Chapter 1

INTRODUCTION

Many people find it hard to contact and go to the bus counter each time they want to find whether there is available ticket on the time they want to depart.It will take quite some time to do the process of finding the available ticke and often involved some cost to do so. Nowadays, mobile application would be a more popular choice for user to help them to solve their daily lives problem.

This mobile application is operate in smartphones operating system such as iOS, Android and Windows OS. There are a lot of smartphones sold in the market such as the iPhone, Samsung, HTC, and Sony Ericsson. It could be accessed by anyone who using smartphones. This project is developed using Android Operating System as it is the most popular operating system which is widely used by smartphones. Mobile Application of bus reservation system for Bus , Bus booking system is an web application that helps the bus company staff and the costumer to reserve the bus tickets. TT Bus Sdn. Bhd with its continuing aim to provide quality services and giving the best facilities to their customer had asked to developed a mobile application system to help them solve the bus reservation issue. It is developed so that the costumer can easily

check the bus availability status. The system must have a database which contains the customer information, it must be able to accurately give the bus information such as the bus schedule and also have a nice interface to make it easier to use. Finally, the application must also be a user-friendly for commercial purpose.

Chapter 2

LITERATURE REVIEW AND PROBLEM STATEMENT

2.1 Introduction to Database Management System:

DBMS stands for Database Management System. We can break it like this DBMS=Database+Management System. Database is a collection of data and Management System is a set of programs to store and retrieve those data. Based on this we can define DBMS like this: DBMS is a collection of inter-related data and set of programs to store and access those data in an easy and effective manner.

Database systems are basically developed for large amount of data. When dealing with huge amount of data, there are two things that require optimization: Storage of data and retrieval of data. According to the principles of database systems, the data is stored in such a way that it acquires a lot less space as the redundant data(duplicate data) has been removed before storage.

Along with storing the data in an optimized and systematic manner, it is also important that we retrieve the data quickly when needed. Database system ensures that data is retrieved as quickly as possible. A database management system is a collection of programs that enables users to create and maintain a database.

The DBMS is general purpose software system that facilitates the process of defining, constructing, manipulating and sharing databases among various users and applications. Defining a database involves specify the data types, structures and constraints of the data to be stored in the database. The database definition or descriptive information is also stored in the form of a database catalogue or dictionary. It is called Meta data.

○ Applications of Database

The development of computer graphics has been driven both by the needs of the user community and by the advances in hardware and software. The applications of database are many and varied; it can be divided into four major areas:

1. Hierarchical and network

2. Flexibility with relational database

3. Object oriented application.

4. Interchanging the data on the web for e-commerce.

5. Extending database capabilities for new applications

6. Databases versus information retrieval work system.

○ Hierarchical and Network System:

Many early database applications maintained records in large organization such as corporations, universities, hospitals, and banks. In many of these applications, there were large numbers of records of similar structure. For example in a university application, information would be kept for each student, each course, each grade record and so on. There were also many interrelationships among them .Another short coming of early systems was that they provided only programming language interfaces .This made it time consuming and expensive to implement a new queries and transactions .Since new program had to be written, tested and debugged.

○ Flexibility with relational database:

Relational database were originally proposed to separate the physical storage of data from its conceptual representation and to provide a mathematical foundation for content storage. The relational data model also introduced high-level query languages that provided an alternative to programming language interfaces, hence,it was a lot quicker to write new queries.
Relational systems were initially targeted to the same applications as earlier systems. But

were meant to provide flexibility to develop new queries quickly and to recognise the database as requirements change.

- **Object oriented application:**

The object oriented databases were considered a competitor to relational databases, since they provide a more general data structures. They also incorporated many of the useful object-oriented paradigms, such as abstract data types, encapsulation of operations, inheritance, and object identity. However, the complexity of the model and the lack of an early standard contributed to their limited use. They are now mainly used in specialized applications, such as engineering design, multimedia publishing, and manufacturing systems. Despite expectations that they will make a big impact, their overall penetrations into the database product market remains under 5% today.

- **Interchanging the data on the web for e-commerce:**

The World Wide Web provides a large network of interconnected computers. Users can create documents using a Web publishing language, such as Hyper Text Mark-up Language (HTML) and stores these documents on the web server who other users (clients) can access them. Documents can be linked through hyperlinks, which points to other documents. It quickly became apparent that parts of the information on e-commerce web pages were often dynamically extracted data from DBMSs. A variety of technique uses were developed to interchange the data on the web.

- **Extending Databases Capabilities for New Applications:**

The success of database systems in traditional applications encouraged developers of other types of applications to attempt to use them. Such applications traditionally used their own specialized file and data structures. Scientific applications that store large amount of data resulting from scientific experiments in areas such as high energy physics or the mapping of the human genome. Storage and retrieval of images, from scanned news or personal photographs to satellite photograph images and images from medical procedure such as x rays and MRI.

- **Databases versus information retrieval:**

Traditionally, database technology applies to structured and formatted data that arises in routine applications in government, businesses, an industry .Database technology is heavily used in manufacturing, retail, banking, insurance, finance, health care industries where structured data originates as form such as invoices or patient registration documents .There has been a concurrent development of field called information retrieval(IR) that deals with books and various forms of based articles. Data is indexed, catalogued and annotated using keywords.

2.2What is MySQL?

MySQL is multithreaded, multi user SQL database management system (DBMS).The basic program run as server providing multi user access to a number of databases.The project's source code is available under terms of the GNU General Public Liquescence, as well as under a variety of property arguments. MySQL is a database .The data in a MySQL is stored in a Database objects called tables. A table is a collection of related data entries and it consists of columns and rows.The databases are useful when storing information categorically. A Higher primary school textbook distribution database may have tables such as "school","subject","cluster",etc.

MySQL is a central component of the LAMP open source web application software stack (and other "AMP" stacks).LAMP is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. Application that use the MySQL database include TYP03,MODx, Joomla, Word Press, phpBB,MyBB and Drupal. MySQL is also used in many high profiles, large scale web sites, including Google (Though not for the searches).

Chapter 3

PROBLEM FORMULATION

3.1 Problem Statement:

The database aims at maintaining the records of Bus details to be supplied to the passenger. It mainly aims that each and every passenger in that particular cluster gets ticket booked and hence the problem of standing in queue is reduced.

- Currently, staffs at the ticket counter are using the manual system to sell tickets and manage the bus seat booking
- The booking system is done manually by the staffs using the booking book. It will take quite some time to complete this process. Therefore, it results in long queue to buy bus ticket. This issue makes the bus company services is inconvenient to customer and wasting the customer's time.
- In addition, the issue of bus telephone line needs to be solved. Since there is only one official contact number to make reservation, customer often finds the telephone line is busy. It will take quite some cost for customer when the line is not reached.
- So, it is developed in HTML, JSP, and MYSQL it has been implemented in WINDOWS platform.

3.2 Aim of the work

The scope of the project is managing a consistency and storage of data by dedicated data administrator .It provides most of the features that a Database Management System should have. It is developed by using MySQL database. It has been implemented in WINDOWS platform.

3.3 Objectives

- The main objective of the bus booking system is to manage the details of bus ,ticket, booking, agents ,seats.
- It manages all the information about bus ,customer ,seats.
- The project is totally build at administrative end and thus only the administrator is guaranteed the access.
- The of the project is to build an application program to reduce the manual work for managing the bus ,ticket ,customer, booking.

Chapter 4

Requirement Specification

4.1 Functional Requirements

- **Admin Login :**

  1. Add route details, bus details ,ticket fare
  2. Update bus ,route details
  3. Delete route, bus details
  4.Veiw passenger details
  5. Logout

- **User Login :**

  1. Book bus ticket
  2. Cancelation of Booking
  3. Veiw bus details
  3. Logout

4.2 Non Functional Requirements

Sl.No	Hardware/Equipment	Specification
1	Processor	Intel Processor
2	Clock speed	1.8GHz
3	Screen Resolution	1024*786 Pixel
4	Keyboard	QWERTY
5	RAM	4 GB

4.3 Software Requirements

Sl.No	Software	Specification
1	Web Browser	Google chrome
2	Operating System	Windows 10
3	Scripting Languages	HTML, CSS, PHP, JavaScript
4	Database	MySQL 8.0.12
5	Server used	Apache Tomcat server
6	Workbench used	Eclipse IDE 2019-09

Chapter 5

Database Design

5.1 Entity-Relationship Diagram

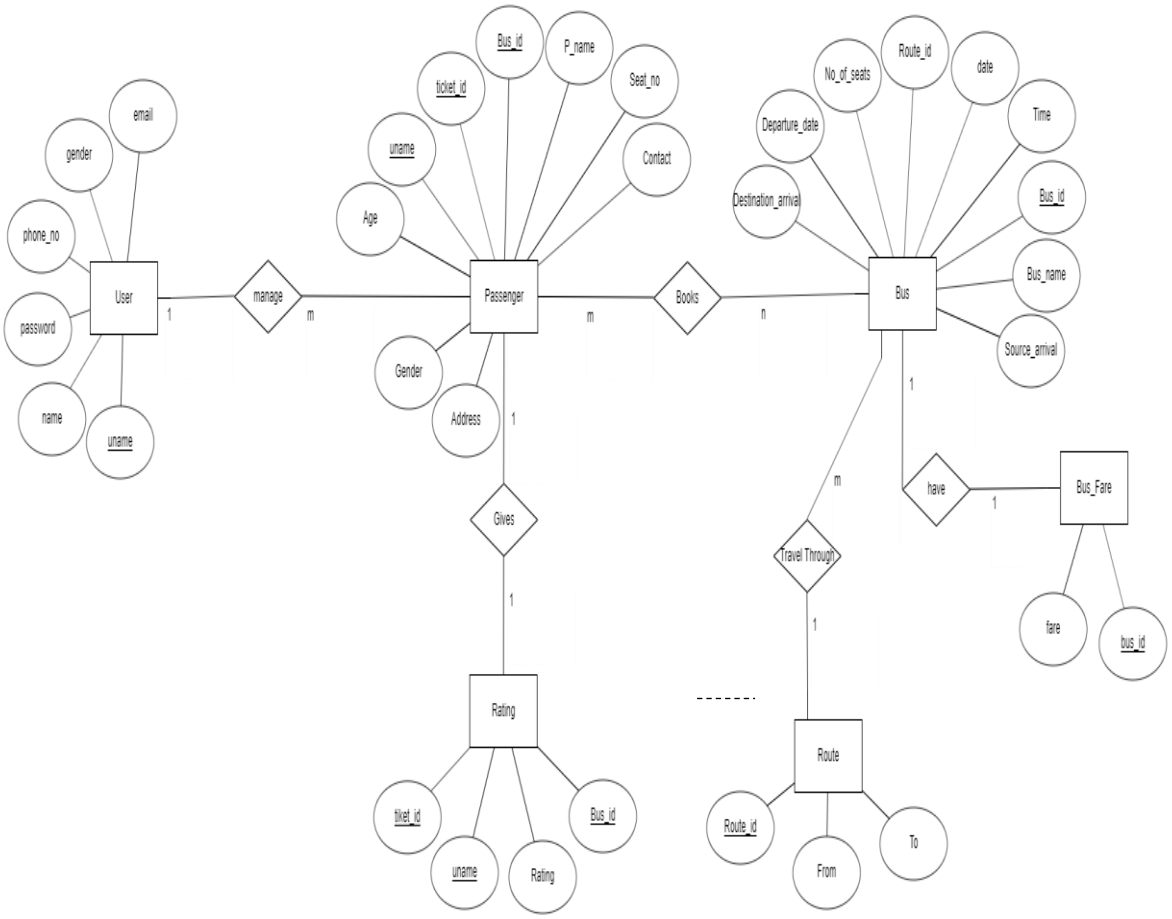

Figure no 5.1: E-R diagram of Bus Booking System

5.2 Conversion from ER diagram to Schema:

In order to convert from Entity-Relationship diagram to Schema Diagram need to apply the following methods such as relational mapping algorithm, normalization and must follow the application development process.

- **Relational database design using ER to relational mapping algorithm:**

ER to relational mapping algorithm has following steps:

Step 1: Mapping of regular entity types

Step 2: Mapping of weak entity types

Step 3: Mapping of binary one to one relationship types

Step 4: Mapping of binary one to N relationship types

Step 5: Mapping of binary M:N relationship types

Step 6: Mapping of multivalued attributes

Step 7: Mapping of N-ary relationship types (N>2)

5.3 Normalization:

Normalization is the process of decomposing larger relations into smaller relations by analysing the given relation schemas, based on their functional dependencies and primary keys to achieve the desirable properties of

(i) Minimizing redundancy
(ii) Minimizing insertion, deletion and updation anomalies.

There are mainly five types of Normalization, but in this particular project we are mainly deals with the first three types of Normalization.

○ First Normal Form:

A table is said to be in 1 NF if all the attributes of a table are atomic (simple, indivisible) and single valued.

○ Second Normal Form:

A table is said to be in 2 NF if

(i) The table should be in 1NF

(ii) There should not be any partial dependencies in the table

○ Third Normal Form:

A table is said to be in 3 NF if

(i) The table should be in 2NF

(ii) There should not be any transitive dependency in the

5.4 Schema Diagram:

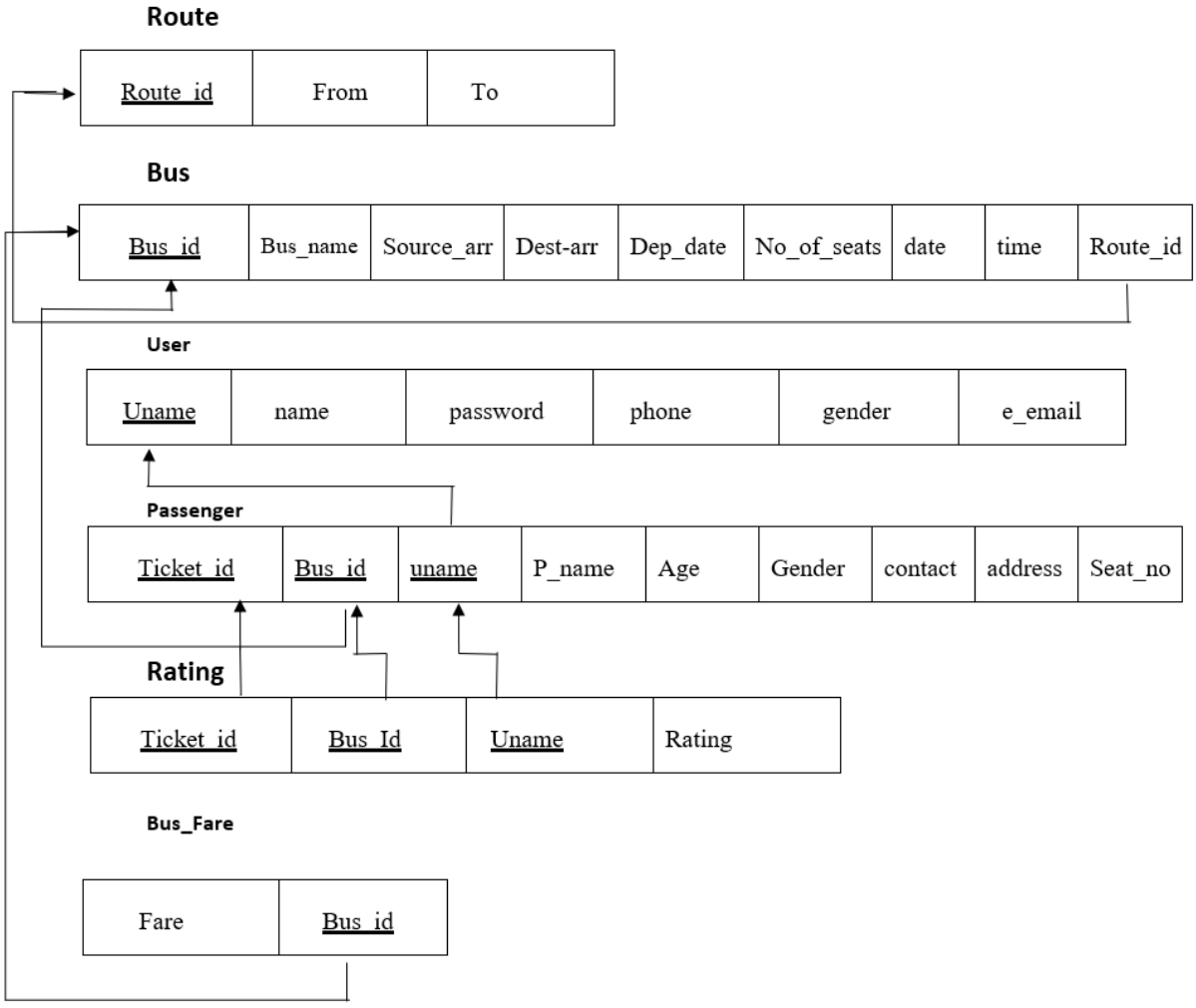

Figure no.5.4: Schema diagram of Bus Booking System

Chapter 6

IMPLEMENTATION

The project is implemented using MySQL database along with JSP and HTML. In this project, we use the below mentioned concepts for implementing different type of effects like

HTML pages to demonstrate on a web page as a stand-alone or web based application. It is a three tier architecture project. A three-tier architecture is a client-server architecture in which the functional process logic, data access, computer data storage and user interface are developed and maintained as independent modules on separate platforms. Three-tier architecture is a software design pattern and a well-established software architecture.

6.1 TOOL DESCRIPTION:

○ User Interface:HTML

The user interface (UI),in the industrial design field of human-computer interaction in the space where interactions between humans and machines occur. The goal of this interactions is to allow effective operations and control of the machine from the human end, whilst the machine simultaneously feeds back information that aids the operators decision making process. In this particular project, we used HTML as user interface tool. The main goal of this HTML user interface design is to produce a user interface which makes it easy(self-explanatory) efficient, and enjoyable(user friendly) to a machine in the way which produces the desired result. This generally means that the operator needs to provide minimal input to achieve the desired output, and also that the machine minimizes undesired outputs to the human.

○ Overview of JSP

Java Server Pages (JSP) is a server-side programming technology that enables the creation of dynamic, platform-independent method for building Web-based applications. Architecturally, JSP may be viewed as a high-level abstraction of Java servlets. JSPs are translated into servlets at runtime, therefore JSP is a Servlet.

The SELECT statement is used to select data from a database. The result is stored in a result table, called the result-set. The WHERE clause is used to extract only those records that fulfill a specified criterion.

Syntax: SELECT column_name(s)
FROM table_name
WHERE column_name operator value;

○ Overview of JDBC

JDBC stands for Java Data Base Connectivity. JDBC is a collection of Java classes and interfaces that enables database access java programs. The classes and interfaces are part of the java .sql package. There are FIVE steps in the JDBC process for a java program to communicate with database.

- Loading the Driver.
- Connect to the DBMS.
- Create and Execute the SQL Statement.
- Process Data returned by the DBMS.
- Terminate the Connection to the DBMS

○ ApplicationServer:Servlet

An application server is a software frame work that provides both facilities to create web applications and a server environment to run them. Application server frame work contains a comprehensive service layer model. An application server acts as a set of component accessible to the software developer through a standard API defines for the platform itself. For web applications, these components are usually performed in the same running environment as their Web server(s), and their main job is to support the construction of dynamic pages. In the case of java application server, the server behaves like an extended virtual machine for running applications, transparently handling connection to the database on one side, and often, connections to the Web client on the other.

The bottom layer includes all data management services. If the bottom layer is split into two layers(a Web server and a database server),then this becomes a four tier architecture. It is customary to divide the layers between the user and the stored data further into final components; thereby giving rise to n-tier architecture where n may be 4 or 5. Typically, the business logic layer is divided into multiple layers. Besides distributing programming and data through a network, entire applications afford the advantage that any one tier can run on an appropriate processor or operating system platform and can be handled independently. Another

layer typically used by vendors of ERP(Enterprise Resource Planning),and CRM(Customer Relationship Management) packages is the middleware layer which accounts for the front end modules communicating with a number of back-end databases.

In this particular project, we need to implement mainly the two mentioned concepts such as

(i) Stored procedure

(ii) Triggers

6.2 Stored Procedures:

A stored procedure is a set of Structured Query Language(SQL) statements with an assigned name, which are stored in a relational database management system as a group, so it can be reused and shared by multiple programs. Stored procedure can access or modify data in a database, but it is not tied to a specific database or object, which offers a number of advantages.

6.3 Triggers:

A database trigger is a procedural code that is automatically executed in response to certain events on a particular table or view in a database. The trigger is mostly used for maintaining the integrity of the information on the database.

6.4 Scripts

- **Welcome page script:**

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="ISO-8859-1">
  <meta charset='utf-8'>
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <script src="http://code.jquery.com/jquery-latest.min.js" type="text/javascript"></script>
  <script src="script.js"></script>
  <title>Insert title here</title>
<link rel="stylesheet" href="css/style.css">
   <meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
<div id='cssmenu'>
<ul>
  <li class='active'><a href='#'>Home</a></li>
  <li><a href='Admin.html'>Admin Login</a></li>
  <li><a href='Userlog.html'>User Login</a></li>
</ul>
</div>


<div class="contact-section">
<h1>Bus Booking System</h1>
</body>

</html>
```

- **Insertion script:**

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@page import="java.sql.*" %>
        <%@page import="java.io.*" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<%!int flag; %>
<%
int val=0;
String bus_id=request.getParameter("bi");
String bus_name=request.getParameter("bn");
String sorce_arrival=request.getParameter("sa");
String dist_arr=request.getParameter("da");
String depart_date=request.getParameter("dd");
String no_of_seat=request.getParameter("ns");
String src=request.getParameter("rt");
String drc=request.getParameter("rtt");
String route="";
Class.forName("com.mysql.jdbc.Driver");
String url=("jdbc:mysql://localhost:3306/bus_booking");
String un="root";
```

```
String pwd="root";

Connection con=DriverManager.getConnection(url,un,pwd);

PreparedStatement os=con.prepareStatement("select r_id from route where src=? and
distination=?");

os.setString(1,src);

os.setString(2,drc);

        ResultSet rs=os.executeQuery();


if(rs.next())

{

 val=1;

route=rs.getString(1);

}

if(val==0)

{

out.println("<script>alert('Route Does Not
Exists');window.location='buses_input.jsp';</script>");

}

%>

<%

PreparedStatement ps=con.prepareStatement("insert into bus
values(?,?,?,?,?,?,?,CURRENT_DATE(),CURRENT_TIME())");

ps.setString(1,bus_id);

ps.setString(2,bus_name);

ps.setString(3,sorce_arrival);

ps.setString(4,dist_arr);

ps.setString(5,depart_date);

ps.setString(6,no_of_seat);
```

```
ps.setString(7,route);
%>
<%try{%>
<%  flag=ps.executeUpdate();%>
<%out.println("<script>alert('Inserted Sucessfully');window.location='busdis.jsp';</script>");
%>
<% }
catch(Exception e)
{%>
        <%out.println("<script>alert(' Not Inserted
Sucessfully');window.location='buses_input.jsp';</script>");
%>
<%}%>
</body>
</html>
```

- **Stored Procedure Script:**

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
   <%@page import="java.sql.*" %>
      <%@page import="java.io.*" %>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<body>
<%int i=0,j=0; %>
<%
```

```
if(null == session.getAttribute("username"))
{
response.sendRedirect("Userlog.html");
}
%>
<%!String user_id; %>
<%
user_id=(String)session.getAttribute("username");
Class.forName("com.mysql.jdbc.Driver");
String url=("jdbc:mysql://localhost:3306/bus_booking");
String un="root";
String pwd="root";
Connection con=DriverManager.getConnection(url,un,pwd);
%>
<%CallableStatement st=con.prepareCall("{call display1(?)}");
st.setString(1,user_id);
st.execute();
ResultSet rs=st.executeQuery();
%>
<center>
<p><h3 style="color:brown;"> Booking  Details</h3></p>
<div class="wrapper">

 <div class="table">

   <div class="row header">
    <div class="cell">
      Booking Date
    </div>
```

```
<div class="cell">
      Booking Time
    </div>
    <div class="cell">
      Ticket Id
    </div>
    <div class="cell">
    Bus Id
    </div>
    <div class="cell">
    Bus Name
    </div>
    <div class="cell">
    Passenger Name
    </div>
    <div class="cell">
    Gender
    </div>
<div class="cell">
    Source
    </div>
<div class="cell">
    Scheduled Arrival
    </div>
<div class="cell">
     Distination
    </div>
<div class="cell">
      Destination Arrival
```

```
    </div>
<div class="cell">
      Departure date
    </div>
<div class="cell">
   Seat No
    </div>
<div class="cell">
    Fare
    </div>
    <div class="cell">
    Update
    </div>
    <div class="cell">
    Delete
    </div>
    <div class="cell">
    Your Ratings
    </div>
    <div class="cell">
    Share Your Experience
    </div>
  </div>
<%while(rs.next()) {%>
<div class="row">
<div class="cell" data-title="Bus_ID">
      <%=rs.getString(14) %>
    </div>
    <div class="cell" data-title="Bus name">
```

```
      <%=rs.getString(13) %>
    </div>
    <div class="cell" data-title="Seating Capacity">
     <%=rs.getString(1) %>
    </div>
<div class="cell" data-title="Source">
     <%=rs.getString(2) %>
    </div>
<div class="cell" data-title="Scheduled Arrival">
     <%=rs.getString(3) %>
    </div>
<div class="cell" data-title="Destination">
     <%=rs.getString(4) %>
    </div>
<div class="cell" data-title="Scheduled destination">
     <%=rs.getString(5) %>
    </div>
<div class="cell" data-title="Departure date">
     <%=rs.getString(6) %>
    </div>
<div class="cell" data-title="Fare">
     <%=rs.getString(7) %>
    </div>
    <div class="cell" data-title="Fare">
     <%=rs.getString(8) %>
    </div>
    <div class="cell" data-title="Fare">
     <%=rs.getString(9) %>
    </div>
```

```
    <div class="cell" data-title="Fare">
     <%=rs.getString(10) %>
    </div>
    <div class="cell" data-title="Fare">
     <%=rs.getString(11) %>
    </div>
    <div class="cell" data-title="Fare">
     <%=rs.getString(12) %>
    </div>
    </div>
   </div>
<%i=1; %>
<%}%>
</body>
</html>
```

Stored Procedure :

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `display1`(in un varchar(15))
BEGIN
Select
t_id,p.b_id,b.b_name,p_name,gender,src,scdl_srcarrv,distination,scdl_destarr,
dep_date,seat_no,t.tik_price,p.time_,p.date
```

```
from passenger p,ticket t,bus b,route r
where p.b_id=t.b_id and p.b_id=b.b_id and r.r_id=b.r_id and uname=un;
END
```

Triggers:

1.
```
CREATE TRIGGER `dbms`.`consumer_AFTER_DELETE` ON `consumer`
FOR EACH ROW
BEGIN
insert into seat values(old.t_id,old.b_id,old.seat_no);
insert into passdlt values(old.uname,old.p_name,old.gender,old.b_id,old.seat_no,
old.t_id,CURRENT_DATE(),CURRENT_TIME());
END
```

Chapter 7

Results and Discussion

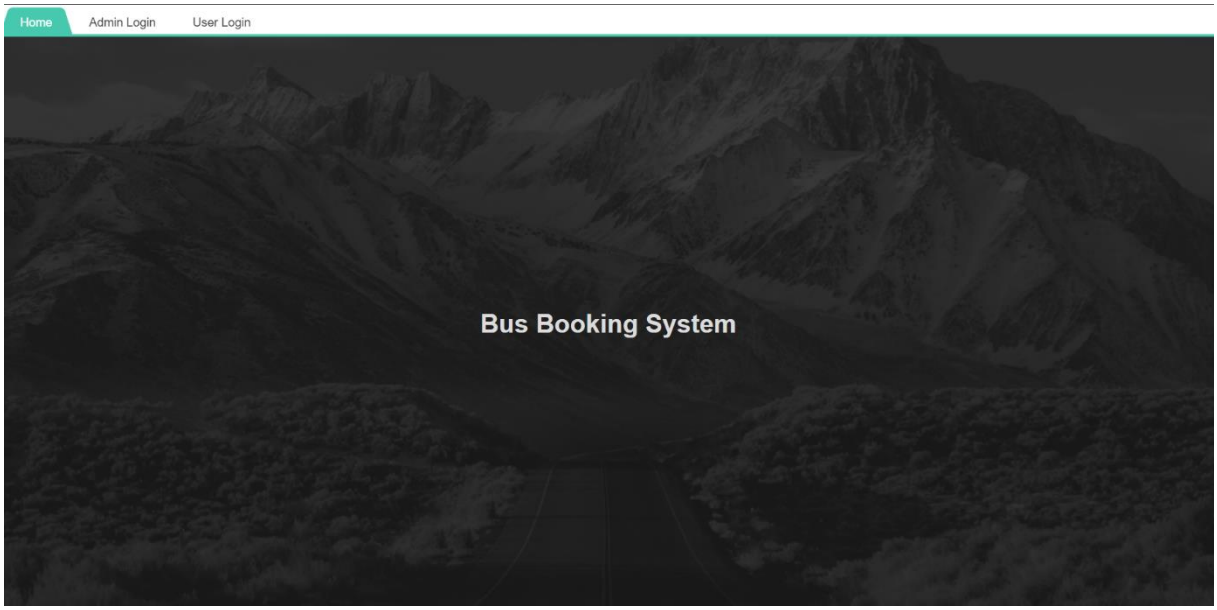

Figure no 7.1: Welcome

The above snapshot is the beginning of bus booking system it contains the admin and the user login.

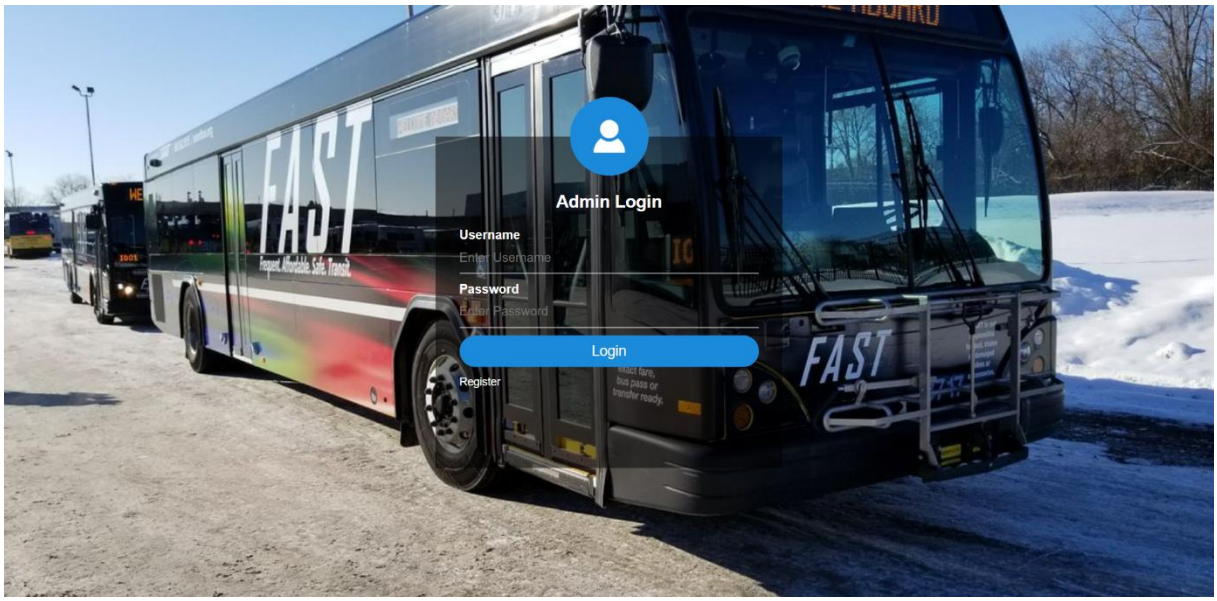

Figure no 7.2: Admin login

Admin login takes place from above snapshot

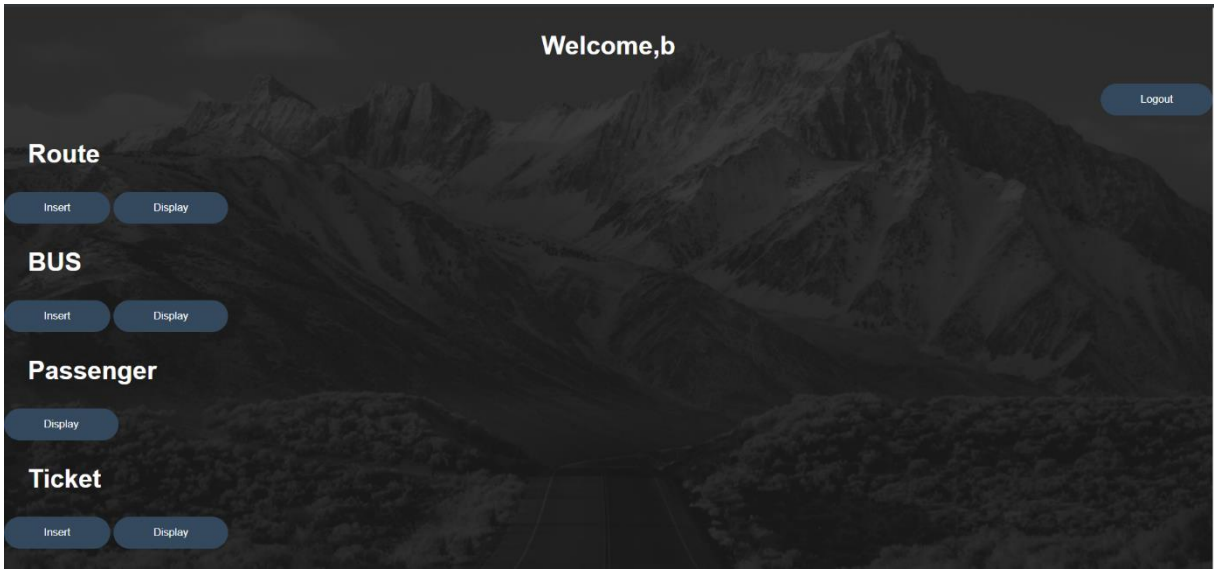

Figure no 7.3 Admin menu

The above snapshot is the display of admin menu ,the menu contains Route,Bus,Passenger,Ticket

Bus Details

Date	Time	Bus_ID	Root ID	Bus Name	Seating Capacity	Source	Scheduled Arrival	Distination	Destination Arrival	Departure Date	Update	Delete
2019-12-02	23:24:44	1	1	neetha	5	mumbai	12:11:00	pune	12:44:00	2019-01-19	Update	Delete
2019-12-03	10:20:37	2	1	Neeta travels	8	mumbai	12:05:00	pune	12:05:00	2019-01-18	Update	Delete
2019-12-03	10:20:43	3	1	Neeta travels	8	mumbai	12:05:00	pune	12:05:00	2019-01-18	Update	Delete
2019-12-03	10:21:04	4	4	vishal travels	8	mumbai	12:05:00	mangalore	12:05:00	2019-01-18	Update	Delete
2019-12-03	10:21:38	5	11	jolly travels	8	pune	12:05:00	agra	12:05:00	2019-01-18	Update	Delete
2019-12-03	10:21:57	6	10	ksrtc travels	8	pune	12:05:00	bangalore	12:05:00	2019-01-18	Update	Delete
2019-12-03	10:22:09	7	9	bmtc travels	8	delhi	12:05:00	bangalore	12:05:00	2019-01-18	Update	Delete
2019-12-03	10:22:38	8	5	h.t.co travels	8	nashik	12:05:00	mangalore	12:05:00	2019-01-18	Update	Delete

Figure no 7.4 : Bus details page

The above snapshot is used display the bus details

Figure no 7.5 : Register Page.

This is the registration page for both admin and user

.

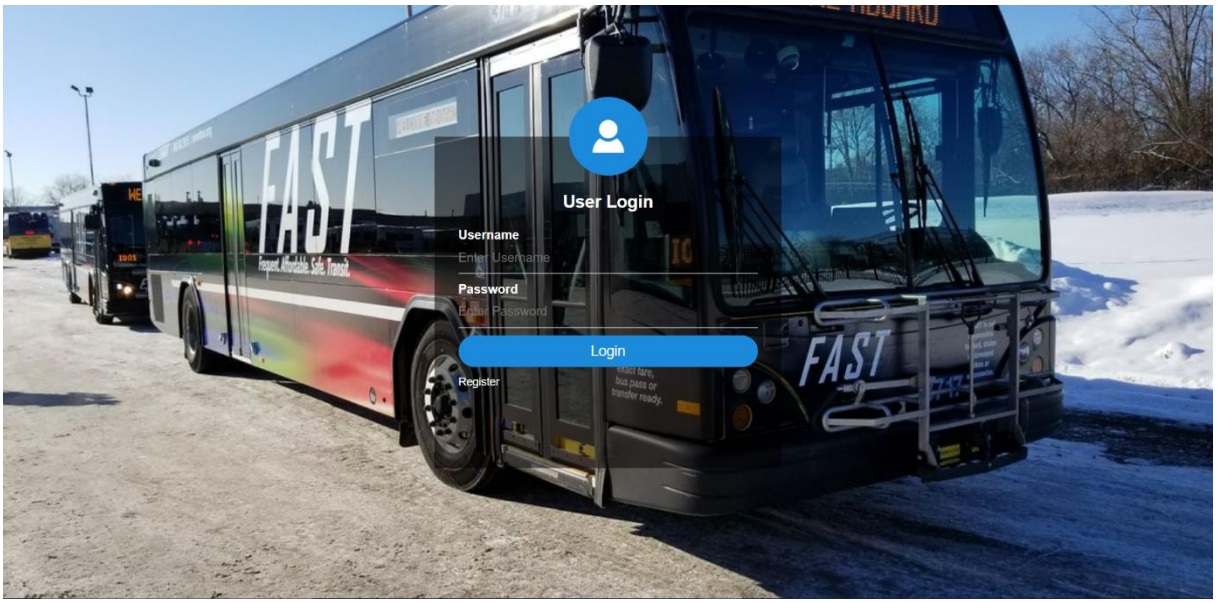

Figure no 7.6 : User login page.

User login takes place from above snapshot

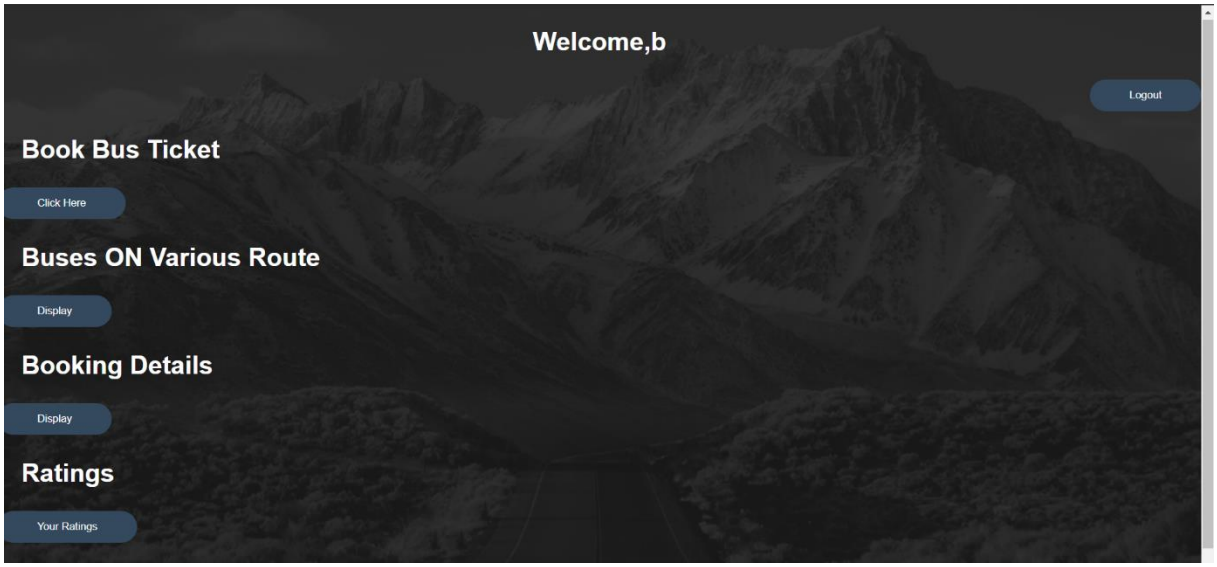

Figure no 7.7 : User menu page

The above snapshot is the display of user menu

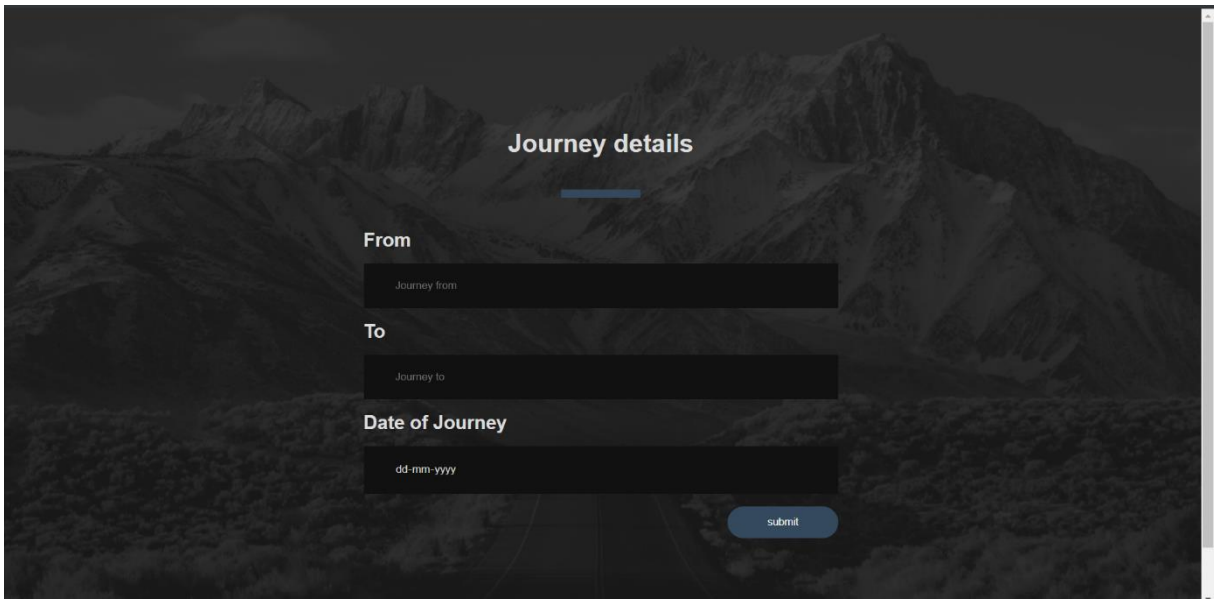

Figure no 7.8 : Journey details page.

The above snapshot is used to fetch the bus details as per the Source and destination for user end

On Route Bus Details

BUS ID	BUS NAME	SOURCE	Source Arrival	Distination	Distination Arrival	Date	Fare	ROOT ID	BOOK Ticket
1	neetha	mumbai	12:11:00	pune	12:44:00	2019-01-19	1000	1	Book Now
10	ksrtc travels	mumbai	12:05:00	pune	12:05:00	2019-01-18	500	1	Book Now
2	Neeta travels	mumbai	12:05:00	pune	12:05:00	2019-01-18	12000	1	Book Now
3	Neeta travels	mumbai	12:05:00	pune	12:05:00	2019-01-18	300	1	Book Now
9	h.t.co travels	mumbai	12:05:00	pune	12:05:00	2019-01-18	500	1	Book Now

Figure no 7.9 : on route bus details

The above snapshot contains the bus details,on which user want's to travel

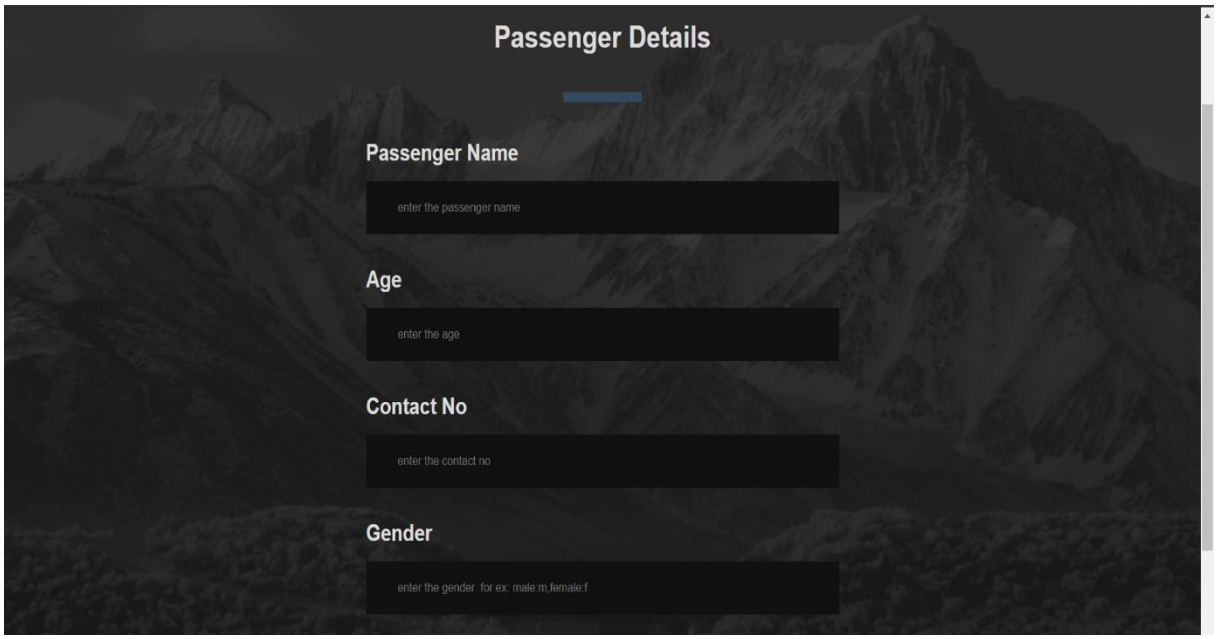

Figure no 7.10 : Passenger details Page.

The above snapshot is used to insert the passenger details

Chapter 8

CONCLUSION

The system is completely menu driven and extremely user friendly since it is developed in an efficient front end tool . Appropriate error messages are also provided too guide the user in a proper and user friendly manner.

The software “Bus Booking System” has been developed in windows 2007 environment using HTML/CSS as front end and JSP as back end. Time consumptions reduced to a great extent and user as less complexity in handling it database.

Searching and retriving of bus records will be easier since there would be a search module that will filter all bus records.

FUTURE ENHANCEMENT AND SCOPE

Further expansion of the system also can be done in future if needed. The application can be enhanced in the future with the needs of the organization. The database and the information can be updated to the latest forthcoming versions.
There are also possibilities for enhancing and further developing the project with customized reports according to the latest information and needs of the company.

The scope of the project is managing a consistency and storage of data by dedicated data administrator. It provides most of the features that a Database Management System should have. It is developed by using MySQL database. It has been implemented in WINDOWS platform.

Features that can be added in future:

Customer Interaction

Customer Reviews

Effective Payment Integration

Google Map Integration

Compelling Photos and Images

BIBLIOGRAPHY

- www.programmingworld.com
- www.vbnetcode.com
- www.w3schools.com
- www.vbcode.com
- www.vbdevelopplus.com
- www.codepark.com

Personal Profile

Prof. Saritha.M received B.E. degree in Computer Science & Engineering from UBDTCE College in the year 2008 and M.Tech in Computer Science & applications engineering from NMAMIT in 2014.

Interested in internet of things, Published 2 papers in international journal and one national conference paper.

Prof.Saritha.M,Asst.Prof

Project Guide

Name: NISHANTH B.S
USN: 4SU17CS052
Address: Chandragiri Nilaya House
Shirlalu post ,village
Belthangady -574214
Dakshina Kannada, Karnataka
E-mail ID: Nishanthbangera107@gmail.com
Contact Phone No: 9741963630