The Nemesis Travel Agency Portal

Submitted by

Anand Lad (Exam No 604022)
Sneh Shah (Exam No. 604051)
Sanket Patel (Exam No.604040)

Guided By

Mrs. Anjali Jivani
Ms. Mamta Padole

Submitted to

Department of Computer Science
Faculty of Technology & Engineering
The Maharaja Sayajirao University of Baroda
Year 2015

THE MAHARAJA SAYAJIRAO UNIVERSITY OF BARODA

सत्यं शिवं सुन्दरम्

Faculty of Technology & Engineering
The Maharaja Sayajirao University

CERTIFICATE

This is to certify that **Anand Lad (Roll No 522), Sneh Shah (Roll No 551) & Sanket Patel (Roll No 566)** have successfully completed their project work titled “Nemesis – The Travel Management System” under our supervision and guidance to our utmost satisfaction. This project was carried out during second semester of B.E.III Computer Science & Engineering.

Date: 20/05/2015

Head
Computer Science &Engg.

Acknowledgement :

We sincerely feel the credit of the project work could not be narrowed down to only on individual. The development of this project involves many valuable contributions.

In this venture we are constantly guided and encouraged by **Ms. Mamta Padole**. We gratefully acknowledge their encouragement without which our task would have been futile.

We would also thank our Head of the Department **Mrs. Anjali Jivani** for giving us more ideas about our project and guided us during the whole project.

Overview :

The project is a clear Agent side software which shall be used for Ticket Booking , Itinerary considerations, Cancellations, Financing and Service Comparisions Routewise.

The Main aim of the project is to bring up all the requirements of a travel agent and the agent's overheld travel agencies at the maximum ease replacing all the manual efforts of the agent starting from basic travel booking to Managing the finance, Date to Date.

Facilities Upheld :

- Travel Booking (Availibility Check, User Seat requirement Consideration, Amenities check).

- Booking Cancellation (Check the Current Status, Date of Booking and the Cancellation charges accordingly,
  Updating the seat availability status).

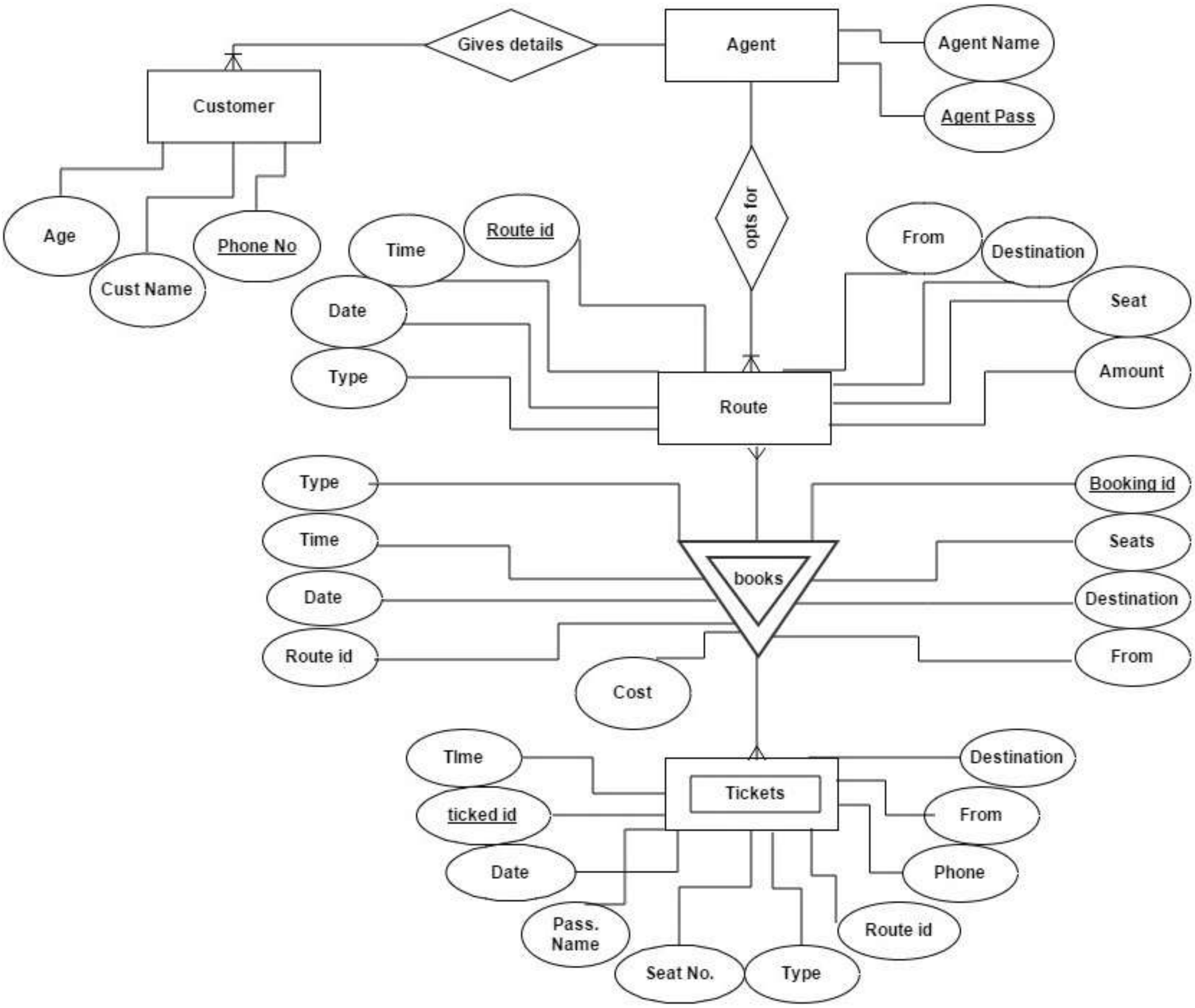
- Prepare and Print a Ticket with all the prompt and Precise information.
- Compare services selected and Tickets sold for a specific period via Line Chart and Bar Chart.

The Assumptions made for the Project Development are:

- The Software serves one agent for an instance of time (No multiusers).
- The Services Provided by the agent are for limited Routes across the country.
- The Booking facilities prevail for a limited number of days pertaining to the start.

A Brief Description of the Nemesis Travel Management Portal is explained hereby !

DATABASE END :



ER Diagram

Tables and Triggers on Tables Implemented in Database :

As witnessed from the E R Diagram above, the Database end has the Following tables :

- **Booking.**
- **Ticket.**
- **Route:**
- **Customer.**
- **Agent.**

The triggers have been implemented to serve the following purpose:

1. To automatically update the cost incurred by the customer along the cost attribute with respect to the number of seats in the Booking table from route table (Trigger ct).

2. To automatically fetch the Travel locations in the form of Route id via Route table into the booking table alongwith the time of Journey. (Trigger cst).

3. To automatically update (Subtract) the No. of Available seats
after the selection of No. of seats to be booked by the customer and thus update the Route table too.
(Trigger ct1).

4. To automatically update (Add) the No. of Available seats after the Cancellation of seats by the customer and thus update the Route Table too.

(Trigger ct2).

5. To automatically select the Travel locations in the form of Route id via Route Table into Ticket along with the time of Journey (Trigger st).

6. To automatically make the seat available after deletion of a Ticket and thus update the Route table. (Trigger st2).

The implementation of the respective triggers is as shown:

Trigger ct :

```
create or replace trigger ct
```

```
before insert on booking for each row
begin
select (r.amount* (:new.seats)) into :new.cost
FROM route r
where :new.frm=r.fromp and :new.destination=r.dest and :new.type=r.type and
:new.dat=dats;
end ct;
```

Trigger cst:

```
create or replace trigger cst

before insert on booking for each row
declare
rdi number(10);
begin
select r.rid,r.time into :new.rid,:new.time
FROM route r
where :new.frm=r.fromp and :new.destination=r.dest and :new.type=r.type and
:new.dat=dats;
end cst;
```

Trigger ct1:

```
create or replace trigger ct1
before insert on booking for each row

declare
st number(10);
begin
select (r.seat-(:new.seats)) into st
FROM route r
where :new.frm=r.fromp and :new.destination=r.dest and :new.type=r.type and
:new.dat=dats;

update route
set seat=st
```

```
where :new.frm=fromp and :new.destination=dest and :new.type=type and
:new.dat=dats;
end ct1;
```

Trigger ct2:

```
create or replace trigger ct2
before delete on booking for each row

declare
st number(10);
begin
select (r.seat + (:old.seats)) into st
FROM route r
where :old.frm=r.fromp and :old.destination=r.dest and :old.type=r.type;
update route
set seat=st
where :old.frm=fromp and :old.destination=dest and :old.type=type;

end ct2;
```

Trigger st:

```
create or replace trigger st
before insert on ticket for each row
declare
rdi number(10);

begin
select r.rid,r.time into :new.rid,:new.time
FROM route r
where :new.frm=r.fromp and :new.dest=r.dest and :new.type=r.type and
:new.dates=dats;


end st;
```

Trigger st2:

```
create or replace trigger st2
before delete on ticket for each row

declare
st number(10);
begin

update route r
set seat=seat+1
where :old.rid=r.rid ;
end st2;
```

The database end also serves a function as follows :

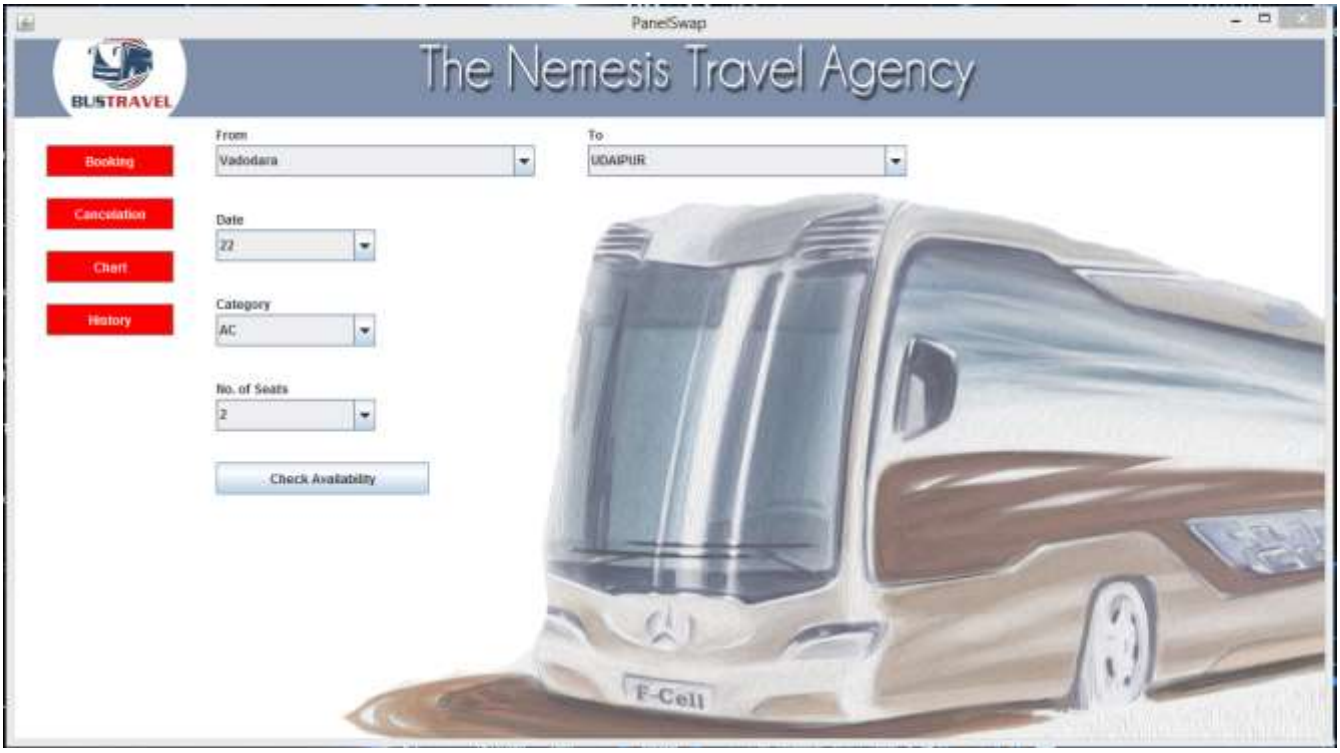
```
create or replace function del (t ticket.tid%type)
return  number is

t1 number(10);
begin
delete from ticket where tid=t;
return t;
end;
```

This function serves the purpose of deleting the details of a ticket from the Ticket table via the ticket number and thus the cancellation Operation is backed by a database function which looks after the ticked id passind for the deletion.

Graphical User Interface using Java :

Screen Shots displaying the Implementations of Java Framework of AWT, Swing & Charts using Multithreading:



From: Vadodara

To: UDAIPUR

Date: 22

Category: AC

No. of Seats: 2

This is the Homescreen for the Agent after the login.

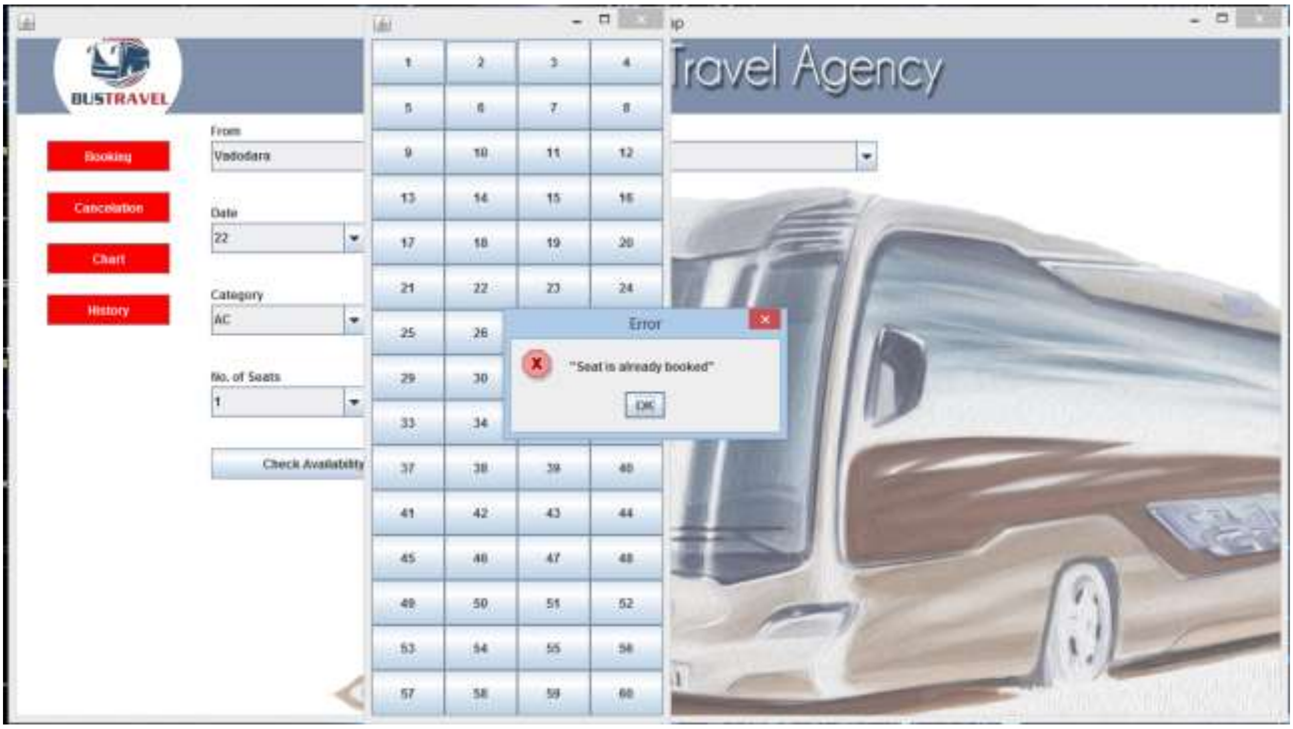
The Login Screen is the basic screen for the Agent to

Work ahead.
JPanels and JFrames are the carriers of our Information for the agent.
Project Nemesis is an extensive application of all the features of AWT Event Handelling and Swing.

The Homescreen itself is an amalgamation of **JCombobox,**
Jframe,JPanel,JButtons,JTextField,JLabeletc.

All the Operations are controlled from the Homescreen.

The Booking Screen :



From: Vadodara

Date: 22

Category: AC

No. of Seats: 1

Error

"Seat is already booked"

Entering the Necessary details pertaining to the Journey, we have the Implementation of GridLayout for the selection of seats for travel.

Necessary Constraints for the Entry of details and the Route selection backet by Database Backend forms the base for the Error Dialogue Box incase the Seat is already booked for some other Passenger.

JComboBox enables selection from the list and Event Handelling results into consideration of Users actions with the system.

JOptionPane enables the popping up of the Error Showing Dialogue box which appears incase the seat is already

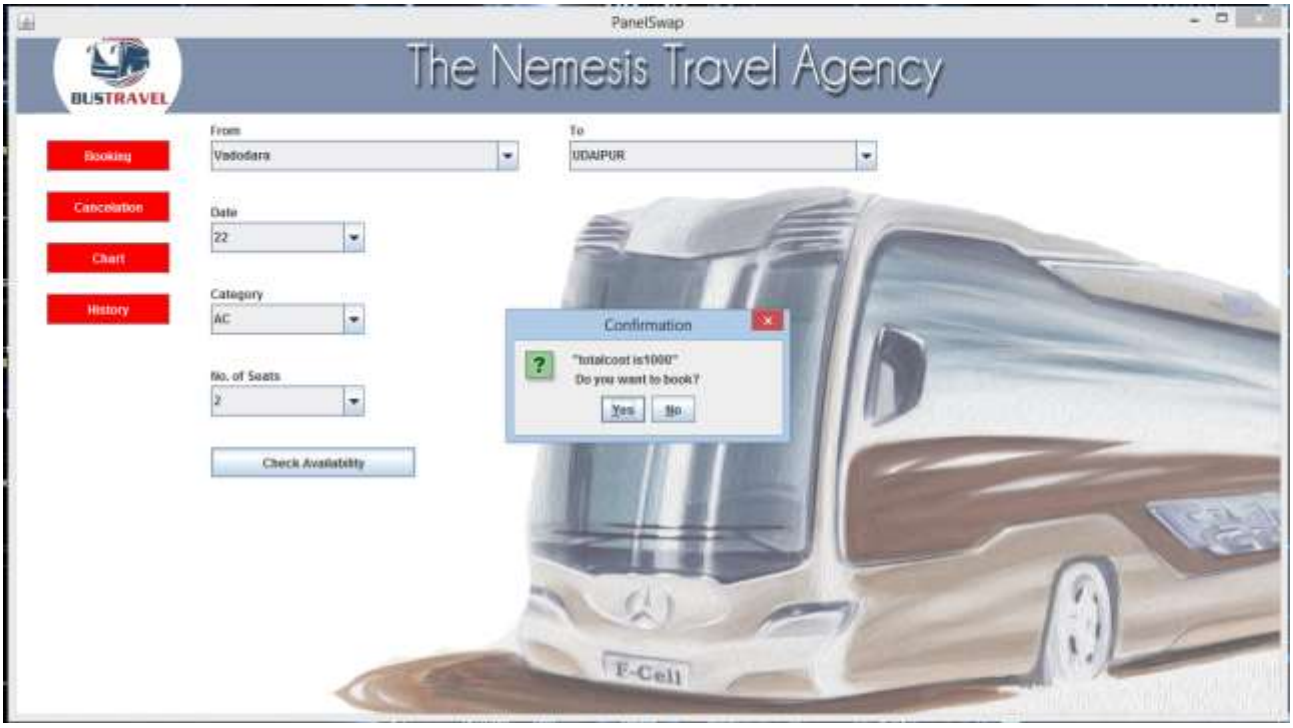
booked.

The Confirmation Screen:

Once the number of seats are selected by the agent, the Confirmation Dialogue box appears asking if he wants to Continue further.

The Cost Updation is a result of the Database Trigger mentioned above and as a result of this the Final cost after booking is displayed to the agent asking if he wants to continue ahead.

All this is still backed by a constant background system Frame carrying all the basic operations.



PanelSwap

The Nemesis Travel Agency

From: Vadodara

To: UDAIPUR

Date: 22

Category: AC

No. of Seats: 2

Confirmation

"totalcost is1000"
Do you want to book?

The Seat Booking Screen :

Once the seat is Available, A frame asking the Passenger details is the next turn up for the System.

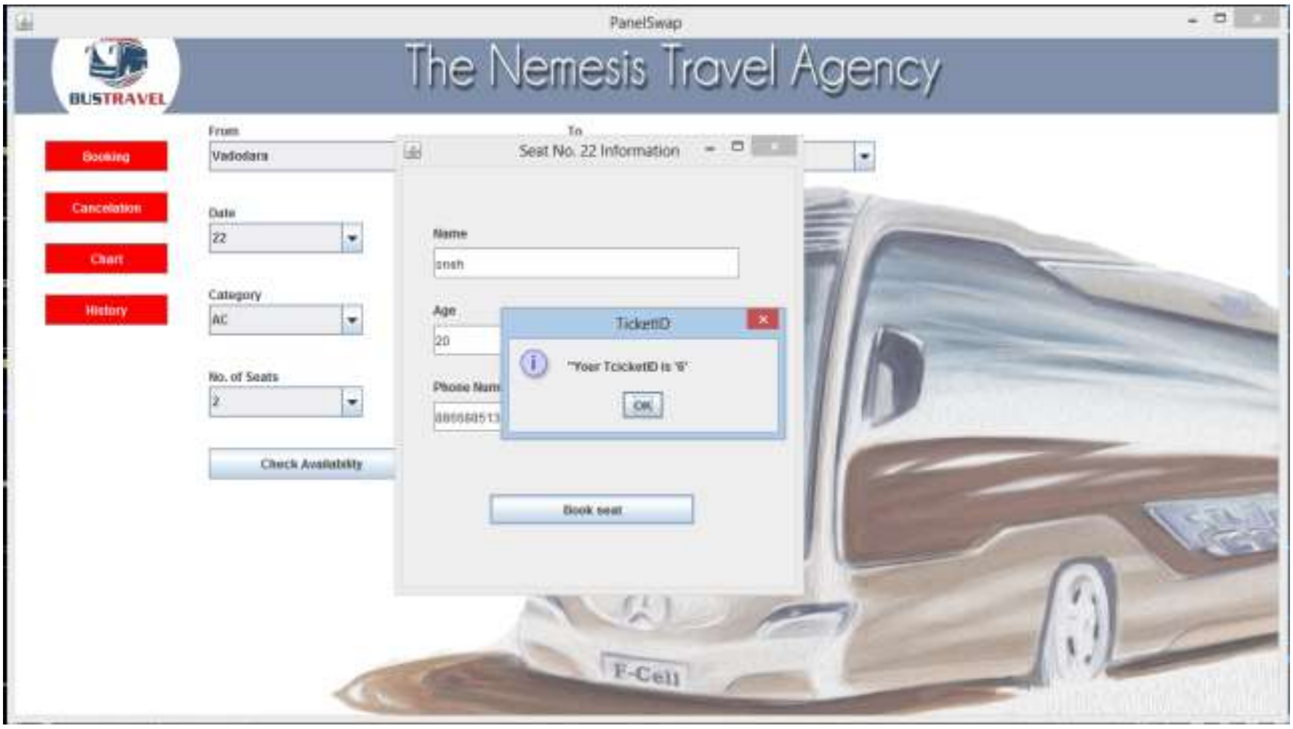
This too is accompanied by constraints checking the entry of Valid Data only. Eg – The Phone Number must have 10 digits.

Incase of any Invalid entry, Appropriate error box appears.

If all the details entered by the agent are correct, the

ticket is booked and the Ticket id is generated accordingly which uniquely identifies the ticket.

All this too is handelled strongly by Backend database triggers modifying respective tables of the database.



From: Vadodara

To:

Date: 22

Category: AC

No. of Seats: 2

Name: snsh

Age: 20

Phone Num: 886668513

"Your TcicketID is '6'

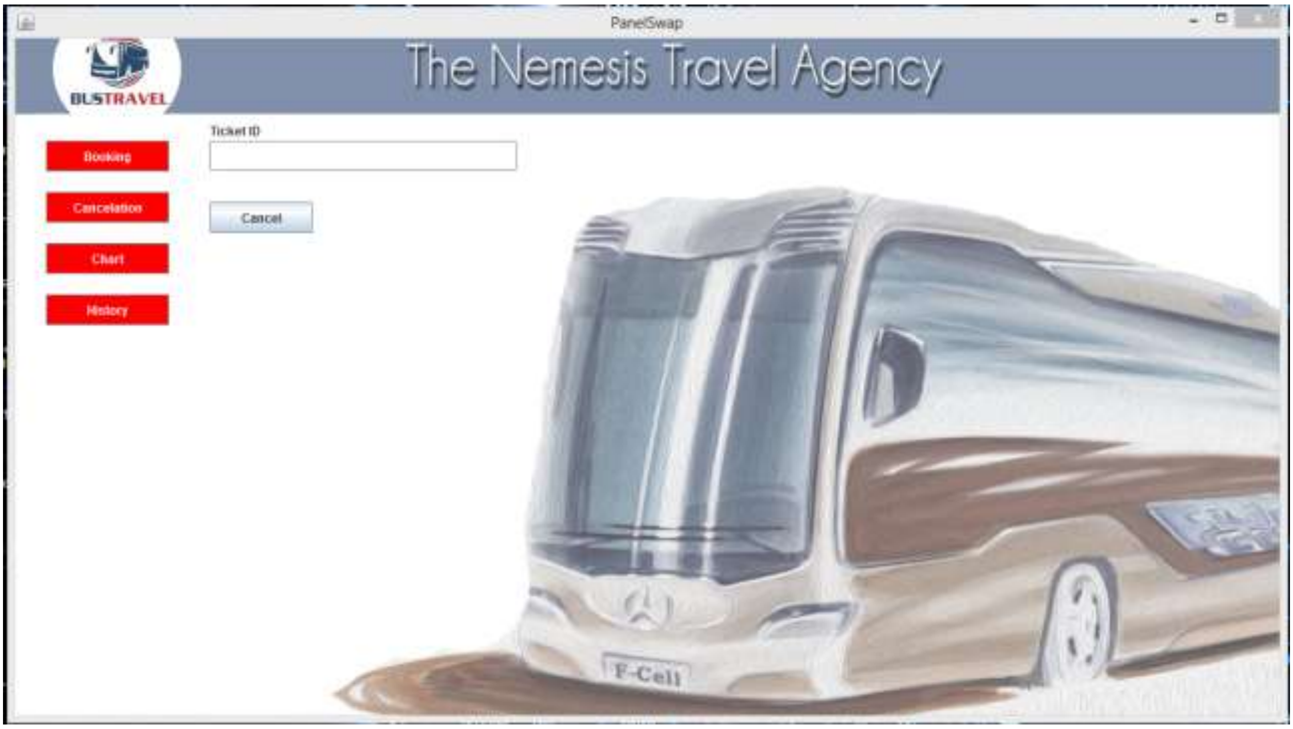
The Cancellation Screen:

As a part of the standard operations, Cancellation is

one of the Prime holders of the system.

Incase the Customer wants to cancel the tickets, the agent thus enters his ticket id in the JTextField which leads to the omittion of the concerned entry.

Database is thoroughly updated with each cancellaton with the help of Triggers.

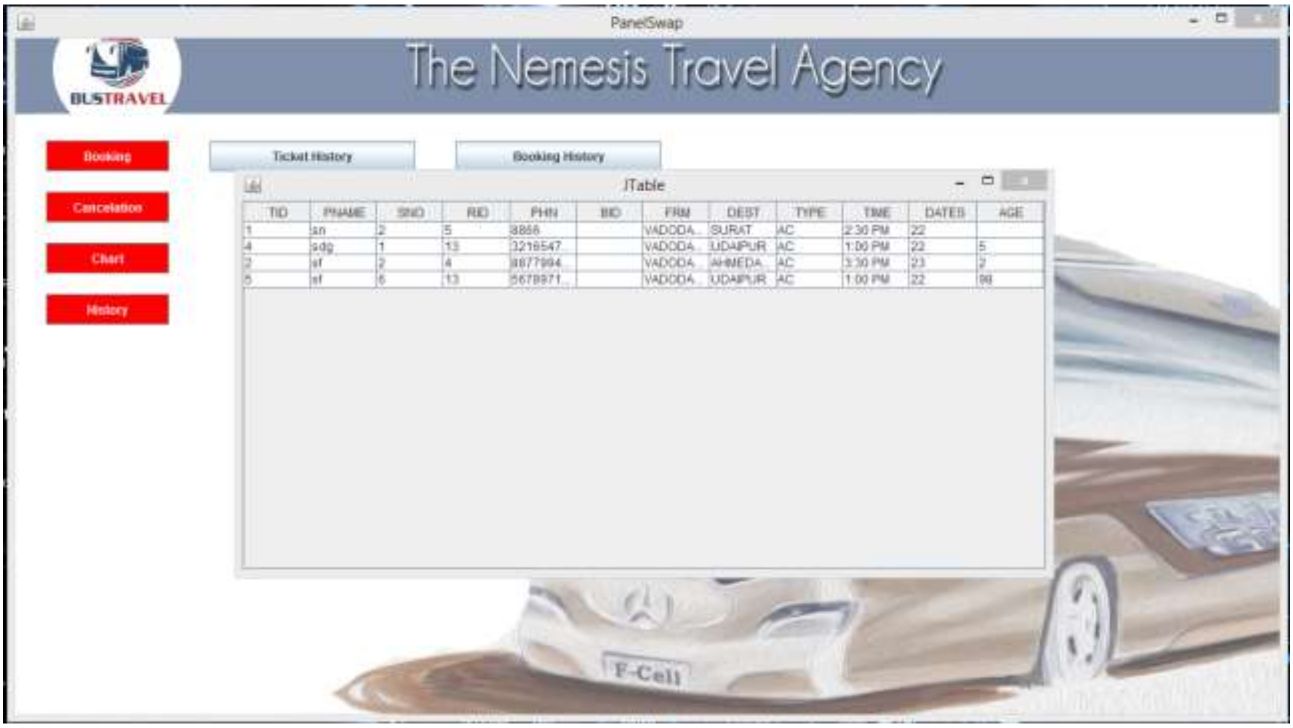


Ticket History and Booking History :

Checking the Ticket and Booking history is the part of an Agent's agenda list.

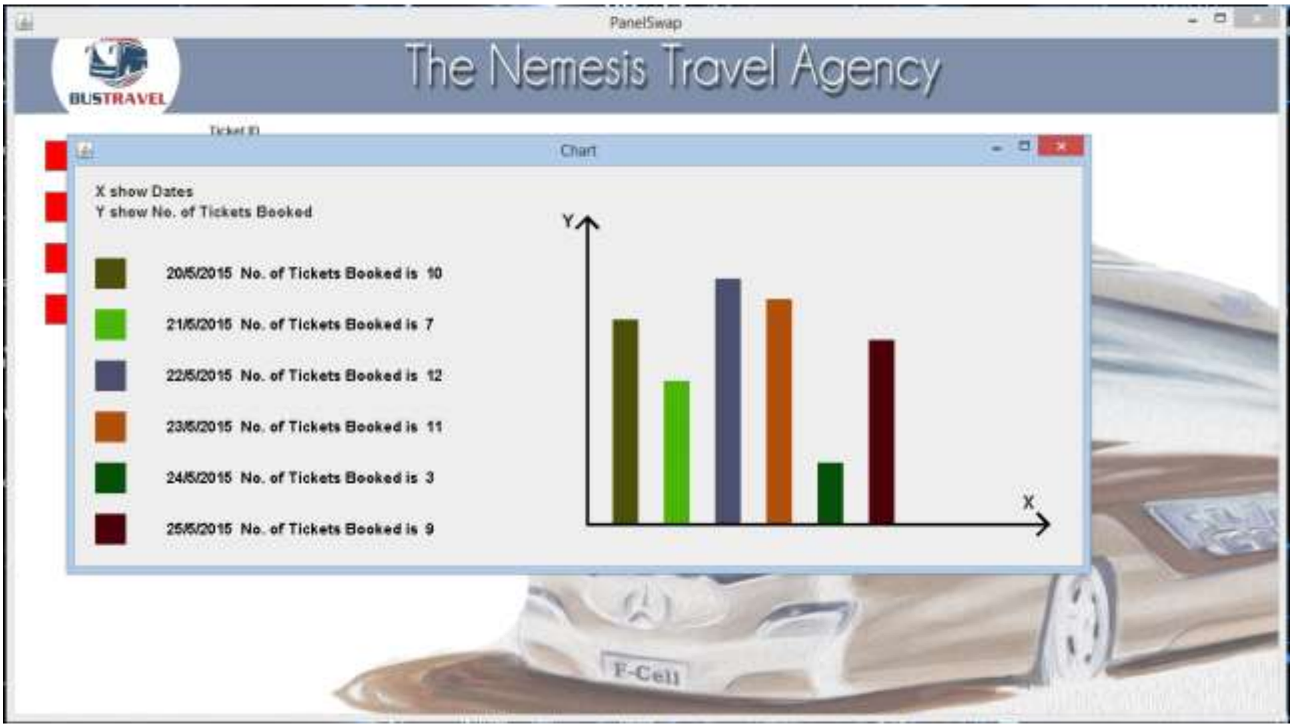
This is briefly elaborated by Jtable showing each of the Ticket History and Booking history under the Major Operations Frame.

This is also Backed by Database updation with each entry and Cancellation of Tickets.



TID	PNAME	SNO	RID	PHN	BID	FRM	DEST	TYPE	TIME	DATES	AGE
1	sn	2	5	8868		VADODA...	SURAT	AC	2:30 PM	22	
4	sdg	1	13	3216547...		VADODA...	UDAIPUR	AC	1:00 PM	22	5
2	sf	2	4	8877994...		VADODA...	AHMEDA...	AC	3:30 PM	23	2
5	sf	6	13	5678971...		VADODA...	UDAIPUR	AC	1:00 PM	22	98

The Transaction and Finance History Chart-Bar chart and Line Chart:



X show Dates
Y show No. of Tickets Booked

🟫 20/5/2015 No. of Tickets Booked is 10

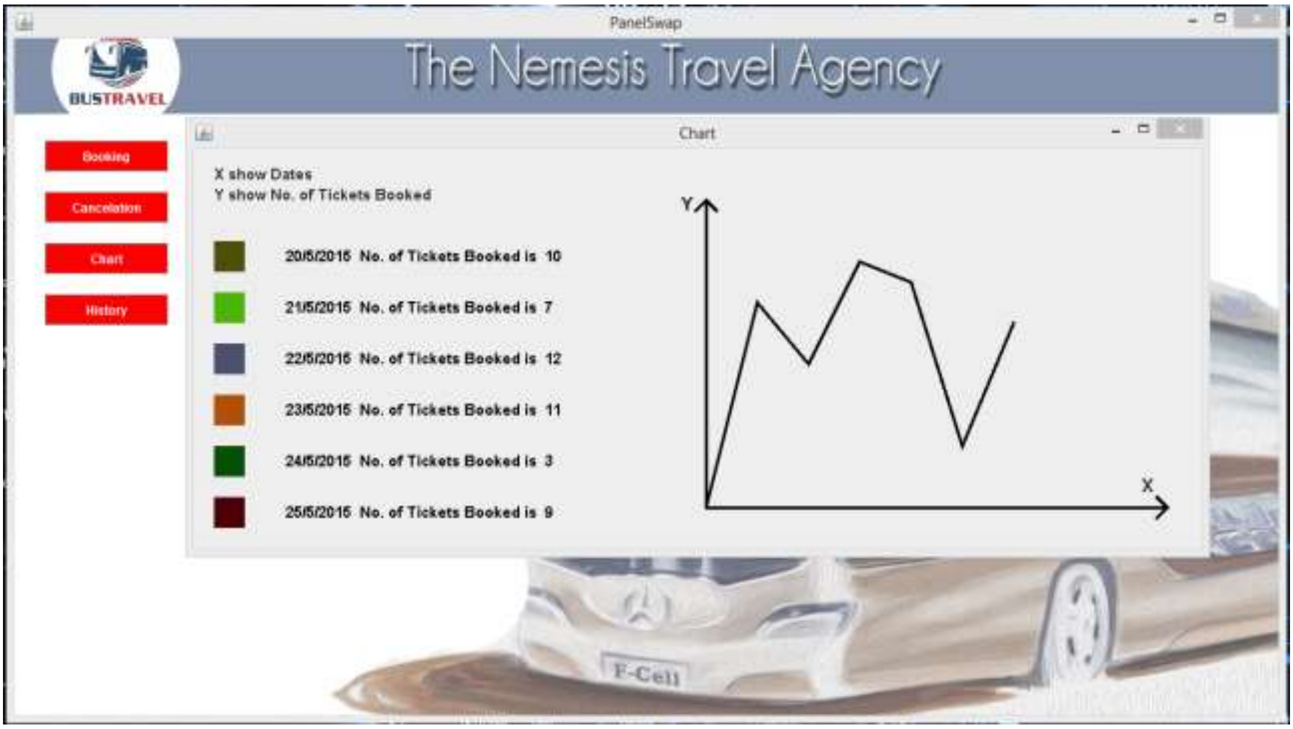
🟩 21/5/2015 No. of Tickets Booked is 7

🟦 22/5/2015 No. of Tickets Booked is 12

🟧 23/5/2015 No. of Tickets Booked is 11

🟩 24/5/2015 No. of Tickets Booked is 3

🟥 25/5/2015 No. of Tickets Booked is 9



X show Dates
Y show No. of Tickets Booked

🟫 20/5/2015 No. of Tickets Booked is 10

🟩 21/5/2015 No. of Tickets Booked is 7

🟦 22/5/2015 No. of Tickets Booked is 12

🟧 23/5/2015 No. of Tickets Booked is 11

🟩 24/5/2015 No. of Tickets Booked is 3

🟥 25/5/2015 No. of Tickets Booked is 9

Brief Explanation:

This is the feature that allows the agent to look after the Statistics in a graphical manner.

The two charts multithreaded together are displayed together showing the statistics for the No of Tickets booked and The Financial turnover over the specific time span.

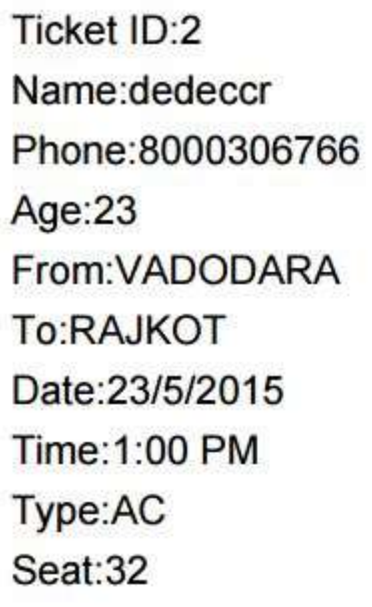
Multithreading accompanied by Dynamic variables for bounds of Lines , the charts are ployed with a multiplacation factor for a significant showcase of difference in statistics for the span of time.

The Dynamic Variable that considers the chart forms the base of the database that leads to the graphical display.

The Agent can view the updated statistics with each transaction in the system.

This is a result of the Multithread Implementation and Variable set Bounds.

Generate Ticket as PDF for Print :



Ticket ID:2

Name:dedeccr

Phone:8000306766

Age:23

From:VADODARA

To:RAJKOT

Date:23/5/2015

Time:1:00 PM

Type:AC

Seat:32

Using PDFBox external package, we have also given an additional facility to the agent to generate a ticket for a given Ticket id the details of which are fetched via SQL queries in the Database Backend.

The agent can thus print the desired ticket anytime.