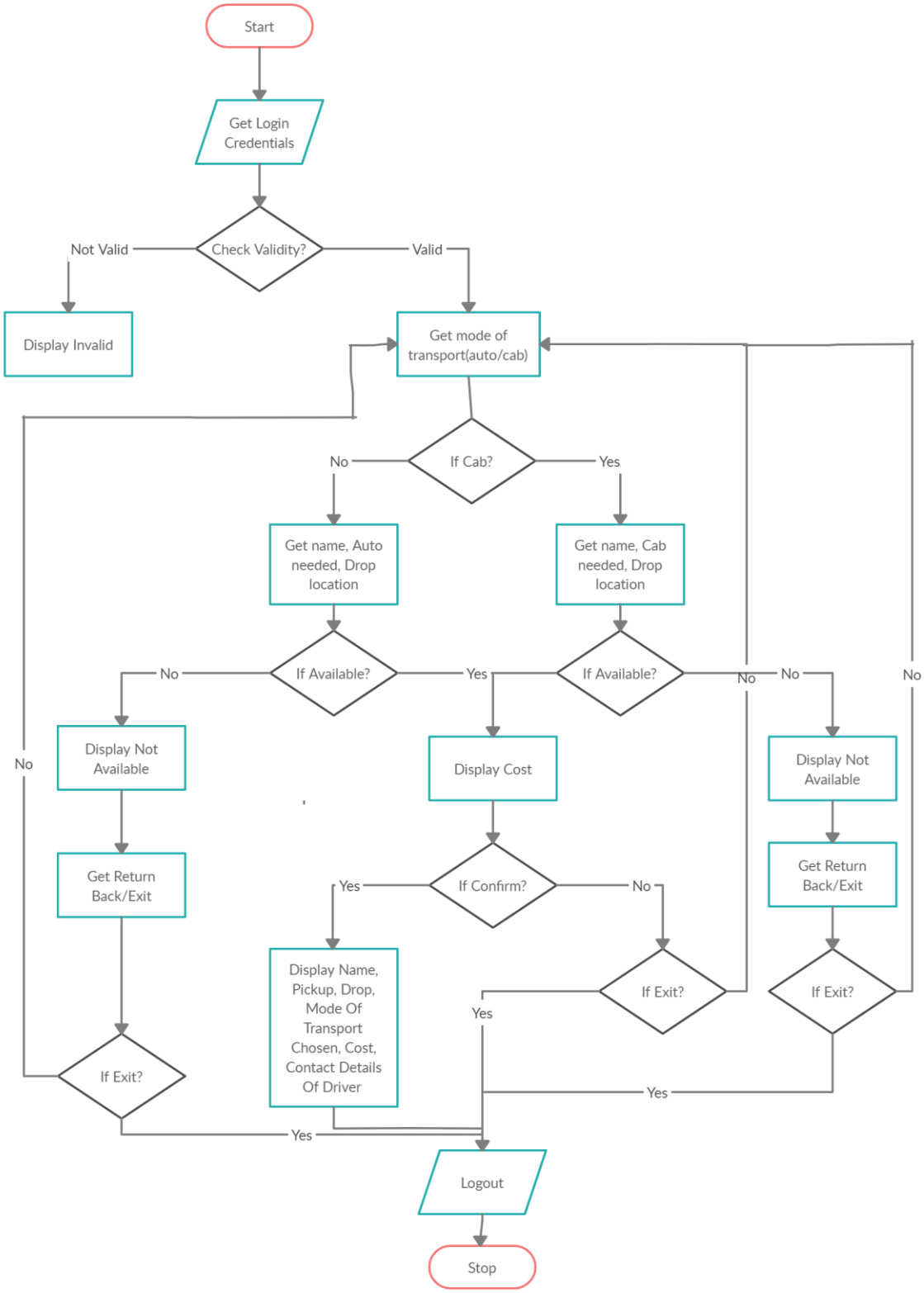
a.Activity Diagram :



b.Data Diagram :



16 Flowchart :



Sample Code :

LOGIN FORM:

CODE:-

```
import static java.time.Clock.system;
import javax.swing.JFrame;
import javax.swing.JOptionPane;
/*
* To change this license header, choose License Headers in Project Properties.
* To change this template file, choose Tools | Templates
* and open the template in the editor.
*/
/**
*
* @author Sanjana
*/
public class Loginform extends javax.swing.JFrame {
/**
* Creates new form Loginform
*/
public Loginform() {
initComponents();
this.setLocationRelativeTo(null);
}
Loginform(String v) {
throw new UnsupportedOperationException("Not supported yet."); //To change body of generated
methods, choose Tools | Templates.
}
/**
* This method is called from within the constructor to initialize the form.
* WARNING: Do NOT modify this code. The content of this method is always
* regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {
jPanel1 = new javax.swing.JPanel();
jLabelminimize = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
jLabelclose = new javax.swing.JLabel();
jPanel2 = new javax.swing.JPanel();
username = new javax.swing.JLabel();
password = new javax.swing.JLabel();
ufield = new javax.swing.JTextField();
pfield = new javax.swing.JPasswordField();
login = new javax.swing.JButton();
cancel = new javax.swing.JButton();
```

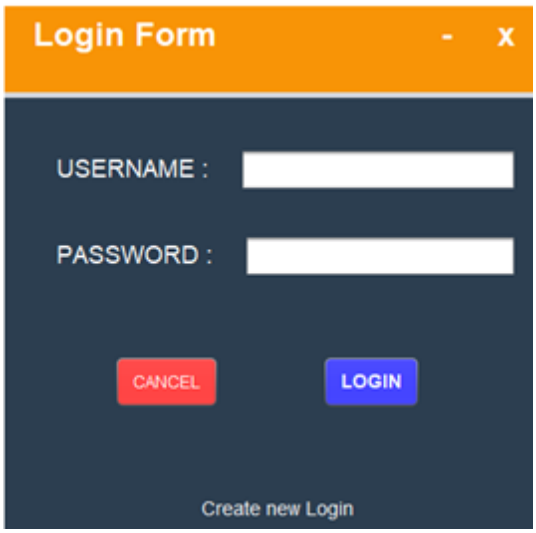
```
newlogin = new javax.swing.JLabel();
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setUndecorated(true);
jPanel1.setBackground(new java.awt.Color(248, 148, 6));
jLabelminimize.setFont(new java.awt.Font("Arial", 1, 24)); // NOI18N
jLabelminimize.setForeground(new java.awt.Color(255, 255, 255));
jLabelminimize.setText("-");
jLabelminimize.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
jLabelminimize.addMouseListener(new java.awt.event.MouseAdapter() {
public void mouseClicked(java.awt.event.MouseEvent evt) {
jLabelminimizeMouseClicked(evt);
}
});
jLabel2.setFont(new java.awt.Font("Arial", 1, 24)); // NOI18N
jLabel2.setForeground(new java.awt.Color(255, 255, 255));
jLabel2.setText("Login Form");
jLabelclose.setFont(new java.awt.Font("Arial", 1, 24)); // NOI18N
jLabelclose.setForeground(new java.awt.Color(255, 255, 255));
jLabelclose.setText(" x");
jLabelclose.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
jLabelclose.addMouseListener(new java.awt.event.MouseAdapter() {
public void mouseClicked(java.awt.event.MouseEvent evt) {
jLabelcloseMouseClicked(evt);
}
});
javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
.addGap(21, 21, 21)
.addComponent(jLabel2)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
.addComponent(jLabelminimize, javax.swing.GroupLayout.PREFERRED_SIZE, 22,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabelclose, javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addContainerGap())
);
jPanel1Layout.setVerticalGroup(
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel1Layout.createSequentialGroup()
.addContainerGap()
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELIN
E)
.addComponent(jLabel2)
.addComponent(jLabelminimize)
.addComponent(jLabelclose))
.addContainerGap(28, Short.MAX_VALUE))
);
```

```
jPanel2.setBackground(new java.awt.Color(44, 62, 80));
username.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N
username.setForeground(new java.awt.Color(255, 255, 255));
username.setText("USERNAME : ");
password.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N
password.setForeground(new java.awt.Color(255, 255, 255));
password.setText("PASSWORD : ");
ufield.setFont(new java.awt.Font("Arial", 0, 16)); // NOI18N
ufield.setForeground(new java.awt.Color(51, 51, 51));
ufield.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent evt) {
ufieldActionPerformed(evt);
}
});
pfield.setFont(new java.awt.Font("Arial", 0, 16)); // NOI18N
pfield.setForeground(new java.awt.Color(51, 51, 51));
login.setBackground(new java.awt.Color(51, 51, 255));
login.setFont(new java.awt.Font("Arial", 1, 14)); // NOI18N
login.setForeground(new java.awt.Color(255, 255, 255));
login.setText("LOGIN");
login.addMouseListener(new java.awt.event.MouseAdapter() {
public void mouseClicked(java.awt.event.MouseEvent evt) {
loginMouseClicked(evt);
}
});
login.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent evt) {
loginActionPerformed(evt);
}
});
cancel.setBackground(new java.awt.Color(255, 51, 51));
cancel.setForeground(new java.awt.Color(255, 255, 255));
cancel.setText("CANCEL");
newlogin.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
newlogin.setForeground(new java.awt.Color(255, 255, 255));
newlogin.setText(" Create new Login");
newlogin.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
newlogin.addMouseListener(new java.awt.event.MouseAdapter() {
public void mouseClicked(java.awt.event.MouseEvent evt) {
newloginMouseClicked(evt);
}
});
javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel2Layout.createSequentialGroup()
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel2Layout.createSequentialGroup()
.addGap(38, 38, 38)
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
```

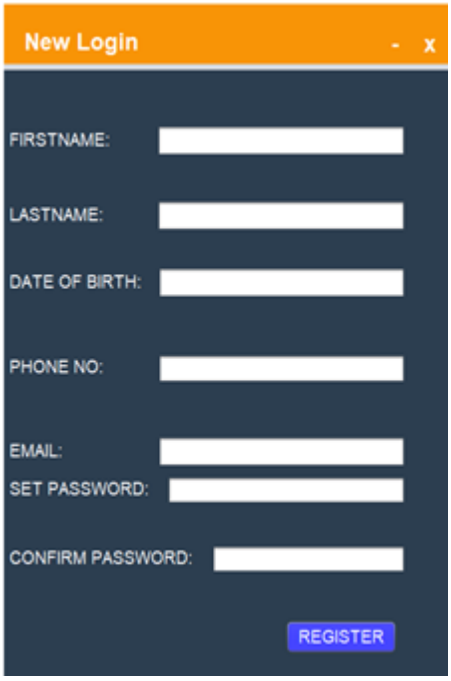
```
.addGroup(jPanel2Layout.createSequentialGroup()
.addComponent(username)
.addGap(18, 18, 18)
.addComponent(ufield, javax.swing.GroupLayout.PREFERRED_SIZE, 202,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addGroup(jPanel2Layout.createSequentialGroup()
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
)
.addComponent(cancel)
.addComponent(password))
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel2Layout.createSequentialGroup()
.addGap(18, 18, 18)
.addComponent(pfield))
.addGroup(jPanel2Layout.createSequentialGroup()
.addGap(75, 75, 75)
.addComponent(login)
.addGap(0, 0, Short.MAX_VALUE))))))
.addGroup(jPanel2Layout.createSequentialGroup()
.addGap(120, 120, 120)
.addComponent(newlogin)))
.addContainerGap(13, Short.MAX_VALUE))
);
jPanel2Layout.setVerticalGroup(
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel2Layout.createSequentialGroup()
.addGap(37, 37, 37)
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELIN
E)
.addComponent(username)
.addComponent(ufield, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(32, 32, 32)
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELIN
E)
.addComponent(password)
.addComponent(pfield, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 57,
Short.MAX_VALUE)
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELIN
E)
.addComponent(cancel, javax.swing.GroupLayout.PREFERRED_SIZE, 39,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(login, javax.swing.GroupLayout.PREFERRED_SIZE, 39,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(65, 65, 65)
.addComponent(newlogin)
.addContainerGap())
);
javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
```

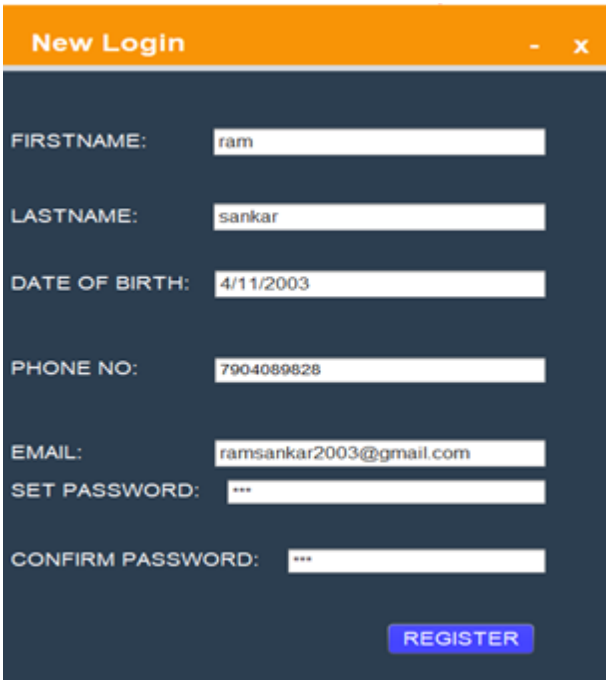
```
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);
layout.setVerticalGroup(
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(layout.createSequentialGroup()
.addGap(67, 67, 67)
.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
);
pack();
}// </editor-fold>
private void ufieldActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
}
private void jLabelcloseMouseClicked(java.awt.event.MouseEvent evt) {
System.exit(0);
}
private void jLabelminimizeMouseClicked(java.awt.event.MouseEvent evt) {
this.setState(JFrame.ICONIFIED);
}
private void loginActionPerformed(java.awt.event.ActionEvent evt) {
}
private void newloginMouseClicked(java.awt.event.MouseEvent evt) {
NewLogin nl=new NewLogin();
nl.setVisible(true);
nl.pack();
nl.setLocationRelativeTo(null);
nl.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.dispose();
}
private void loginMouseClicked(java.awt.event.MouseEvent evt) {
new VehicleSelect(ufield.getText()).setVisible(true);
this.dispose();
}
/**
* @param args the command line arguments
*/
public static void main(String args[]) {
/* Set the Nimbus look and feel */
//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
/* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
*/
```

```
try {
for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
if ("Nimbus".equals(info.getName())) {
javax.swing.UIManager.setLookAndFeel(info.getClassName());
break;
}
}
} catch (ClassNotFoundException ex) {
java.util.logging.Logger.getLogger(Loginform.class.getName()).log(java.util.logging.Level.SEV
ERE, null, ex);
} catch (InstantiationException ex) {
java.util.logging.Logger.getLogger(Loginform.class.getName()).log(java.util.logging.Level.SEV
ERE, null, ex);
} catch (IllegalAccessException ex) {
java.util.logging.Logger.getLogger(Loginform.class.getName()).log(java.util.logging.Level.SEV
ERE, null, ex);
} catch (javax.swing.UnsupportedLookAndFeelException ex) {
java.util.logging.Logger.getLogger(Loginform.class.getName()).log(java.util.logging.Level.SEV
ERE, null, ex);
}
//</editor-fold>
/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
public void run() {
new Loginform().setVisible(true);
}
});
}
// Variables declaration - do not modify
private javax.swing.JButton cancel;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabelclose;
private javax.swing.JLabel jLabelminimize;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JButton login;
private javax.swing.JLabel newlogin;
private javax.swing.JLabel password;
private javax.swing.JPasswordField pfield;
private javax.swing.JTextField ufield;
private javax.swing.JLabel username;
// End of variables declaration
}
```


Login Form
- x
USERNAME :
PASSWORD :
CANCEL
LOGIN
Create new Login


Login Form
- x
USERNAME :
ram sankar
PASSWORD :
***
CANCEL
LOGIN
Create new Login

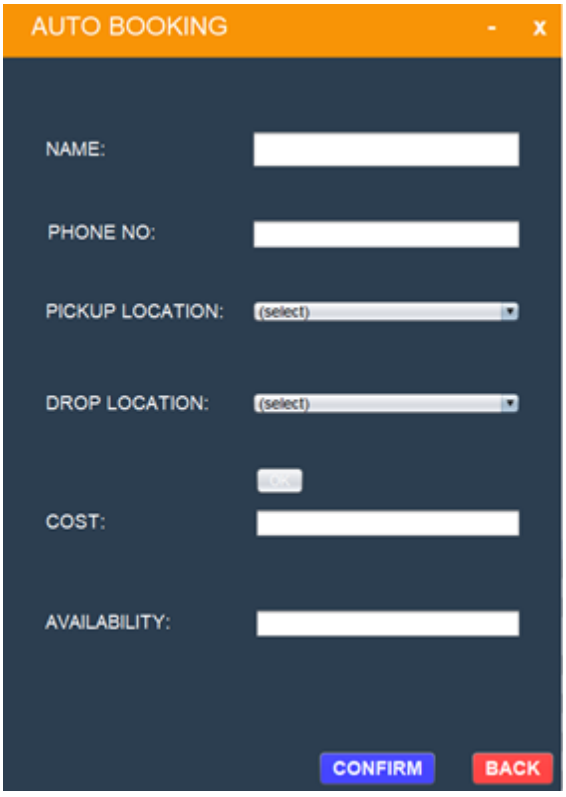

New Login
- x
FIRSTNAME:
LASTNAME:
DATE OF BIRTH:
PHONE NO:
EMAIL:
SET PASSWORD:
CONFIRM PASSWORD:
REGISTER

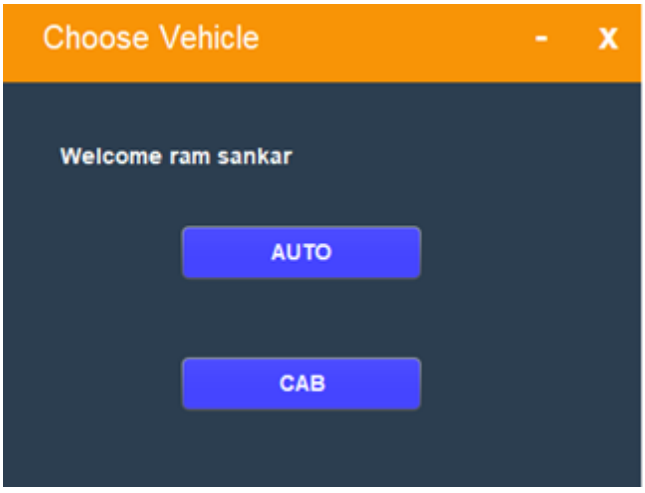

New Login
- x
FIRSTNAME:
ram
LASTNAME:
sankar
DATE OF BIRTH:
4/11/2003
PHONE NO:
7904089828
EMAIL:
ramsankar2003@gmail.com
SET PASSWORD:
***
CONFIRM PASSWORD:
***
REGISTER

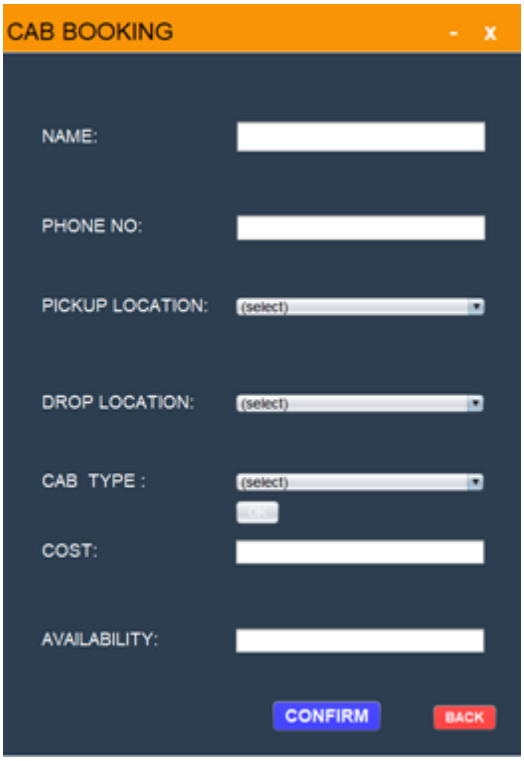

Login Form
- x
Message
Registration success.!!
OK
CANCEL
LOGIN
Create new Login

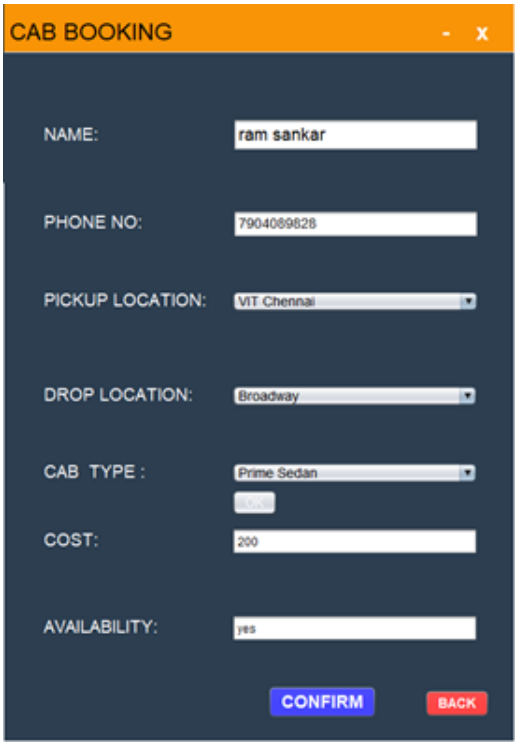

New Login
FIRSTNAME:
ram
LASTNAME:
DATE OF BIRTH:
4/11/2003
PHONE NO:
7904089828
EMAIL:
ramsankar2003@gmail.com
SET PASSWORD:
CONFIRM PASSWORD:
REGISTER

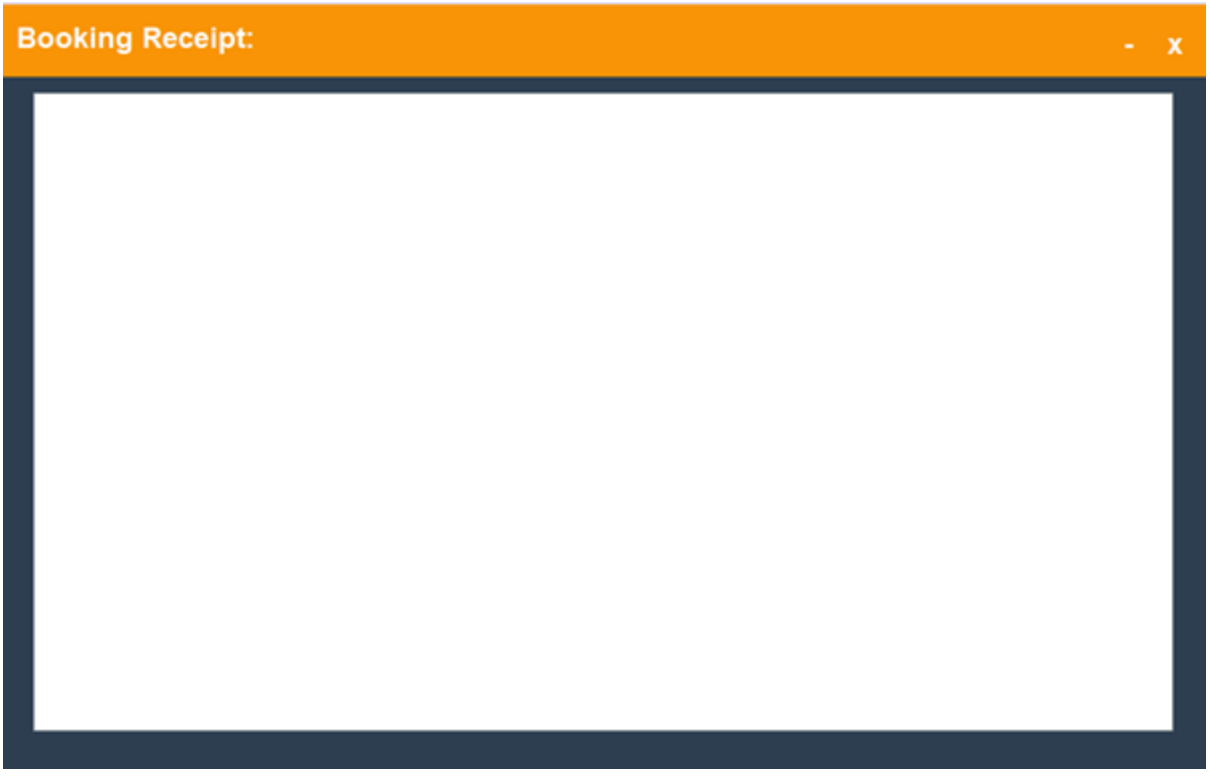

Login Form
Message
One of the field is missing!!!
OK
CANCEL
LOGIN
Create new Login

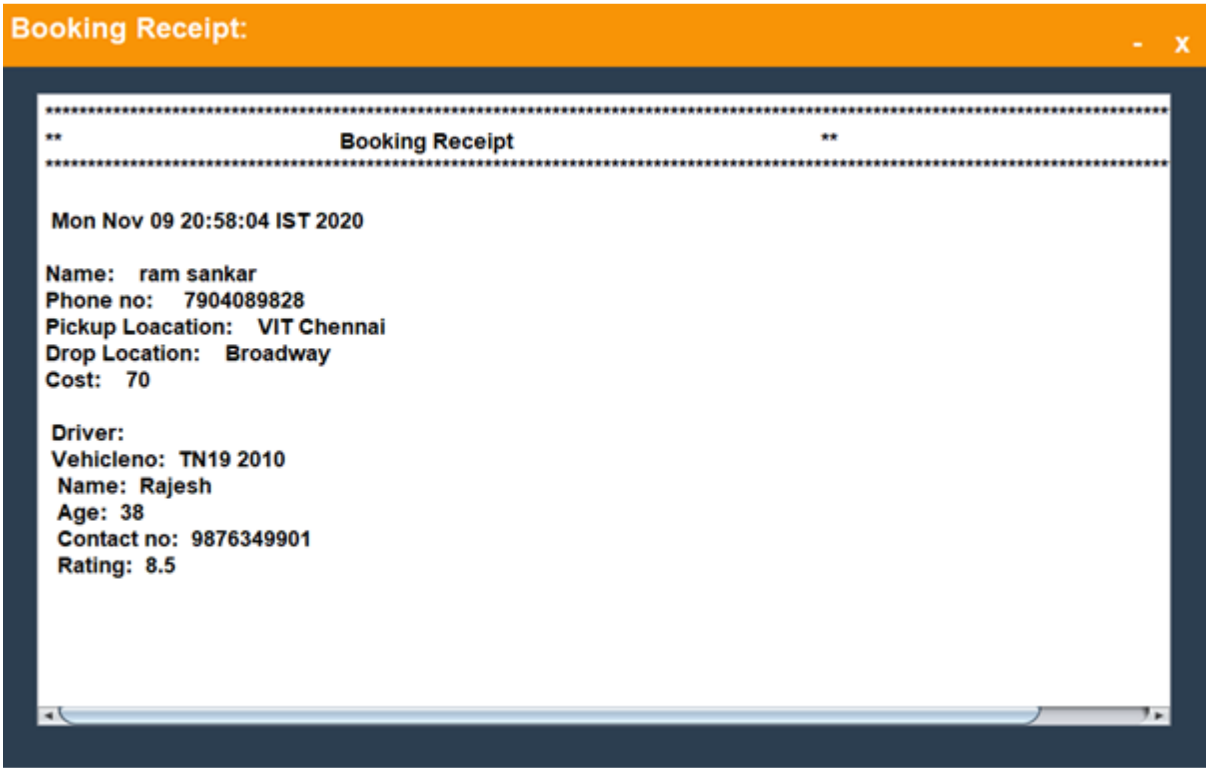

AUTO BOOKING
NAME:
PHONE NO:
PICKUP LOCATION:
(select)
DROP LOCATION:
(select)
COST:
AVAILABILITY:
CONFIRM
BACK


Choose Vehicle
Welcome ram sankar
AUTO
CAB


CAB BOOKING
- x
NAME:
PHONE NO:
PICKUP LOCATION:
(select)
DROP LOCATION:
(select)
CAB TYPE :
(select)
COST:
AVAILABILITY:
CONFIRM
BACK


CAB BOOKING
- x
NAME:
ram sankar
PHONE NO:
7904069828
PICKUP LOCATION:
VIT Chennai
DROP LOCATION:
Broadway
CAB TYPE :
Prime Sedan
COST:
200
AVAILABILITY:
yes
CONFIRM
BACK


Booking Receipt:
- x



Conclusion :

In traditional system cab booking was done manually. Updating driver and customer information is tedious work. In proposed system booking is possible using android application and user friendly in nature. In this system customer book for a ride and depending on the drop location and type of vehicle selected the costs are produced. . Google map also provided for display the distance between source and destination. Main point is the vehicle is always available for the transportation of students , faculties and other workers from college to their nearby places.

References :

- Peng Zhou1, Tamer Nadeem2, Porlin Kang1 and Cristian Borcea, “EZCab: A Cab Booking Application Using Short-Range Wireless Communication”.
- Santosh Thakkar, Namrata Gawade and Prof. Sonia Mehta,”Proposed Advance Taxi Recommender System Based On a Spatiotemporal Factor Analysis Model”.
- Anurag Mandle, Akshay Jaiswal, and Roshan Lokhande, “Taxi Automation Using Real Time Adaptive Scheduling”.
- Bayu Waspodo, Qurrotul Aini and Syamsuri Nur, “Development of Car Rental Management Information System”. Information System, Faculty of Science and Technology Syarif Hidayatullah State Islamic University Jakarta, Indonesia.
- Gauri kumbhojkar, simantini kokate, rajeshwari patil and harshada sonawane,”Web based taxi management system using RFID.”
- Vidya L. Tikone, Snehal M. Tembare,Manisha L. Narad and Bharat V. Supekar, ”Quick Retrieval of Nearest Neighbor by using Keywords”.