COMP1610- PROGRAMMING ENTERPRISE COMPONENTS

Tutor : **Dr Markus Wolf**

Submitted

by

Girija Selvakumar

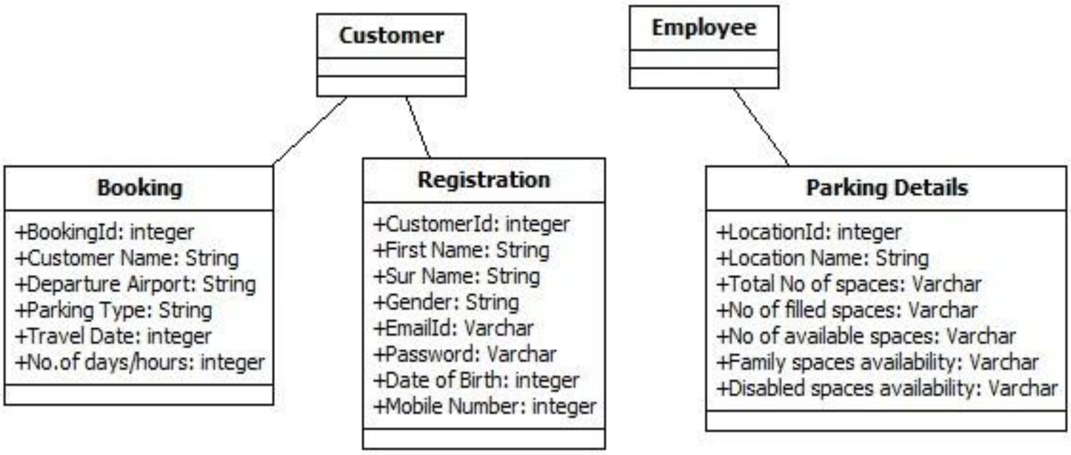
on

26.03.2015

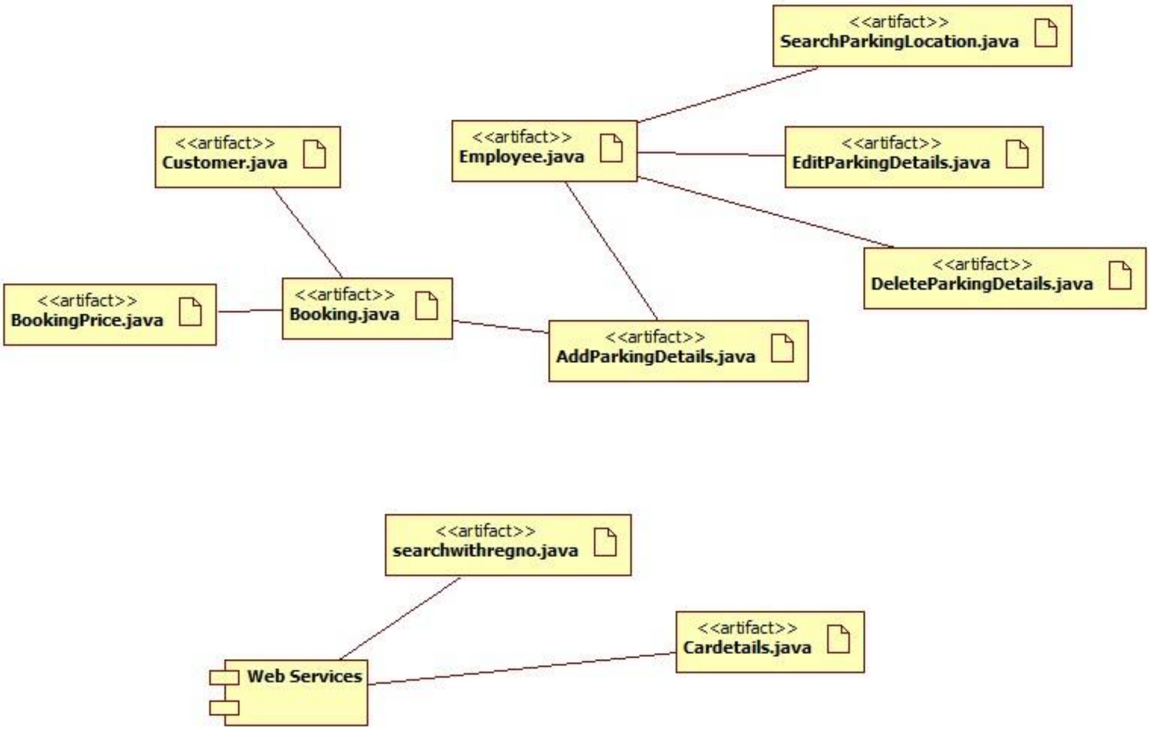
Table of Contents

1. UML AND ERD DIAGRAMS

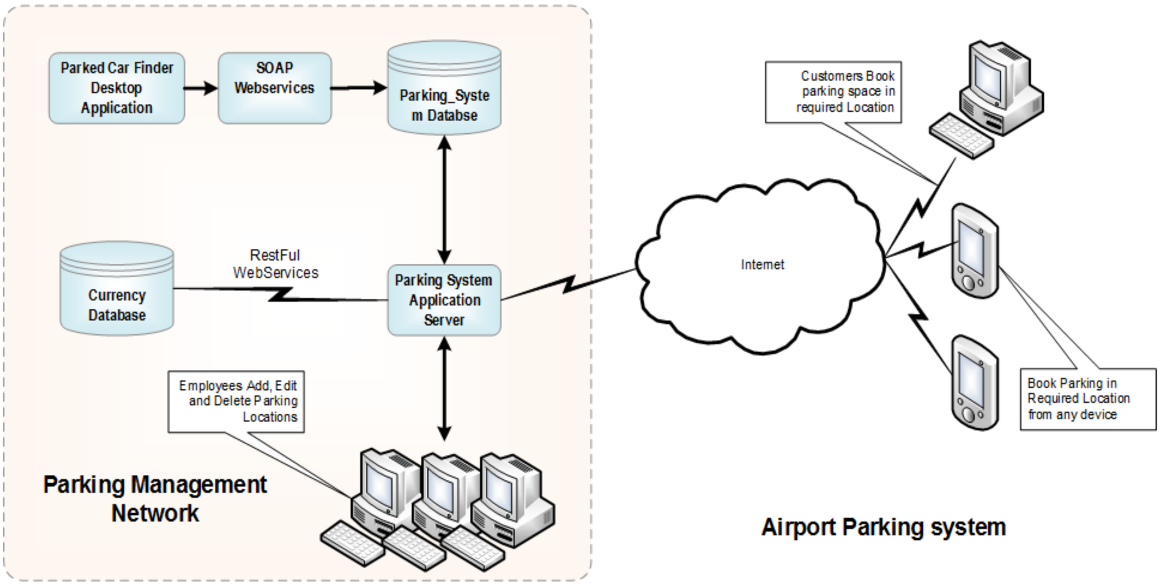
1.1 Class Diagram



1.2 Component Diagram



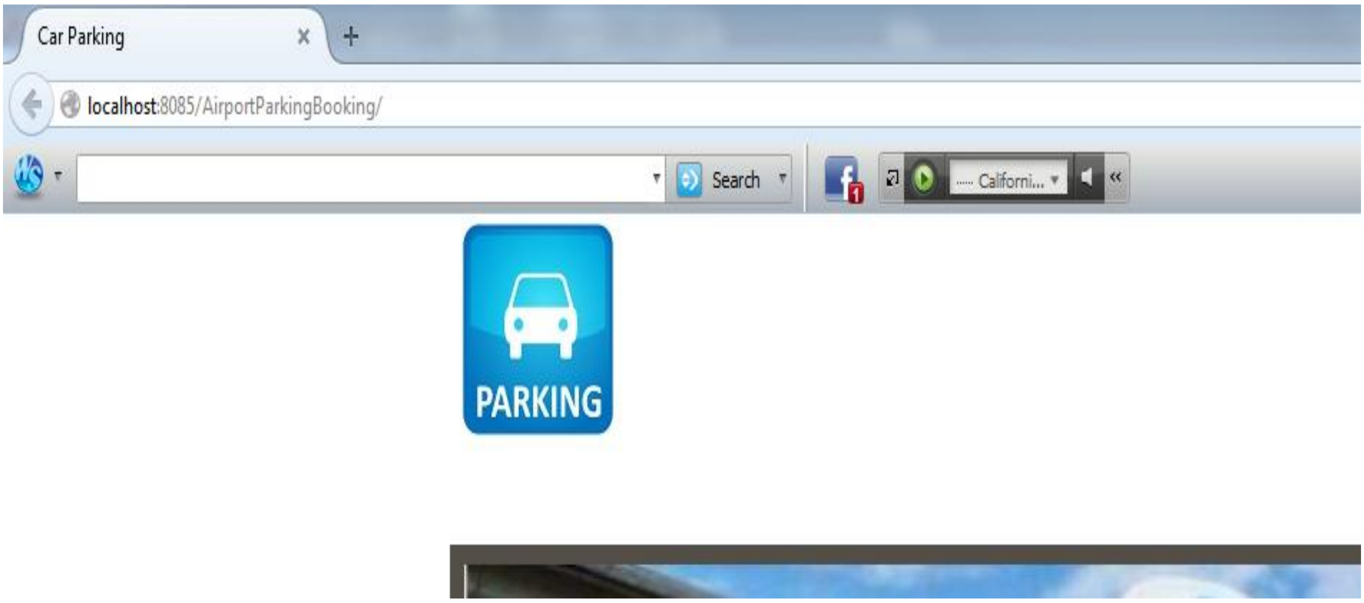
1.3 Architecture Diagram



Parked Car Finder Desktop Application
SOAP Webservices
Parking_System Databse
Currency Database
RestFul WebServices
Parking System Application Server
Employees Add, Edit and Delete Parking Locations
Parking Management Network
Internet
Customers Book parking space in required Location
Book Parking in Required Location from any device
Airport Parking system

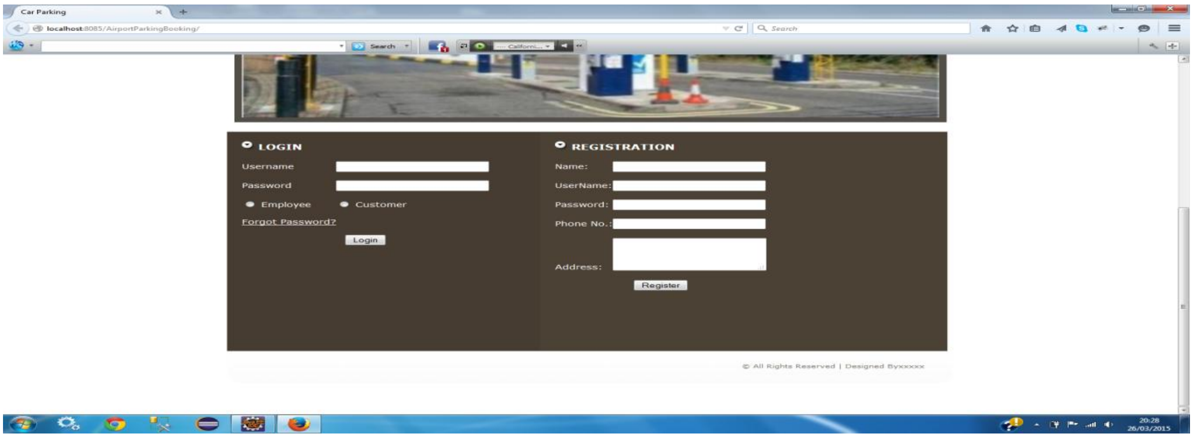
2. APPLICATION SCREEN SHOTS

Index Page



Login and Registration

Employees are provided with the username and password to use the application. But the customer has to register with the applicatiion to make booking.



LOGIN

Username

Password

○ Employee ○ Customer

Forgot Password?

Login

REGISTRATION

Name:

UserName:

Password:

Phone No.:

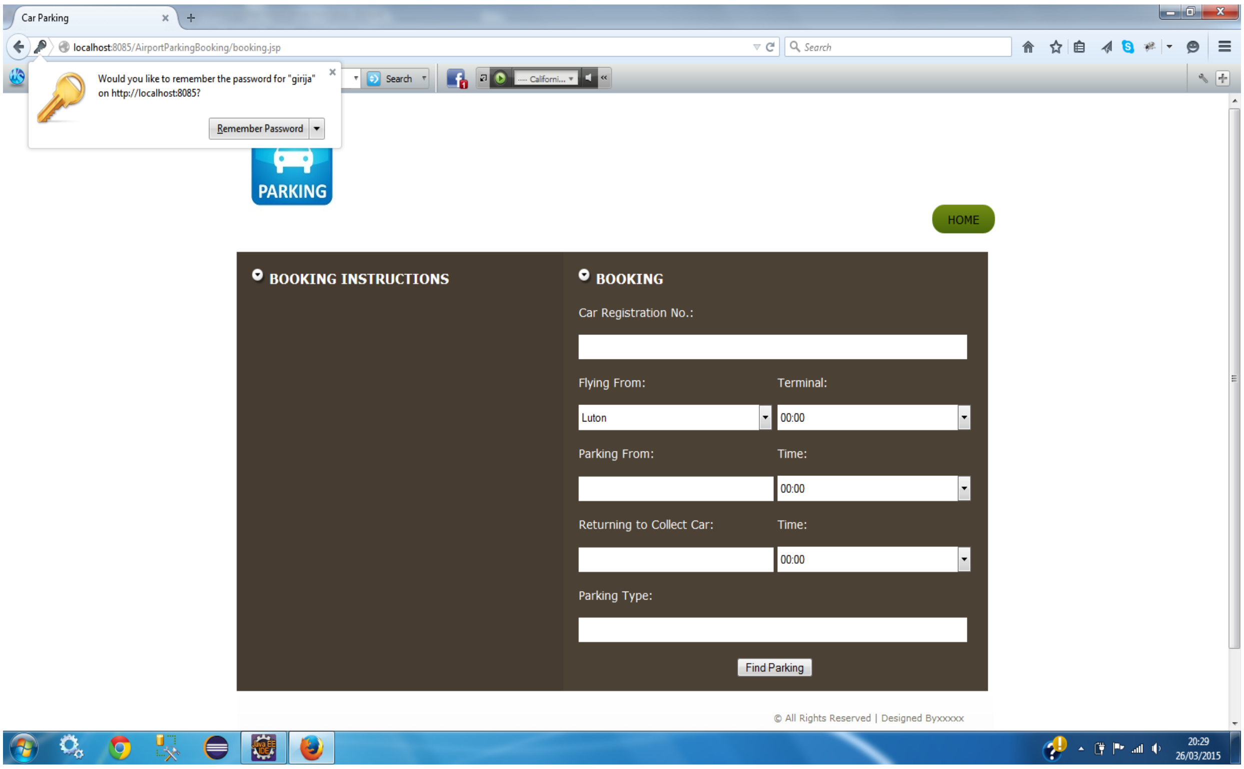
Address:

Register

© All Rights Reserved | Designed Byxxxxx

Booking

After Customer sign in, they will be directed to the booking page. Customers are asked to provide their registration number. Location they are flying from (Parking Location), Parking from date and until when (Parking to date) and also the kind of space they require.



Would you like to remember the password for "girija" on http://localhost:8085?

Remember Password

BOOKING INSTRUCTIONS

BOOKING

Car Registration No.:

Flying From: Luton

Terminal: 00:00

Parking From:

Time: 00:00

Returning to Collect Car:

Time: 00:00

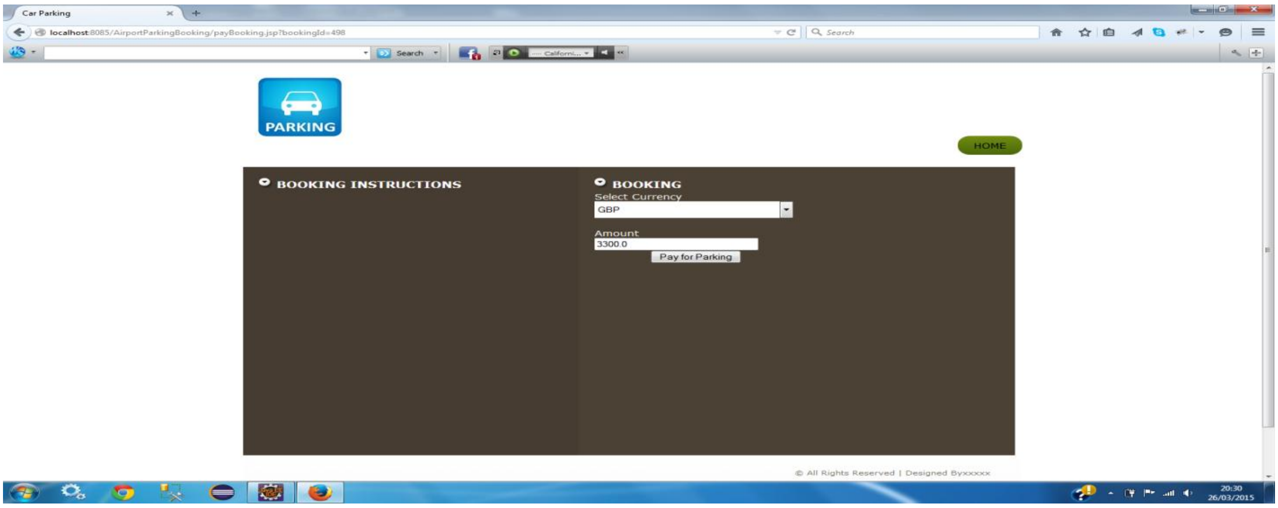
Parking Type:

Find Parking

© All Rights Reserved | Designed Byxxxxxx

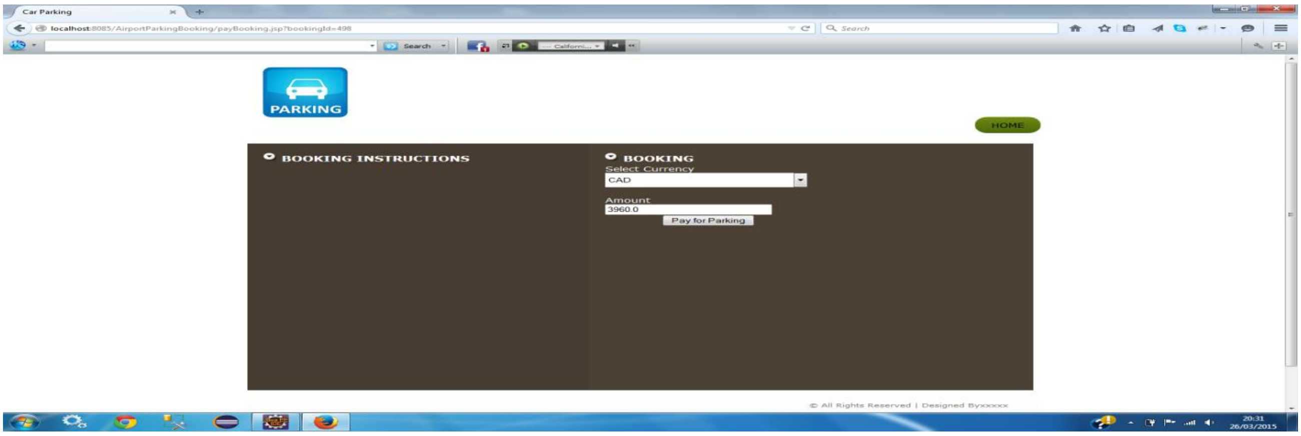
Once they’ve done their booking, customer is directed to payment page., where they can choose the kind of currency they wish to use. Here the booking rate is calculated on per-day basis which is in default 20 GBP.

Rest of the currency values are stored in currency database and are pulled out using RESTFul Web services. This is implemented as a separate restful web application. Following screenshots shows that as soon as the customer selects the currency type from the dropdown, corresponding amount on the basis of no-of days booked is shown.


Car Parking
localhost:8085/AirportParkingBooking/payBooking.jsp?bookingId=498
Search
PARKING
HOME
BOOKING INSTRUCTIONS
BOOKING
Pay for Parking
© All Rights Reserved | Designed Byxxxxx
20:30
26/03/2015

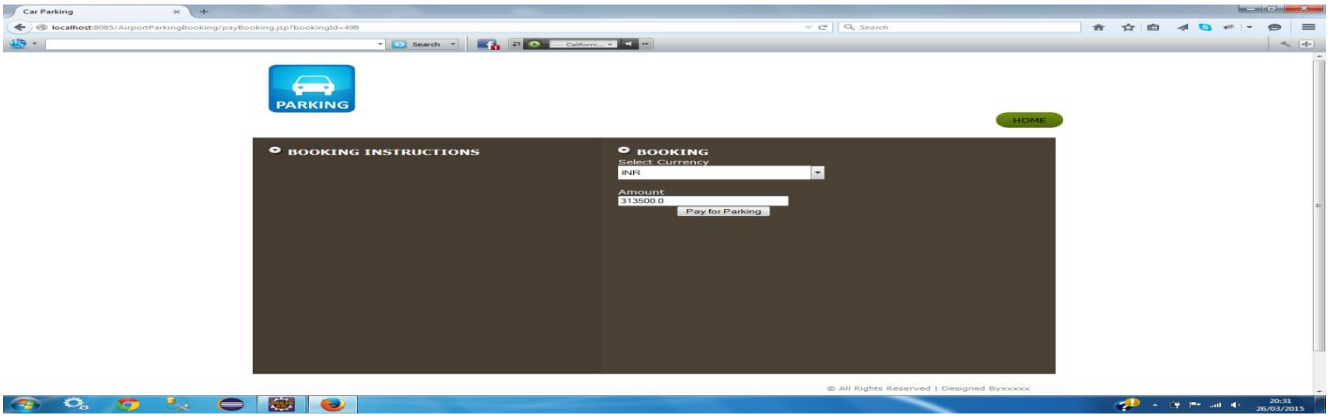
Select Currency: GBP

Amount: 3300.0


Car Parking
localhost:8085/AirportParkingBooking/payBooking.jsp?bookingId=498
Search
PARKING
HOME
BOOKING INSTRUCTIONS
BOOKING
Pay for Parking
© All Rights Reserved | Designed Byxxxxx
20:31
26/03/2015

Select Currency: CAD

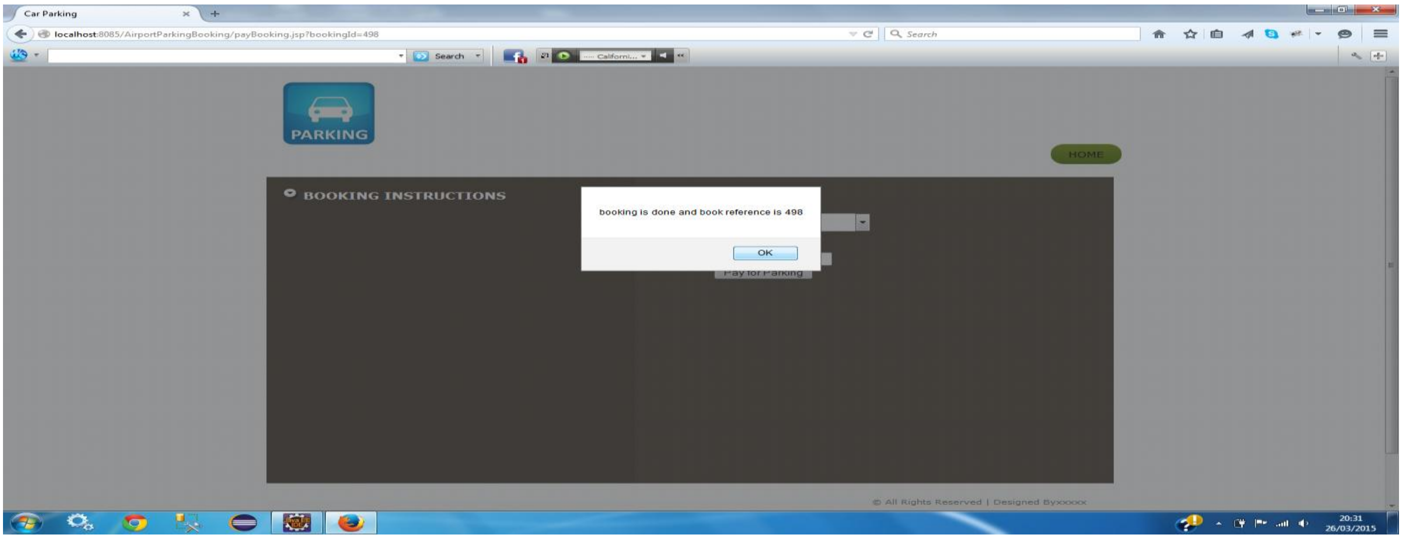
Amount: 3960.0


Car Parking
localhost:8085/AirportParkingBooking/payBooking.jsp?bookingId=498
Search
PARKING
HOME
BOOKING INSTRUCTIONS
BOOKING
Pay for Parking
© All Rights Reserved | Designed Byxxxxx
20:31
26/03/2015

Select Currency: INR

Amount: 313500.0

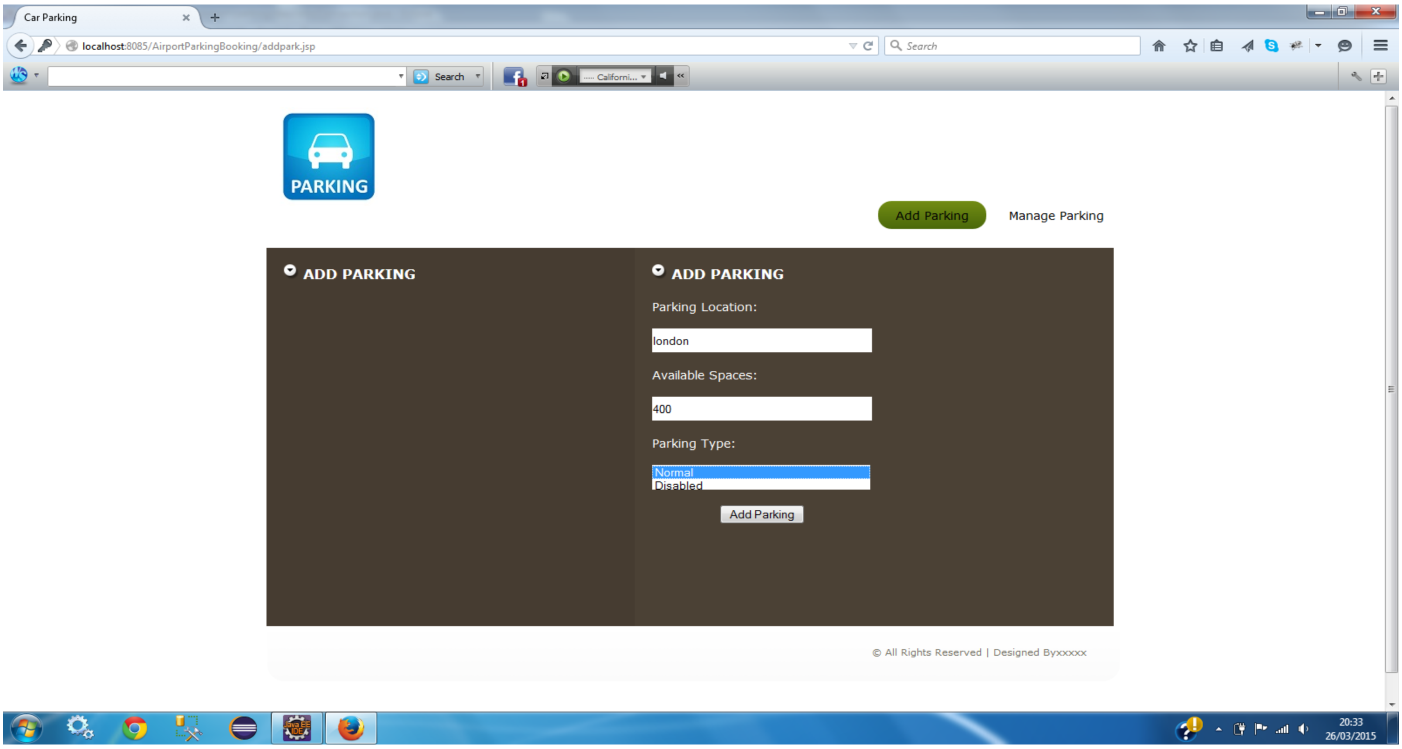
Once the payment option is selected, booking completion will be prompted along with the booking reference Id.


Car Parking
localhost:8085/AirportParkingBooking/payBooking.jsp?bookingId=498
PARKING
HOME
BOOKING INSTRUCTIONS
OK
Pay for Parking
© All Rights Reserved | Designed Byxxxxxx
20:31
26/03/2015

booking is done and book reference is 498

Employees can login with the provided username and password. They can add parking location, update the parking details and also remove the added parking details.

Add Parking Location


Car Parking
localhost:8085/AirportParkingBooking/addpark.jsp
PARKING
Add Parking
Manage Parking
ADD PARKING
Add Parking
© All Rights Reserved | Designed Byxxxxxx
20:33
26/03/2015

ADD PARKING

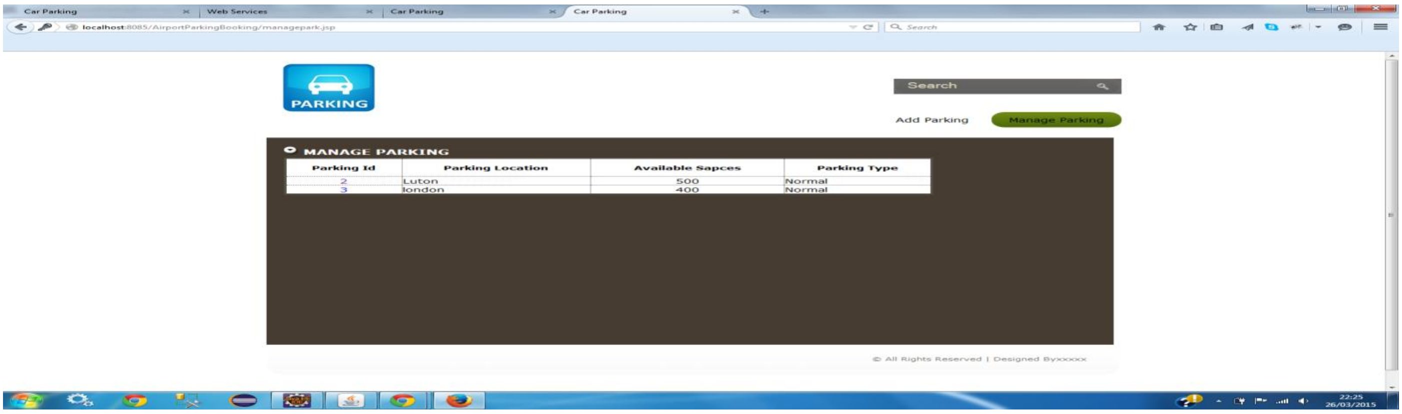
Parking Location: london

Available Spaces: 400

Parking Type: Normal / Disabled

Manage Parking

Complete list of parking location details will be shown, employees can select each location, and then they can edit/delete the location.

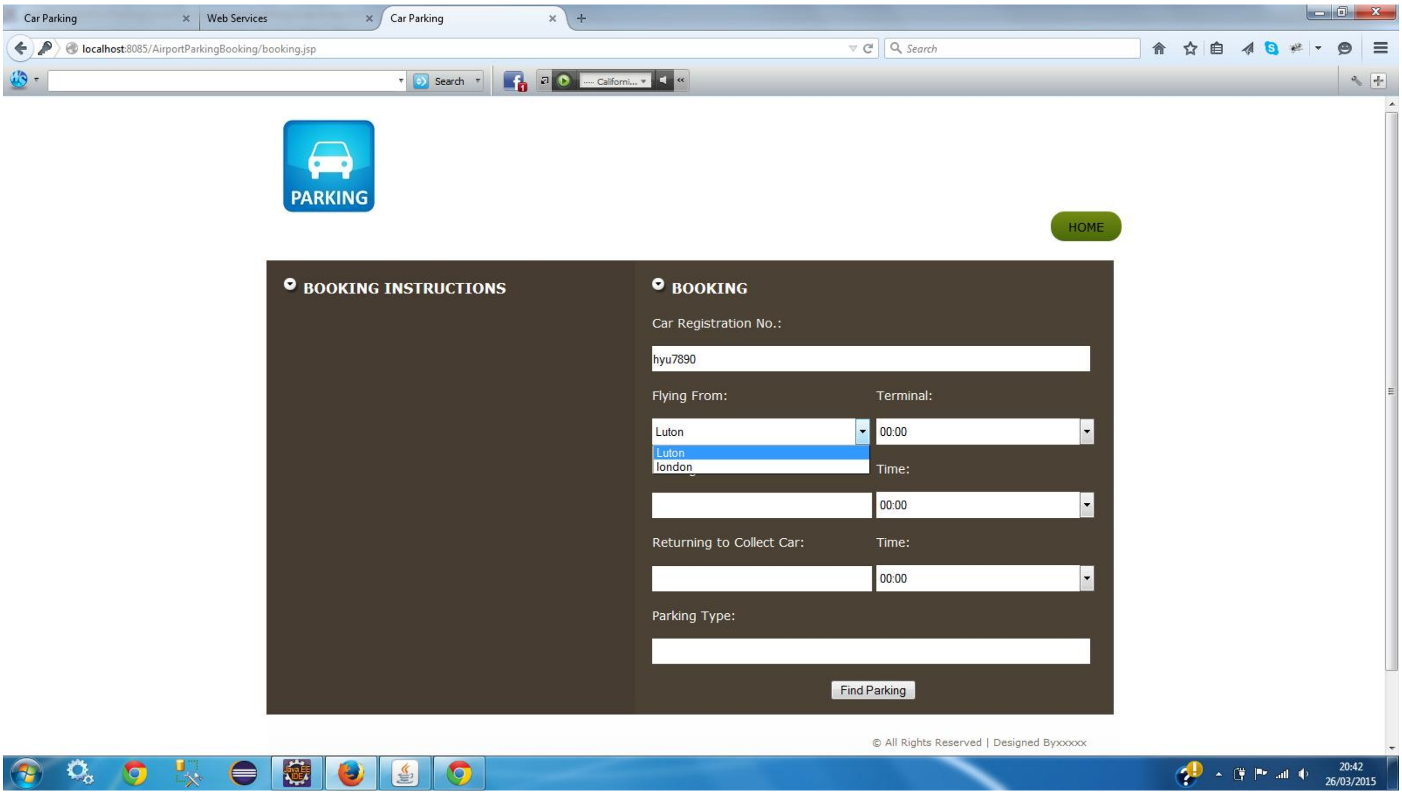


MANAGE PARKING

Parking Id	Parking Location	Available Sapces	Parking Type
2	Luton	500	Normal
3	london	400	Normal

© All Rights Reserved | Designed Byxxxxx

Locations added by employees are displayed as the options for customers to select and book parking lot.



BOOKING

Car Registration No.: hyu7890

Flying From: Luton

Luton
london

Terminal: 00:00

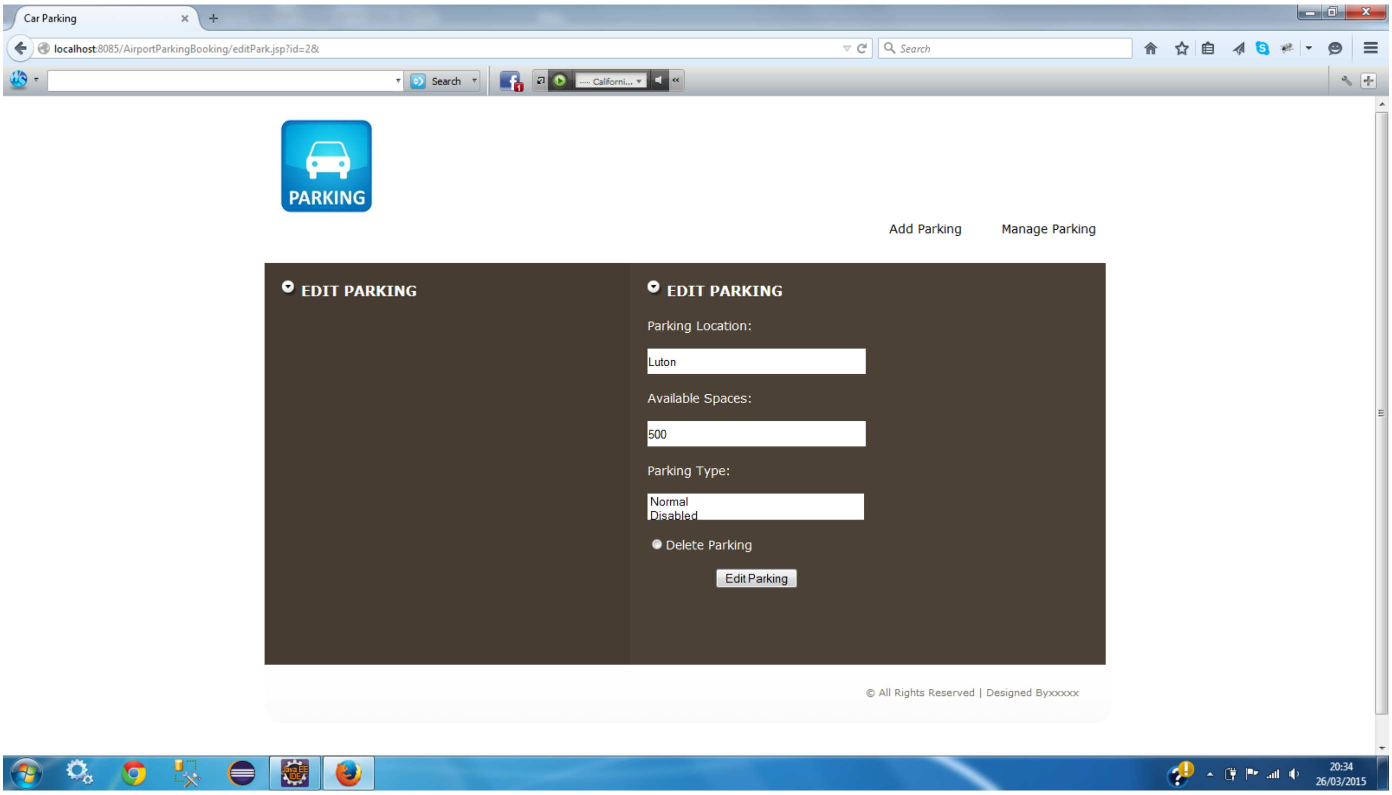
Time: 00:00

Returning to Collect Car:

Time: 00:00

Parking Type:

Edit/Delete Parking Details



Car Parking
localhost:8085/AirportParkingBooking/editPark.jsp?id=2&
PARKING
Add Parking
Manage Parking

EDIT PARKING

EDIT PARKING

Parking Location:

Luton

Available Spaces:

500

Parking Type:

Normal
Disabled

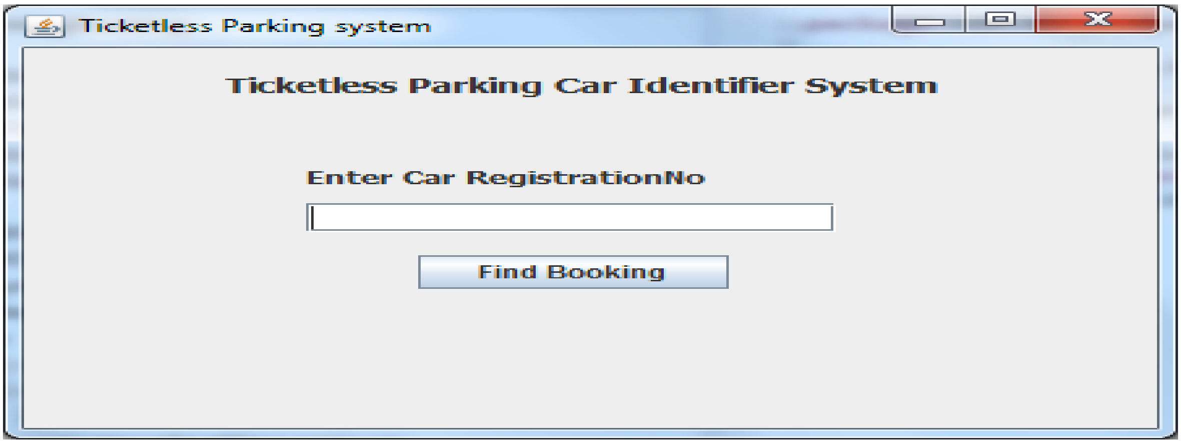
● Delete Parking

Edit Parking

© All Rights Reserved | Designed Byxxxxxx
20:34
26/03/2015

Ticketless system for car parking

A swing desktop application is developed where the system user enters the car registration number to identify the parking status.



Ticketless Parking system

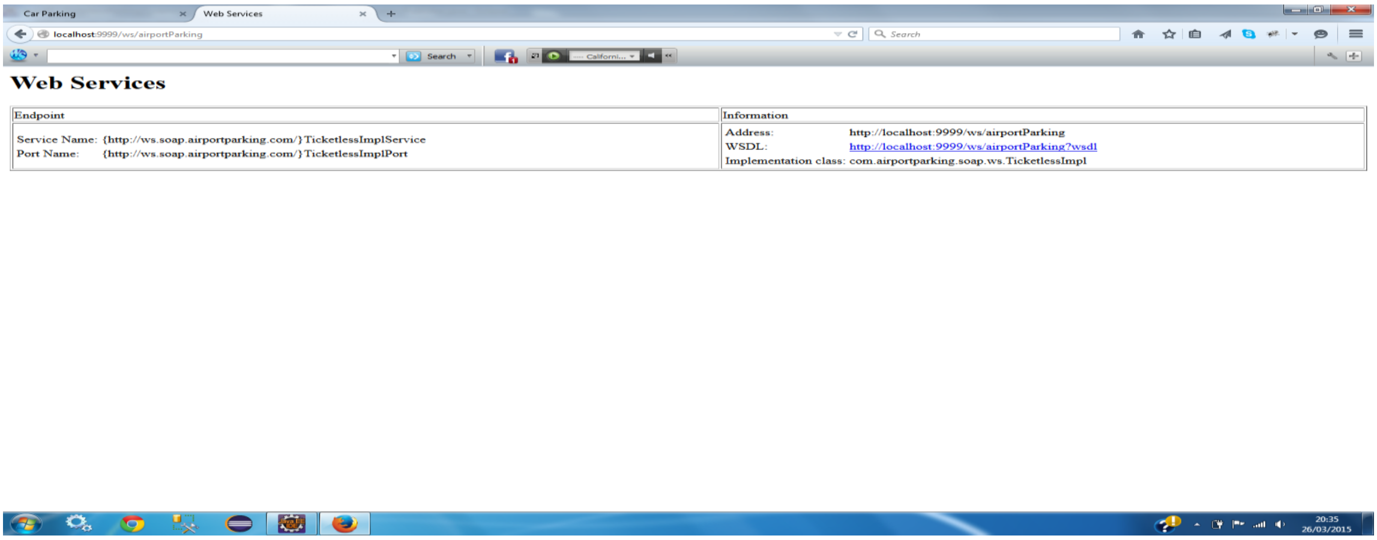
Ticketless Parking Car Identifier System

Enter Car RegistrationNo

Find Booking

Getting the registration number this application call SOAP web services, through which the application communicates with the Booking system database.

SOAP End Point



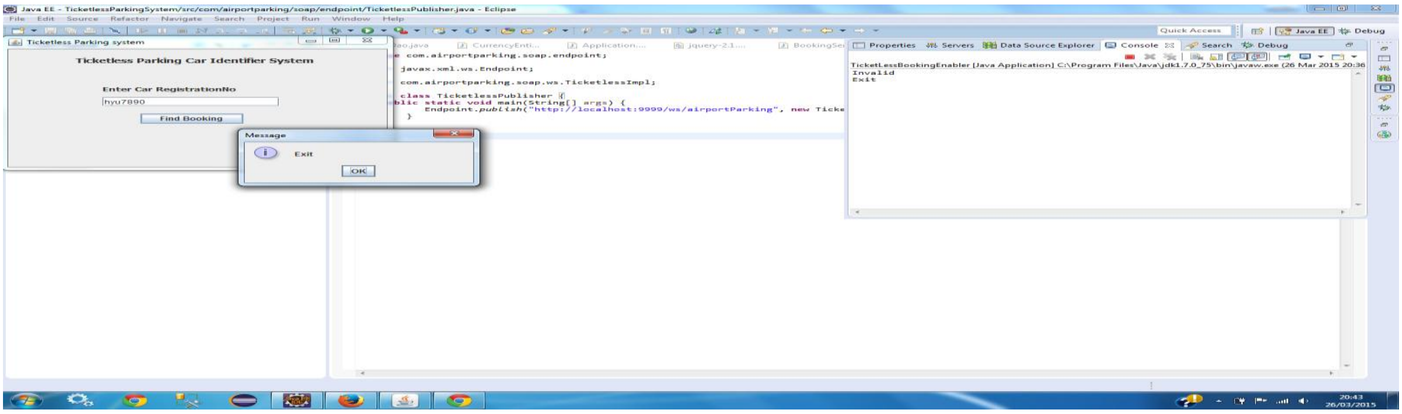
Web Services

Endpoint	Information
Service Name: {http://ws.soap.airportparking.com/}TicketlessImplService Port Name: {http://ws.soap.airportparking.com/}TicketlessImplPort	Address: http://localhost:9999/ws/airportParking WSDL: http://localhost:9999/ws/airportParking?wsdl Implementation class: com.airportparking.soap.ws.TicketlessImpl

This web service will return the booking from and To date from which the status messages are displayed to the user as follows.

Exit, Early, Invalid and Available

Exit is when the booking expires, early – there are still days for the car to reach the parking lot, Invalid – No booking Found. Available – car can come in for parking



Java EE - TicketlessParkingSystem/src/com/airportparking/soap/endpoint/TicketlessPublisher.java - Eclipse

Ticketless Parking system

Ticketless Parking Car Identifier System

Enter Car RegistrationNo: hyu7890

Message: Exit

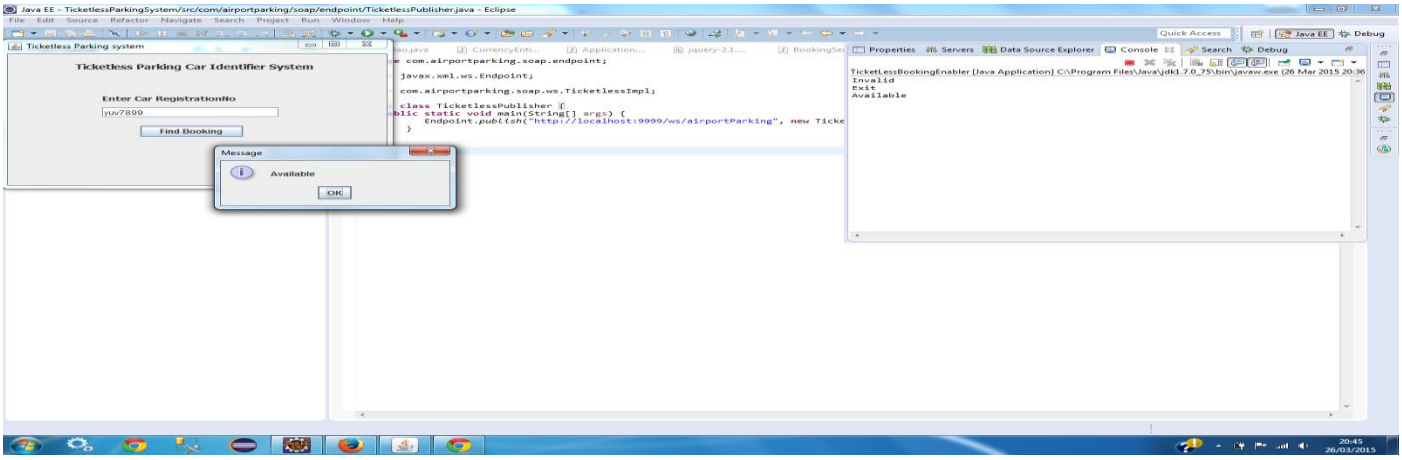
```
e com.airportparking.soap.endpoint;

 javax.xml.ws.Endpoint;

 com.airportparking.soap.ws.TicketlessImpl;

 class TicketlessPublisher {
blic static void main(String[] args) {
    Endpoint.publish("http://localhost:9999/ws/airportParking", new Ticke
 }
```

```
TicketLessBookingEnabler [Java Application] C:\Program Files\Java\jdk1.7.0_75\bin\javaw.exe (26 Mar 2015 20:36
Invalid
Exit
```



Java EE - TicketlessParkingSystem/src/com/airportparking/soap/endpoint/TicketlessPublisher.java - Eclipse

Ticketless Parking system

Ticketless Parking Car Identifier System

Enter Car RegistrationNo: yuv7899

Message: Available

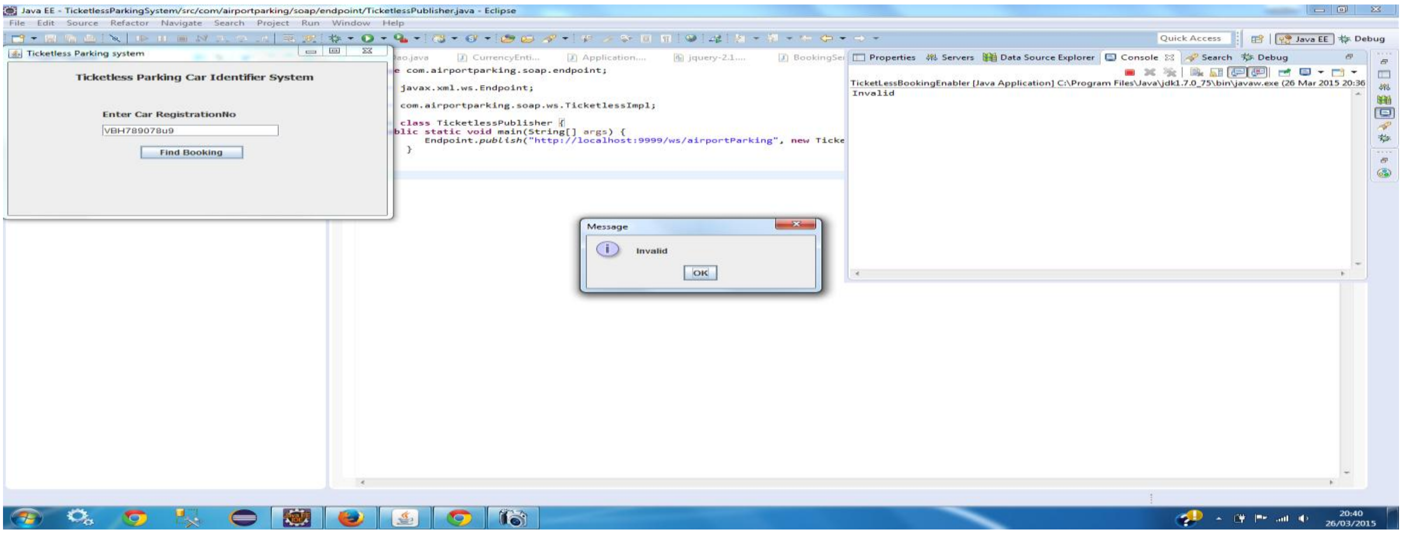
```
e com.airportparking.soap.endpoint;

 javax.xml.ws.Endpoint;

 com.airportparking.soap.ws.TicketlessImpl;

 class TicketlessPublisher {
blic static void main(String[] args) {
    Endpoint.publish("http://localhost:9999/ws/airportParking", new Ticke
 }
```

```
TicketLessBookingEnabler [Java Application] C:\Program Files\Java\jdk1.7.0_75\bin\javaw.exe (26 Mar 2015 20:36
Invalid
Exit
Available
```



Java EE - TicketlessParkingSystem/src/com/airportparking/soap/endpoint/TicketlessPublisher.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Java EE
Debug
Ticketless Parking system
ao.java
CurrencyEnti...
Application....
jquery-2.1....
BookingSe
Properties
Servers
Data Source Explorer
Console
Search
Debug
Message
OK
20:40
26/03/2015

Ticketless Parking Car Identifier System

Enter Car RegistrationNo

VBH789078u9

Find Booking

```
e com.airportparking.soap.endpoint;

 javax.xml.ws.Endpoint;

 com.airportparking.soap.ws.TicketlessImpl;

 class TicketlessPublisher {
blic static void main(String[] args) {
     Endpoint.publish("http://localhost:9999/ws/airportParking", new Ticke
  }
```

Invalid

```
TicketLessBookingEnabler [Java Application] C:\Program Files\Java\jdk1.7.0_75\bin\javaw.exe (26 Mar 2015 20:36
Invalid
```

3. Evolution of the Application

3.1. Booking JSP Application

There are three phases carried out in the application development, which is design, development and Unit Testing.

Design is done using html and css. Development technologies used here are JSP and servlets. MVC pattern is applied for the development and the database transactions are carried out as entities. Development Server used is Apache Tomcat 7.0

3.2. RestFUL web application

To implement restful web services, separate application is developed using Spring and REST framework. Generated URL is provided to the service calling currency values. This application uses separate database called currency database to retrieve and return values. Development Server used is Apache Tomcat 7.0

3.3. Ticketless car Identifier

A swing desktop application is developed to call the SOAP web services which take car registration number as input and returns the booking dates. With which the desktop application identifies the car's booking status as explained in above screenshots.

4. Frameworks applies

4.1. Booking JSP Application

Framework	JAVA EE
Technologies implemented	Servlets, JSP
Design	HTML, CSS
Server	Apache Tomcat 7.0
Database	MY SQL.
Development Pattern	MVC

4.2. RestFul web Application

Framework and Build	Spring and Maven
Technologies implemented	RestFul, Java
Server	Apache Tomcat 7.0
Database	MY SQL.
Development Pattern	MVC

4.3. Ticketless car identifier

GUI	Swing
Technologies implemented	Java
Web Ser vices	SOAP
Server	Apache Tomcat 7.0
Database	MY SQL.
Development Pattern	MVC