**INTRODUCTION**

**1. INTRODUCTION**

**1.1 ABOUT PROJECT:**

Any mobile user using this project is aimed at developing an application that incorporates both location and data call technology, steering you towards the very best activities that your city has to offer. For the selected category, the application retrieves appropriate list of options for any location review. The application provides the full information (Places, Routes and Travel information) for any selected option. User also has an option to call the selected option the info from the application.

The City Searchis an online guide that provides information about Tourist in the categories of dining, entertainment, retail, travel, and professional services in cities. Here the end user would be a Mobile user as it’s an Android technology; so that the user can finally implement all the needed operations in their own mobile device. The Mobile User should be able to search appropriate location. The Mobile User should be able to retrieve the very best activities for the selected city. The user can also be able to retrieve full information (Places, Routes and Travel information) for the any selected option. The Mobile User should also have an option to call the selected option the information from the application.

City Tour Guide is provides the all the information about the cities. It is very useful for the tourism system, in current tourism system, whenever a tourist visits famous spots, to know more about the place the hires a guide. The hired guide then narrates history of the place. The proposed system doesn’t require a physical guide. The mobile application installed on the user mobile then tourist can act as a guide. This application contains the information about famous places like temples, parks, beaches and etc. The android based city guide system has more practical significance.

**Modules:**

* Administration Module
* Place Detection
* Routing Module
* Travelling Module
* Rating Module

**Module Description:**

The project Online City Tour Guide is developed for the purpose of providing better information for tourists system. The reason behind this development is to solve the problem facing by the tourists.

**This application provides:**

* Users are search and selected the one city; the city information is in a single application.
* User selected city contains the all the information about the selected city.
* The application provides to trace the famous place by using the GPS system.
* This application provides the needs for the tourists, tourist needs like hotels, hospitals, resturants, hotels, banks, shoppingmalls address etc..
* The main advantage for users is to search the different cities by using the search option. Users don’t have any idea about the famous cities; this search option is very useful for users.
* In this application to search any city information by using the search option.
* Users use this application, doesn’t require a physical guide.
* Major city details in a single application.

This project implements its task with the help of the functionality of the following modules.

**Administration Module:**

Administrator maintains the all the activities in this system. Adding new states and cities and changing the information if there are any new changes occurs. New changes are occurred those are maintained by the administrator.

Admin will have login form that can enter user name and password to view other features. After login is completed admin can Adding new places and deleting old places etc. In this module administration has done the following job like maintaining all the information about the cities. The admin has overall rights over the system and can modify, delete any details. Admin can see the user ratings.

**Place Detection:**

Place Detection Module provides the information about the places in the cities. User (tourist) can search the famous place in this module. The user should able to search appropriate locations. The user should be able to retrieve the very best activities for the selected city. The user can also be able to retrieve full information for any selected option. If don’t have an idea about the place then user can search the places in a cities. Users have to search the states option first after that he/she selects the one city. Selected city information is displayed on the screen. And also this module provides famous visiting places near to the selected location and also provides details about hotels, hospitals, shopping malls and so on.

**Routing Module:**

Routing Module provides the information about the routes. The user must be search the route to reach location spots. How to reach a place to use GPS system. The user should also have the GPS system on user mobile. User selected the one famous place to reach that place using the route module. The user can also able to retrieve the full information for the any selected city routes. All the Cities route information is available in this module. This module provides route information to the selected place.

**Traveling Module:**

Traveling Module provides the information about the available vehicles present in the user selected city. This module provides the vehicular information, travels information, distance information, time and cost information to reach a place, sites those are used to select the vehicle and so on.

**Rating module:**

After using this application user will gives a rating. One user gives many times.

**1.2 ABOUT SOFTWARE:**

**ABOUT JAVA:**

Initially the language was called as “oak” but it was renamed as “Java” in 1995. The primary motivation of this language was the need for a platform-independent (i.e., architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

* Java is a programmer’s language.
* Java is cohesive and consistent.
* Except for those constraints imposed by the Internet environment, Java gives the programmer, full control.
* Finally, Java is to Internet programming where C was to system programming.

**Importance of Java to the Internet:**

Java has had a profound effect on the Internet. This is because; Java expands the Universe of objects that can move about freely in Cyberspace. In a network, two categories of objects are transmitted between the Server and the Personal computer. They are Passive information and Dynamic active programs. The Dynamic, Self-executing programs cause serious problems in the areas of Security and probability. But, Java addresses those concerns and by doing so, has opened the door to an exciting new form of program called the Applet.

Java can be used to create two types of programs

**Applications and Applets:**

An application is a program that runs on our Computer under the operating system of that computer. It is more or less like one creating using C or C++. Java’s ability to create Applets makes it important. An Applet is an application designed to be transmitted over the Internet and executed by a Java –compatible web browser.

An applet is actually a tiny Java program, dynamically downloaded across the network, just like an image. But the difference is, it is an intelligent program, not just a media file. It can react to the user input and dynamically change.

**Java Virtual Machine (JVM):**

Beyond the language, there is the Java virtual machine. The Java virtual machine is an important element of the Java technology. The virtual machine can be embedded within a web browser or an operating system.

Once a piece of Java code is loaded onto a machine, it is verified. As part of the loading process, a class loader is invoked and does byte code verification makes sure that the code that’s has been generated by the compiler will not corrupt the machine that it’s loaded on.

Byte code verification takes place at the end of the compilation process to make sure that is all accurate and correct. So byte code verification is integral to the compiling and executing of Java code.

**Overall Description:**

# Java Source

## Java byte code

# JVM

**Java**

**.Class**

Picture showing the development process of JAVA Program

Java programming uses to produce byte codes and executes them. The first box indicates that the Java source code is located in a. Java file that is processed with a Java compiler called javac. The Java compiler produces a file called a. class file, which contains the byte code. The .Class file is then loaded across the network or loaded locally on your machine into the execution environment is the Java virtual machine, which interprets and executes the byte code.

**AN OVERVIEW OF JSP:**

**Java Server Pages (JSP):**

Java server Pages is a simple, yet powerful technology for creating and maintaining dynamic-content web pages. Based on the Java programming language, Java Server Pages offers proven portability, open standards, and mature re-usable component model. The Java Server Pages architecture enables the separation of content generation from content presentation.

This separation not eases maintenance headaches; it also allows web team members to focus on their areas of expertise. Now, web page designer can concentrate on layout, and web application designers on programming, with minimal concern about impacting each other’s work.

**Features of JSP:**

**Portability:**

Java Server Pages files can be run on any web server or web-enabled application server that provides support for them. Dubbed the JSP engine, this support involves recognition, translation, and management of the Java Server Page lifecycle and its interaction components.

**Components:**

It was mentioned earlier that the Java Server Pages architecture can include reusable Java components. The architecture also allows for the embedding of a scripting language directly into the Java Server Pages file. The components current supported include Java Beans, and Servlets.

**Processing:**

A Java Server Pages file is essentially an HTML document with JSP scripting or tags. The Java Server Pages file has a JSP extension to the server as a Java Server Pages file. Before the page is served, the Java Server Pages syntax is parsed and processed into a Servlet on the Server side. The Servlet that is generated outputs real content in straight HTML for responding to the client.

**Access Models:**

A Java Server Pages file may be accessed in at least two different ways. A client’s request comes directly into a Java Server Page. In this scenario, suppose the page accesses reusable Java Bean components that perform particular well-defined computations like accessing a database. The result of the Beans computations, called result sets is stored within the Bean as properties. The page uses such Beans to generate dynamic content and present it back to the client. In both of the above cases, the page could also contain any valid Java code. Java Server Pages architecture encourages separation of content from presentation.

**Steps in the execution of a JSP Application:**

* The client sends a request to the web server for a JSP file by giving the name of the JSP file within the form tag of a HTML page.
* This request is transferred to the JavaWebServer. At the server side JavaWebServer receives the request and if it is a request for a jsp file server gives this request to the JSP engine.
* JSP engine is program which can understand the tags of the jsp and then it converts those tags into a Servlet program and it is stored at the server side. This Servlet is loaded in the memory and then it is executed and the result is given back to the JavaWebServer and then it is transferred back to the result is given back to the JavaWebServer and then it is transferred back to the client.
* The Java Server Pages technology provides a textual description for the creation of a response from a request.

The technology builds on the following concepts:

**Template Data:**

A substantial portion of dynamic content is actually fixed. The JSP technology allow for the natural manipulation of this data.

**Addition of Dynamic Data:**

The JSP technology allows the addition of dynamic data to the template data in a way that is simple yet powerful.

**Encapsulation of Functionality:**

The JSP technology provides two related mechanisms for the encapsulation of functionality: the standard Java Beans component architecture and the tag library mechanism.

**Good Tool Support:**

The JSP technology has features that enable the creation of good authoring tools. The result is a flexible and powerful server-side technology.

**Benefits of the Java Server Pages Technology:**

The Java Server Pages technology offers a number of benefits:

**Write Once, Run Anywhere™ properties”:**

The Java Server Pages technology is platform independent, both in its dynamic Web pages, Web servers, and its underlying server components.

You can author JSP pages on any platform, run them on any Web server or Web enabled application server, and access them from any Web browser

**High quality tool support:**

The Write Once, Run Anywhere properties of JSP allows the user to choose best-of-breedtools. Additionally, an explicit goal of the Java Server Pages design is to enable the creation of high quality portable tools.

**Separation of Roles:**

JSP supports the separation of roles: developers write components that interact with server-side objects.

**Reuse of components and tag libraries:**

The Java Server Pages technology emphasizes the use of reusable components such as Java Beans™ components, Enterprise Java Beans™ components and tag libraries.

**Separation of dynamic and static content:**

The Java Server Pages technology enables the separation of static content from dynamic content that is inserted into the static template.

**Support for scripting and actions:**

The Java Server Pages technology supports scripting elements as well as actions. Actions permit the encapsulationof useful functionality in a convenient form that can also be manipulated by tools; scripts provide a mechanism to glue togetherthis functionality in a per-page manner.

**Web access layer for N-tier enterprise application architecture(s):**

The Java Server Pages technology is an integral part of the Java 2 Platform Enterprise Edition (J2EE), which brings Java technology to enterprise computing.

**JAVA DATABASE CONNECTIVITY:**

**What Is JDBC?**

JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for Java Database Connectivity.

It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. The combinations of Java and JDBC lets a programmer write it once and run it anywhere.

**What Does JDBC Do?**

Simply put, JDBC makes it possible to do three things:

* Establish a connection with a database
* Send SQL statements, Process the results.

**JDBC DRIVER TYPES:**

The JDBC drivers that we are aware of at this time fit into one of four categories:

* JDBC-ODBC Bridge plus ODBC driver.
* Native-API partly-Java driver.
* JDBC-Net pure Java driver.
* Native-protocol pure Java driver.

**JDBC-ODBC Bridge:**

If possible, use a Pure Java JDBC driver instead of the Bridge and an ODBC driver. This completely eliminates the client configuration required by ODBC. It also eliminates the potential that the Java VM could be corrupted by an error in the native code brought in by the Bridge (that is, the Bridge native library, the ODBC driver manager library, the ODBC driver library, and the database client library).

**What Is the JDBC- ODBC Bridge?**

The JDBC-ODBC Bridge is a JDBC driver, which implements JDBC operations by translating them into ODBC operations. To ODBC it appears as a normal application program. The Bridge implements JDBC for any database for which an ODBC driver is available. The Bridge is implemented as the Sun.jdbc.odbc Java package and contains a native library used to access ODBC. The Bridge is a joint development of Innersole and Java Soft.

**JDBC connectivity:**

The JDBC provides database-independent connectivity between the J2EE platform and a wide range of tabular data sources. JDBC technology allows an

Application Component Provider to:

* Perform connection and authentication to a database server.
* Manager transactions.
* Move SQL statements to a database engine for preprocessing and execution.
* Execute stored procedures.
* Inspect and modify the results from Select statements.

**JSP, JDBC JAVASCRIPT:**

JavaScript is a script-based programming language that was developed by Netscape Communication Corporation. JavaScript was originally called Live Script and renamed as JavaScript to indicate its relationship with Java. JavaScript supports the development of both client and server components of Web-based applications.

On the client side, it can be used to write programs that are executed by a Web browser within the context of a Web page. On the server side, it can be used to write Web server programs that can process information submitted by a Web browser and then update the browser’s display accordingly.

Even though JavaScript supports both client and server Web programming, we prefer JavaScript at Client side programming since most of the browsers supports it. JavaScript is almost as easy to learn as HTML, and JavaScript statements can be included in HTML documents by enclosing the statements between a pair of scripting tags.

<SCRIPTS>....</SCRIPT>

<SCRIPT LANGUAGE = “JavaScript”>

//JavaScript statements

</SCRIPT>

Here are a few things we can do with JavaScript:

* Validate the contents of a form and make calculations.
* Add scrolling or changing messages to the Browser’s status line.
* Animate images or rotate images that change when we move the mouse over them.
* Detect the browser in use and display different content for different browsers.
* Detect installed plug-ins and notify the user if a plug-in is required.
* We can do much more with JavaScript, including creating entire application.

**Advantages:**

* JavaScript can be used for Sever-side and Client-side scripting.
* It is more flexible than VBScript.
* JavaScript is the default scripting languages at Client-side since all the browsers supports it.

**Hyper Text Markup Language (HTML):**

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows users to produces Web pages that include text, graphics and pointer to other Web pages (Hyperlinks).HTML is not a programming language but it is an application of ISO Standard 8879, SGML (Standard Generalized Markup Language), but specialized to hypertext and adapted to the Web. The idea behind Hypertext is that instead of reading text in rigid linear structure, we can easily jump from one point to another point. We can navigate through the information based on our interest and preference.

A markup language is simply a series of elements, each delimited with special characters that define how text or other items enclosed within the elements should be displayed HTML tags are not case-sensitive. Using graphics, fonts, different sizes, color, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

**Basic HTML Tags:**

**<! -- -->** specifies comments

**<A>………. </A>** Creates hypertext links

**<B>………. </B>** Formats text as bold

**<BODY>…</BODY>** Contains all tags and text in the HTML document

**<CENTER>...</CENTER>** Creates text

**<DL>...</DL>**  Creates definition list

**<FONT>…</FONT>** Formats text with a particular font

**<FORM>...</FORM>** Encloses a fill-out form

**<H#>…</H#>** Creates headings of different levels

**<HEAD>...</HEAD>** Contains tags that specify information about a document

**<HR>...</HR>** Creates a horizontal rule

**<HTML>…</HTML>**  Contains all other HTML tags

**<META>...</META>** Provides meta-information about a document

**<SCRIPT>…</SCRIPT>** Contains client-side or server-side script

**<TABLE>…</TABLE>** Creates a table

**<TR>…</TR>** Designates a table row

**<TH>…</TH>** Creates a heading in a table

**Advantages:**

* A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.
* HTML is platform independent, HTML tags are not case-sensitive.

**Role of Oracle in Database:**

ORACLE 8i is one of the many database services that plug into a client / server model. It works efficiently to manage resources, a database information, among the multiple clients requesting & sending.

**Structured Query Language (SQL):**

SQL is an inter-active language used to query the database and access data in database. SQL has the following features:

* It is a unified language.
* It is a common language for relational database.
* It is a non-procedural language.

**Introduction to Oracle:**

Oracle is a comprehensive operating environment that packs the power of a mainframe system into user microcomputer. It provides a set of functional programs that user can use as tools to build structures and perform tasks.

Because application developed on oracle are completely portable to environment and then move it into a multi user platform. Users do not have to be an expert to appreciate oracle, but the better user understands the programmer, the more productivity and creativity you will use the tools it provides.

**What is a Relational Database Management System?**

A relational database management system (RDBMS) can perform a wide array of tasks. It acts as a transparent interface between the physical storage and a logical presentation of data. It provides a set of more or less flexible and sophisticates tools for handling information. User can use this tool to:

* Define a database, Query the database.
* Add, edit and delete data, Modify the structure of database.
* Secure data from public access.
* Communicate within the networks.
* Export and Import data.

Because it gives so much control over data, a relational DBMS can also save as the foundation for products that generate application and extract data.

A Database Management system may be called fully relational if it supports:

* Relational Databases and
* A language that is at least as powerful as the relational algebra.

Sometimes all the information of interest to a business operation can stored in one table. SQL Server makes it very easy to link the data in multiple tables. Matching an employee to the department in which they work is one example.

**Referential Integrity:**

Not only does SQL Server allow you to link multiple tables, it also maintains consistency between them. Ensuring that the data among related tables is correctly matched is referred to as maintaining referential integrity.

**Data Abstraction:**

A major purpose of a database system is to provide users with an abstract view of the data. This system hides certain details of how the data is stored and maintained. Data are actually stored.

**Physical Level:**

This is the lowest level of abstraction at which one describes how the data are actually stored.

**Conceptual Level:**

At this level of database abstraction all the attributed and what data are actually stored is described and entries and relationship among them.

**View Level:**

This is the highest level of abstraction at which one describes only part of the database.

**Advantages of RDBMS:**

* Redundancy can be avoided, Inconsistency can be eliminated.
* Data can be shared.
* Standards can be enforced.
* Security restrictions can be applied.

**Disadvantages of DBMS:**

A significant disadvantage of the DBMS system is cost. In addition to the cost of purchasing of developing the software, the hardware is to be upgraded to allow for the extensive programs and the workspace required for their execution and storage. While centralization reduces duplication, the lack of duplication requires that the database be adequately backed up so that in case of failure the data can be recovered.

**ANDROID:**

**What is Android?**

Android is an open source and Linux-based Operating System for mobile devices such as smart phones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies. Android offers a unified approach to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android.

Android applications are usually developed in the Java language using the Android Software Development Kit. Once developed, Android applications can be packaged easily and sold out either through a store such as Google Play or the Amazon App store Android powers hundreds of millions of mobile devices in more than 190 countries around the world. It's the largest installed base of any mobile platform and growing fast. Every day more than 1 million new Android devices are activated worldwide. You will be glad to know that you can start your Android application development on either of the following.

**Operating systems:**

* Microsoft Windows XP or later version.
* Mac OS X 10.5.8 or later version with Intel chip.
* Linux including GNU C Library 2.7 or later.

Second point is that all the required tools to develop Android applications are freely available and can be downloaded from the Web. Following is the list of software's you will need before you start your Android application programming.

* Java JDK5 or JDK6, Android SDK.
* Eclipse IDE for Java Developers (optional).
* Android Development Tools (ADT) Eclipse Plug-in (optional).

**ANDROID VERSIONS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code Name** | **Version Number** | **Initial Release Date** | **API Level** | **Support Status** |
| Android (Alpha) | 1.0 | September 23, 2008 | 1 | Unsupported |
| Beta (known as "[Petit Four](https://en.wikipedia.org/wiki/Petit_four)" internally) | 1.1 | February 9, 2009 | 2 | Unsupported |
| [Cupcake](https://en.wikipedia.org/wiki/Android_Cupcake) | 1.5 | April 27, 2009 | 3 | Unsupported |
| [Donut](https://en.wikipedia.org/wiki/Android_Donut) | 1.6 | September 15, 2009 | 4 | Unsupported |
| [Eclair](https://en.wikipedia.org/wiki/Android_Eclair) | 2.0 – 2.1 | October 26, 2009 | 5 – 7 | Unsupported |
| [Froyo](https://en.wikipedia.org/wiki/Android_Froyo) | 2.2 – 2.2.3 | May 20, 2010 | 8 | Unsupported |
| [Gingerbread](https://en.wikipedia.org/wiki/Android_Gingerbread) | 2.3 – 2.3.7 | December 6, 2010 | 9 – 10 | Unsupported |
| [Honeycomb](https://en.wikipedia.org/wiki/Android_Honeycomb) | 3.0 – 3.2.6 | February 22, 2011 | 11 – 13 | Unsupported |
| [Ice Cream Sandwich](https://en.wikipedia.org/wiki/Android_Ice_Cream_Sandwich) | 4.0 – 4.0.4 | October 18, 2011 | 14 – 15 | Unsupported |
| [Jelly Bean](https://en.wikipedia.org/wiki/Android_Jelly_Bean) | 4.1 – 4.3.1 | July 9, 2012 | 16 – 18 | Unsupported |
| [Kit Kat](https://en.wikipedia.org/wiki/Android_KitKat) | 4.4 – 4.4.4 | October 31, 2013 | 19 | Unsupported |
| [Lollipop](https://en.wikipedia.org/wiki/Android_Lollipop) | 5.0 – 5.1.1 | November 12, 2014 | 21 – 22 | Unsupported |
| [Marshmallow](https://en.wikipedia.org/wiki/Android_Marshmallow) | 6.0 – 6.0.1 | October 5, 2015 | 23 | Supported |
| [Nougat](https://en.wikipedia.org/wiki/Android_Nougat) | 7.0 – 7.1.1 | August 22, 2016 | 24 – 25 | Supported |

**SYSTEM ANALYSIS**

**2. SYSTEM ANALYSIS**

System analysis is the most important part of any project study. This phase, system analysis gives us a perfect idea about our system. It also provides us with some additional information. It describes about the entire system, not only about the present system but also the overview of the present system, limitation of the present system and the proposed solution. So, the system study and analysis are the most important parts of a system study progress. In order to tourist can access the all the information and tourist act as a guide.The challenge address by mobile was ability to getexact location from the specified favorites, location. It is fast, easy and reliable commercial means for the all the famous tourist places information. This proposed project is simply an attempt to tourist can easily access the all the information about the different famous places.

**2.1 Existing System:**

In the existing system the details of the metro for different cities is available in different applications. Existing system are individual application for the particular city such as “Delhi Metro Navigator”. The user has to go around searching for the application that contains the details of the required city. It is time consuming to search the details in this manner. Similarly details such as nearby hotels and hospitals are not available in a single application.

**Disadvantages:**

* Individual application for particular city.
* Full version of application is paid one.
* Needs internet connection for its accessing.
* Finding the nearest hotel and hospital not possible.

**2.2 Proposed System:**

The proposed system entitled as "City Tour Guide" is an online application". In the proposed system the details of the cities is shown in a single application. It provides the famous location information of cities, and other details. The nearest hotels and the nearest hospitals for the particular place are shown to make the journey of the user easy. It displays the Locations as per user's choice.

In this proposed system user can act as a guide. Users can access the information through on database. The software can record data in the database, display Cities information; Famous Places information, and many more.

**ADVANTAGES OF PROPOSED SYSTEM:**

* Major city details in a single application.
* Complete online application.
* Open source.
* Searching facility.
* Doesn’t require a physical guide.

**2.3 HARDWARE, SOFTWARE SPECIFICATIONS:**

**HARDWARE REQUIREMENTS:**

* **Processor Type :** Pentium IV
* **Memory (RAM) :** 4GB
* **Hard Disk :** 500GB
* **Drive :** CD-ROM Drive
* **Display :** 32-bit color Monitor
* **Modem Speed :** 36KBPS

**SOFTWARE REQUIREMENTS:**

* **Operating system :**Microsoft Windows XP / Win7/win 8.
* **Platform of development :**Android studio
* **Database :**Oracle10gexpressionEdition
* **Web Browser :** Windows Internet Explorer 8.0
* **Front end :** Android
* **Back end :** My SQL,JSP
* **Web Server :** Apache Tomcat 8.

**2.4 SRS (Software Requirement Specification):**

A software requirements specification(SRS),a requirements specification for a software system, is a complete description of the behavior of a system to be developed and may include set of use cases that describe interactions the users will have with the software .in addition it also contains non-functional requirements. Non-functional requirements impose constraints on the design or implementation. The requirement specification phase consists of two basic activities:

* Problem analysis for “City Tour Guide“.
* Requirement specification for “City Tour Guide “.

A software requirement specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase.

**Problem analysis for “City Tour Guide”:**

The main aim of this City Tour Guide project is to provide information about the famous tourist places in the cities. In the existing system the guide can show the all the famous tourist places in a one city.

The main disadvantage of this process is mentioned below….

* The only source of information is provided manually.
* The tourists have no idea about the all the information like famous places, how to reach a places and how much money will paid about the expense. This process may takes lot of time lot of money wastages.

In the Proposed system, the development of new system will overcome all the problems of the existing system. The Advantages of the Present System are

* Our application is simple, and smart. Our application is quite useful which helps for Tourists.
* It provides the all the information about the city in a single application.
* This application provides a platform for tourist to know about the details about their famous places.

**1. Introduction:**

This page contains SRS documentation for Online City Tour Guide Application. The SRS is easy understood of the analysis task. The function and performance allocated to software as part of the system engineering and refined by establishing a complete information description, a detailed functional description, a representation of system behavior, indication of performance requirements and design constrains, appropriate validation criteria and the other information related to requirements. This document gives detailed functional and nonfunctional requirements for Online City Tour Guide Application. The SRS is technical specification of requirement of Online City Tour Guide Application. This specification describes what the proposed system should do without describing how it will do it. It also describes complete external behavior of proposed system. This document gives detailed functional and nonfunctional requirements for Online City Tour Guide Application.

**1.1 Purpose:**

The City Tour Guide Application Search is an online Application that provides information about cities in the categories of famous tourist places, locations, travel, and professional services in cities. The purpose of the is to facilitate accessing and handling of detailed information of the Cities.

The purpose of the document is to make the functional and non-functional requirements of the Online City Tour Guide Application easy to comprehend. It also serves the purpose of making the functionality clear to end users. The main objective of this product is to automate the manual processing of the tour movement. This product will allow the users to keep track of the cumulative bagging and cumulative dispatch at the end of every day. This product also allows the users can search the needs of their user choice. This information can be easily maintained on database.

**1.2 Scope:**

This website is to display a table of information about the cities. User can select whatever shows of interest, and upon request a customized list of favorite shows is created as a personal guide to the festival.

The Mobile User should be able to search appropriate location. The Mobile User should be able to retrieve the very best activities for the selected city. The user can also be able to retrieve full information for the any selected option.

**Developer Responsibilities**:

1. Installing the software on the client’s hardware, and configuring the software

according to client requirements.

1. Conducting any user training that might be needed for using this system.
2. Maintaining the system for a period of time as per the agreement to a particular

Organization.

**Definitions, Acronyms and Abbreviations:**

* TRGD - Tour Guide.
* GPS - Global Positioning System
* ISO - International Standards Organization.
* DB - Database.
* SRS - Software Requirement Specification.
* JSP - Java Server Page.
* J2SE - Java 2 Standard Edition
* SDK - Software Development K
* HTML - Hyper Text Markup Language.

**1.3 Document Conventions:**

This describes any standards or typographical conventions that were writing this SRS, such as fonts or highlighting that have special significance. In this preparation of document there are three kinds of font-sizes namely 16, 14, 12.The font style used in Times New Roman. The main headings are of size 16 and subheadings are of size 14 and inner headings are of 12, and are in bold**.**

**2. Overall Description:**

**2.1 Overview:**

The Online City Tour Guide Application maintains the information about cities like Places, Locations, Routes, and Travels etc. City search is an online guide that provides information user can select whatever shows of interest.

City Tour Guide contains the complete information about the particular city like places, routes and travels information about the organization that provides history of the city. It gives us the valuable information about the city and saves the time. This project android city tour guide provides the tourist with city map depending on its current location entered by the android phone user.

This information helps the tourists to find the desired locations to visit. Well it consists of entire details of those locations or how to reach the place as well as needs like hotels, hospitals, restaurants, shopping malls etc but it provides the basic information to decide the place to visit. This project is mainly beneficial for the tourists having no idea about the places they want to visit. By providing geographic based information system the tourist’s and people shifting to new cities can get a better guidance of the places they want to visit.

The android based city tour guide system can realize to query information for hotels, restaurants, hospitals and so on. The android based city tour guide system has more practical significance. In the current scenario there isn’t any application that would help a tourist to get information about the place they are currently visiting in their mobile phone. Our application android city tour guide system based on web service is aimed to solve this problem.

In current tourism system, whenever a tourist visits famous spots, to know more about the place he hires a guide. The hired guide then narrates history of the place. The proposed system does not require a physical guide. The mobile application installed on the mobile of tourist can act as a guide. Without having a guide it will help one to get information of the place in their mobile and check out the place which they are currently visiting. Our objective is to utilize an android city tour guide to extract information about a place. We use the GPS(for getting the location) and GPRS(for internet connectivity between mobile and server)features of the android phone. Using web service, user can query information about the places he/she visiting.

**2.2 User Characteristics:**

The administrator of this system **“**Online City Tour Guide Application” will know about this system that how to maintain the master’s data, how to maintain data, how to insert the data, how to prepare information reports.

In all aspects that are described above, we have to give training to the administrator who will be work on this system.

**2.3 General Constraints:**

This system should run on Intel Pentium running on Microsoft-Windows XP

or higher versions.

**3. Requirements:**

**3.1 Functional Requirements:**

This section contains specification of all the functional requirements needed to develop this system.

**Administration Module:**

The administrator is the super user of this application. Admin Logins to the system by giving id and the password. It checks whether the id and the password matches or not. If it matches than Admin page is opened. Otherwise an error message is occurred.

* **Inputs:**

User Logins to the system by giving registration id and the password.

* **Processing:**

After entering the all information in the tables then click the save button then entire information should be stored on the database.

* **Output:**

All the information stored.

**Place Detection Module:**

Place Detection Module Provides the information about the new places for the tourists. Such as famous locations, temples, beaches etc.. & this module provides famous visiting places near to the selected location and also provides details about hotels, hospitals, shopping malls and so on.

* **Input**

User searches the place information.

* **Processing:**

Software searches for entries of all places.

* **Output:**

Users got the all the information about the famous tourist places.

**Routing Module:**

This module provides route information to the selected place.

* **Input:**

User selects their route number.

* **Processing:**

The looks up information about the route

* **Output:**

A map of the route and list of stops is displayed on the screen.

**Traveling Module:**

This module provides the vehicular information, travels information, distance information, time and cost information to reach a place, sites those are used to select the vehicle and so on.

* **Input:**

User selects route number, direction of travel.

* **Processing:**

The system looks up the travel information.

* **Output:**

Automated message user departure times based on input information.

**Rating Module:**

In this module the users gives rating to the application.

* **Input:**

User will give the rating about the application.

* **Processing:**

The system looks up the user rating details.

* **Output:**

The user rating details will be store in database.

**3.2 Design Constraints:**

**Standard Compliance**

The IEEE format is used for the technical documentation specification in this document.

**Hardware Constraints**

The software system runs on Windows98 and above that requires Network Interface Card (NIC) for connecting two systems.

**Software Constraints (limitations)**

The system shall meet all performance requirements running on the JDK1.5.0 application with Apache Tomcat Server, Remote Method Invocation (RMI) on the hardware specified in.

**3.3 Operational and other Requirements:**

**Database table retention**

The properties file will be kept for storing the relevant details.

**Recovery**

In this system, there is no data loss, if there is any data loss we can recover from that problem.

**System Availability**

This system availability for any users that are if we are using we can uses those facility.

**Performance**

This system will provide high performance, because when we entering data that will automatically copied into level1 servers.

**Capacity**

This System stores the data with unlimited capacity.

**Portability**

This system will move from one place to other place without modifications because here we use java platform hence it is platform independence.

**OBJECT ORIENTED DESIGN**

**3. OBJECT ORIENTED DESIGN**

**UML Diagrams:**

Unified modeling language (UML) is a graphical language for visualizing specifying, constructing and documenting the facts of a software-intensive system. UML provides blue prints for business process, system function, programming language statements, database schemas and reusable components.

**Advantages of UML:**

As the name suggests UNIFIED MODELING LANGUAGE. Modeling has been around for years, not only in software field but also in other trades like civil, mechanical etc. Example in civil engineering drawing the main architecture built of diagram is a model by itself. Modeling makes a complex and huge system to break up in to simple and discrete pieces that can be individually understood. Example simple flow drawing is modeling.

There are two main advantages of modeling:

* **Readability:** Representing your whole architecture in flowchart, class diagrams, ER diagrams etc makes your project more readable. Especially when programmers change jobs handover becomes easier. More the project is not readable more the dependencies.
* **Reusability:** After the system is more readable and broken down into pieces, it becomes easier to identify redundant and similar modules. Thus increasing reusability.

**List of UML Diagrams:**

1. Use case diagrams
2. Class diagrams
3. Sequence diagrams
4. Activity diagrams
5. State chart diagrams
6. Component diagrams
7. Deployment diagrams

**3.1 Use Case Diagram:**

They describe what of a system rather than how the system does it. They are used to identify the primary elements and processes than from the system. The primary elements are termed as actors and processes are called use cases. Use case diagram shows actors and their roles.

**3.2 Class Diagram:**

A class captures the common structure and common behavior of a set of objects. A class is an abstraction of real-world items. When these items exist in the real world, they are instances of the class and referred to as objects. A class diagram is a picture for describing generic description of possible system. Class diagram and object diagram are alternative representations of object models. Class diagrams contain classes and objects when dealing with various kinds of metadata, so the separation is not rigid.Class diagram are more prevalent than object diagrams. Class diagrams contain icon representing classes, interfaces, and their relationships.

**3.3 Sequence Diagram:**

Sequence diagrams can be used to explore the logic of a complex operation, function or procedure. They are called sequential nature is shown via ordering of message. The important accept of a sequence diagram is that it is time ordered. This means that the exact sequence of the interaction between the object is represented step by step. Different objects of the interaction between interact with each other by passing “messages”.

**3.4 Activity Diagram:**

The activities that occur within a use case or within objects behavior typically occur in a sequence. An activity diagram is a special kind of a state chart diagram that shows the flow from activity to activity within a system. They are especially important in modeling the function of a system emphasize the flow of control among objects.

**3.5 State Chart Diagram:**

A state chart diagram shows the state space of a given class, the events that cause a transaction from one state to another and actions that result from a state change.

* The state space of a given class.
* The events that cause a transaction from one state to another.
* The actions that result from a state change each state chart diagram is associated with one class or with a higher-level state diagram.
* A state diagram is a directed graph of states connected by transactions.

**3.6 Deployment Diagram:**

* Deployment diagrams are a set of nodes and their relationships. There nodes are physical entities where the nodes are deployed
* Deployment diagrams are used for visualizing deployment view of a system this is generally used by the deployment team.

**3.7 Component Diagram:**

Component diagram shows components, provided and required interfaces, ports, and relationships between them. This type of diagrams is used in Component-Based Development (CBD) to describe systems with Service-Oriented Architecture (SOA).

Component-based development is based on assumptions that previously constructed components could be reused and that components could be replaced by some other "equivalent" or "conformant" components, if needed.

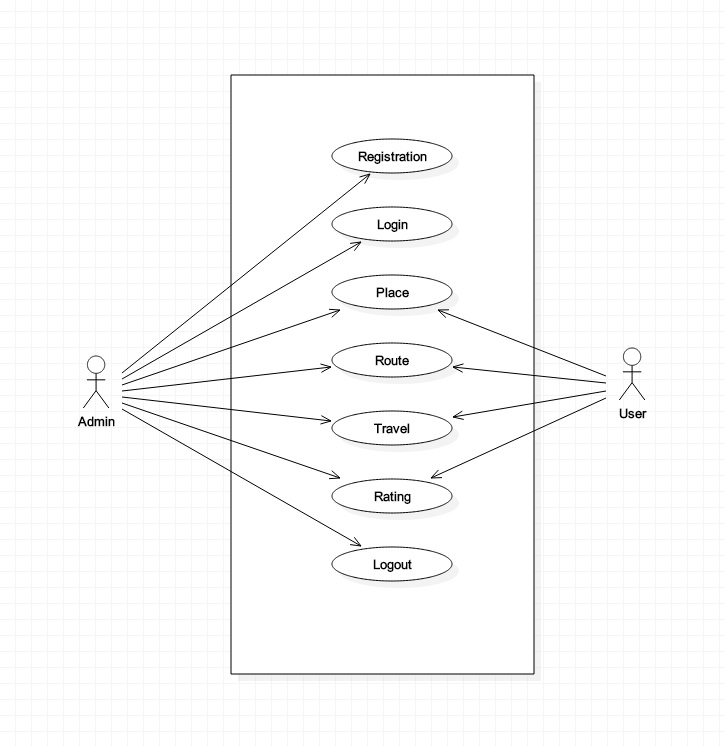
The artifacts that implement component are intended to be capable of being deployed and re-deployed independently, for instance to update an existing system.

Components in UML could represent

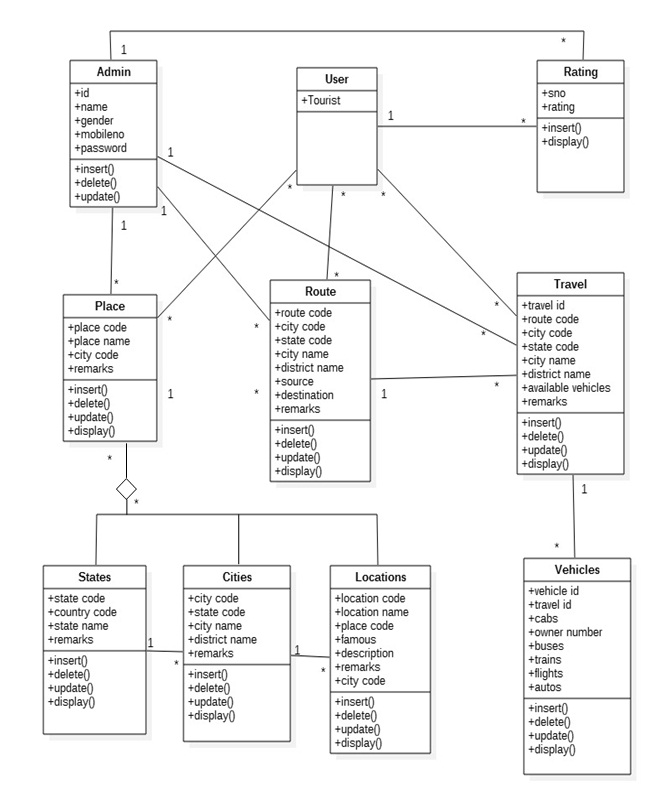
* **Logical components** (e.g., business components, process components), and
* **Physical components** (e.g., CORBA components, EJB components, COM+ and .NET components, WSDL components, etc.).

Along with the artifacts that implement them and the nodes on which they are deployed and executed. It is anticipated that profiles based around components will be developed for specific component technologies and associated hardware and software environments.

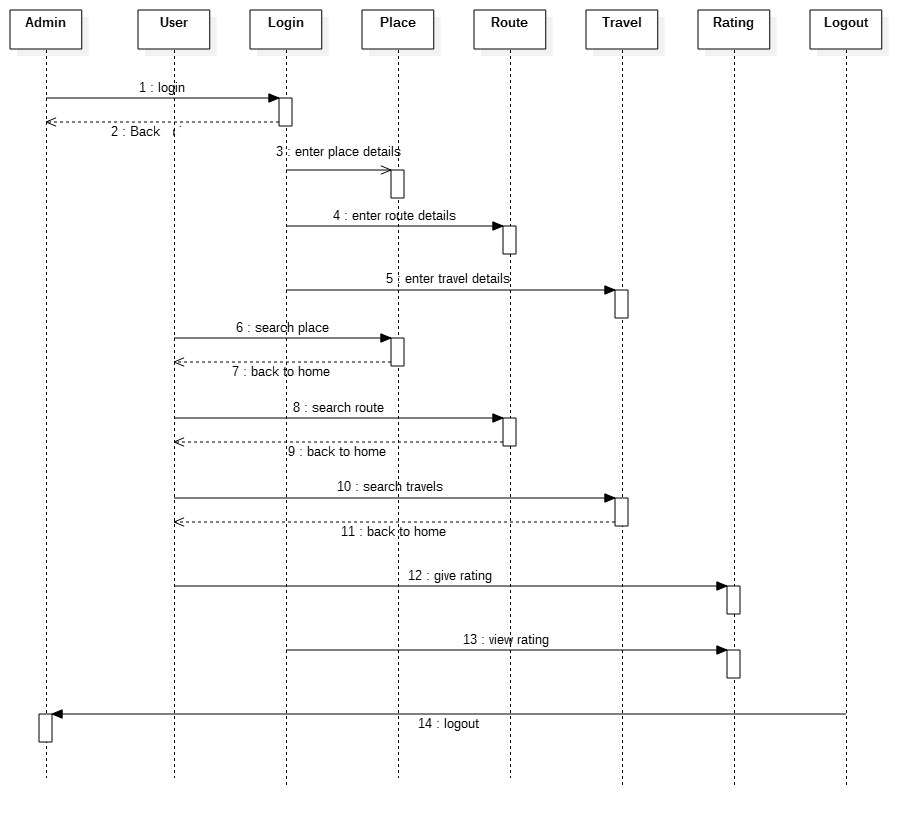
**USE CASE DIAGRAM**

****

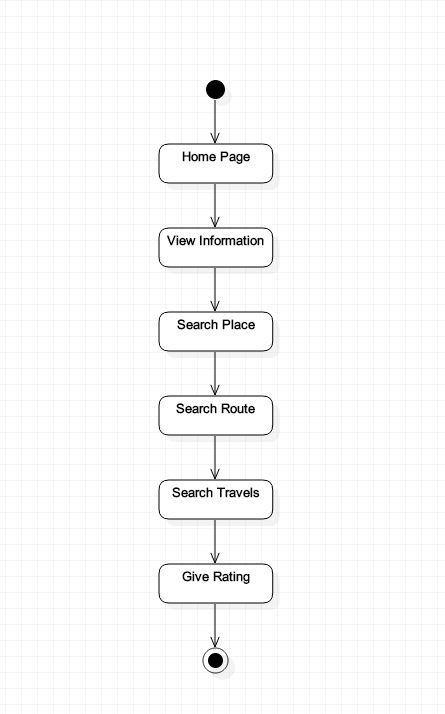
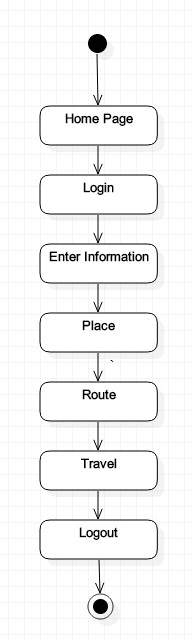
**CLASS DIAGRAM**

****

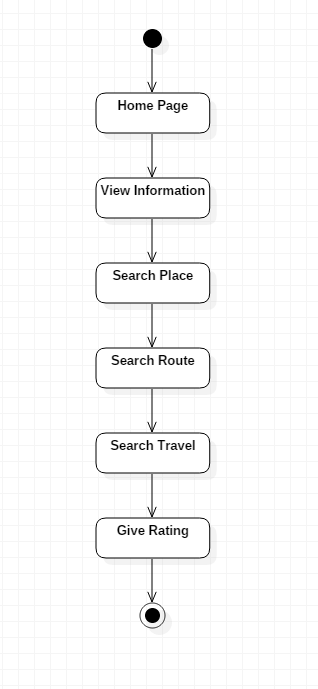
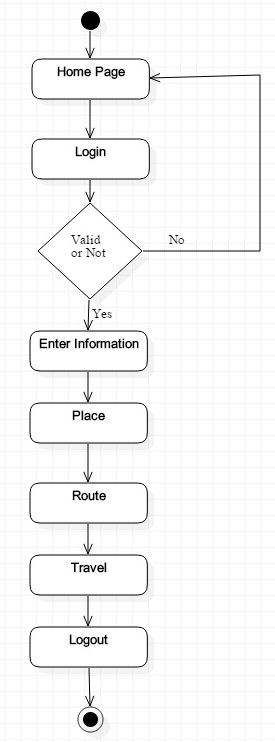
**SEQUENCE DIAGRAM**

****

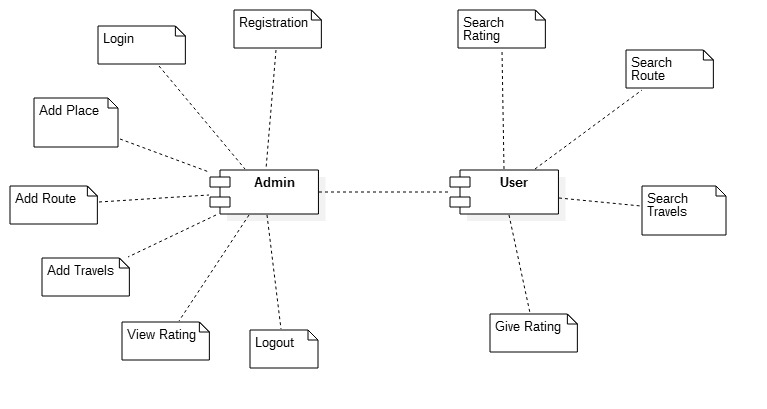
**STATECHART DIAGRAM**

****

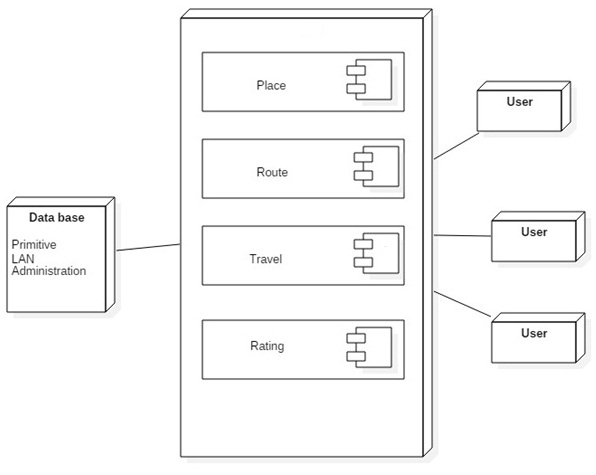
**ACTIVITY DIAGRAM**

****

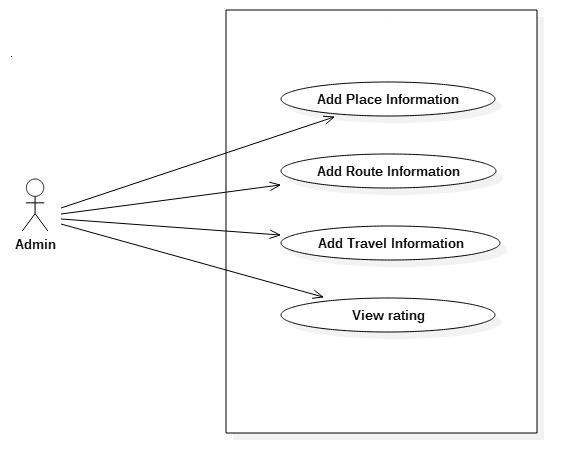
**COMPONENT DIAGRAM**

****

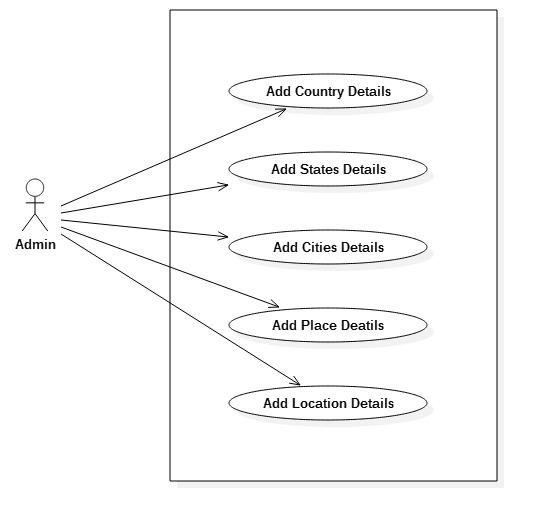
**DEPLOYMENT DIAGRAM**

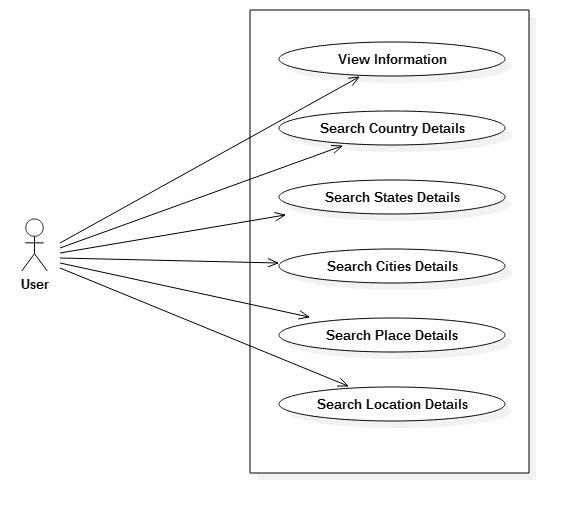
****

**Use case Diagram for Administration Module**

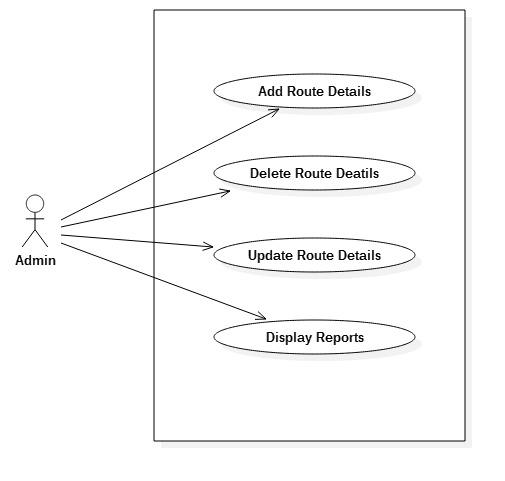
****

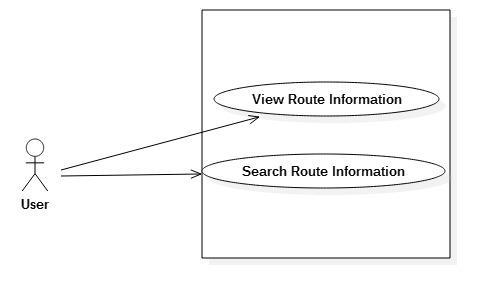
**Use case Diagram for Place Module**

****

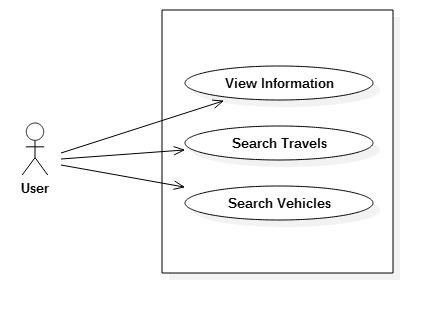
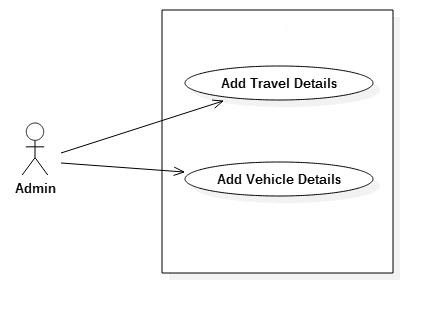
****

**Use case Diagram for Route Module**

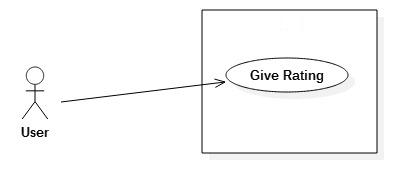
****

****

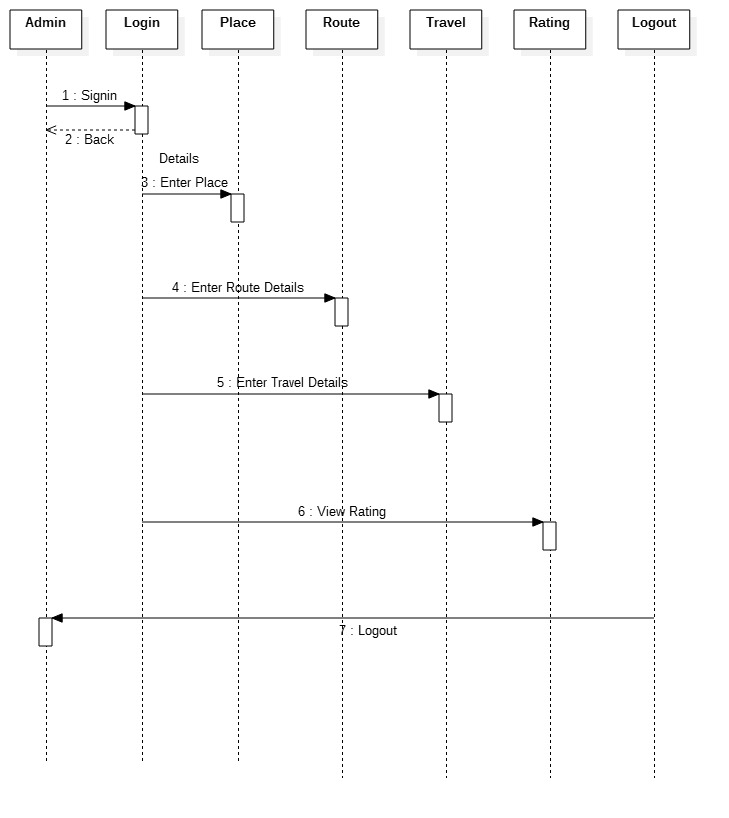
**Use Case Diagram for Travel Module**

****

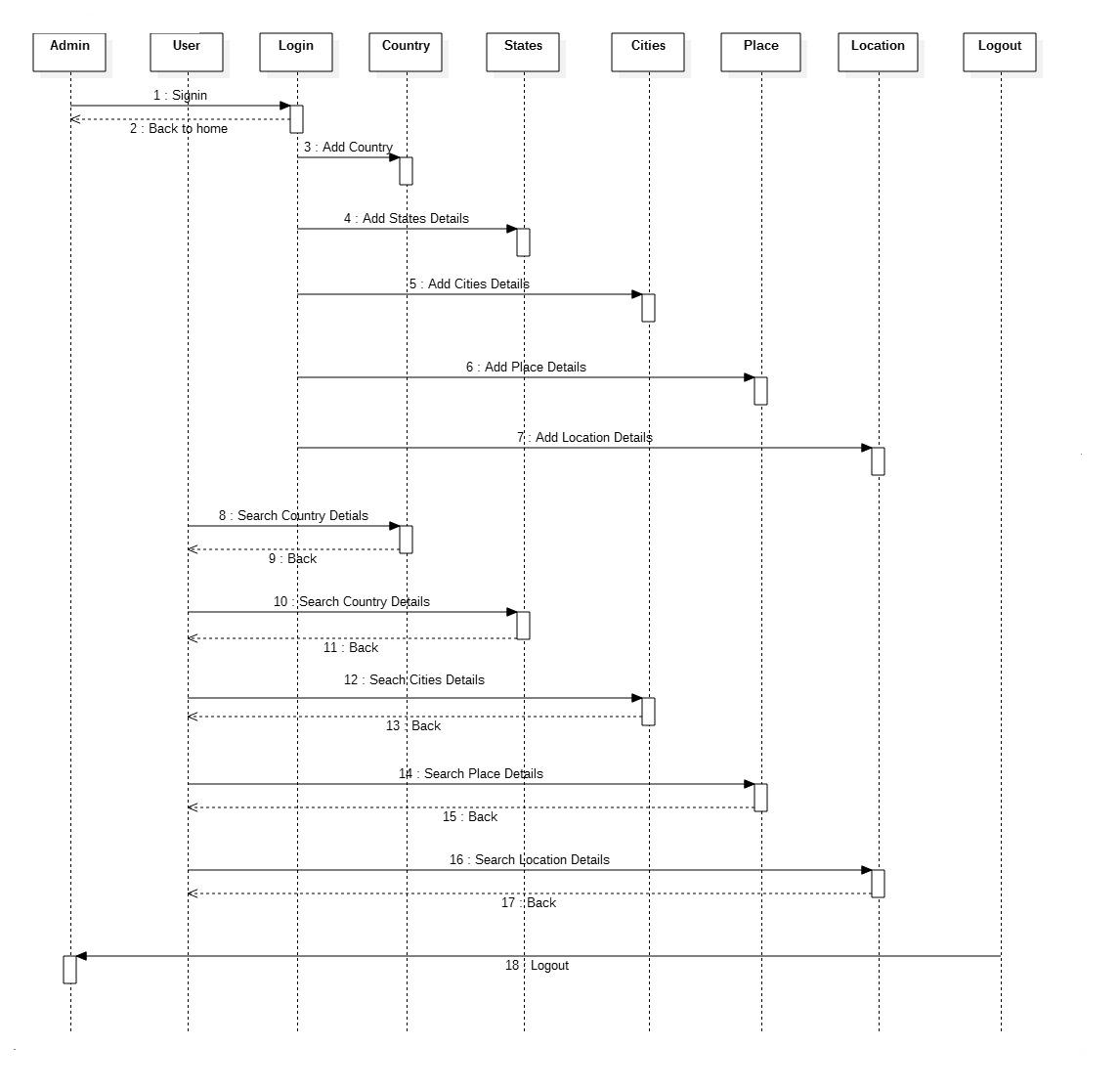
**Use case Diagram for Rating Module**

****

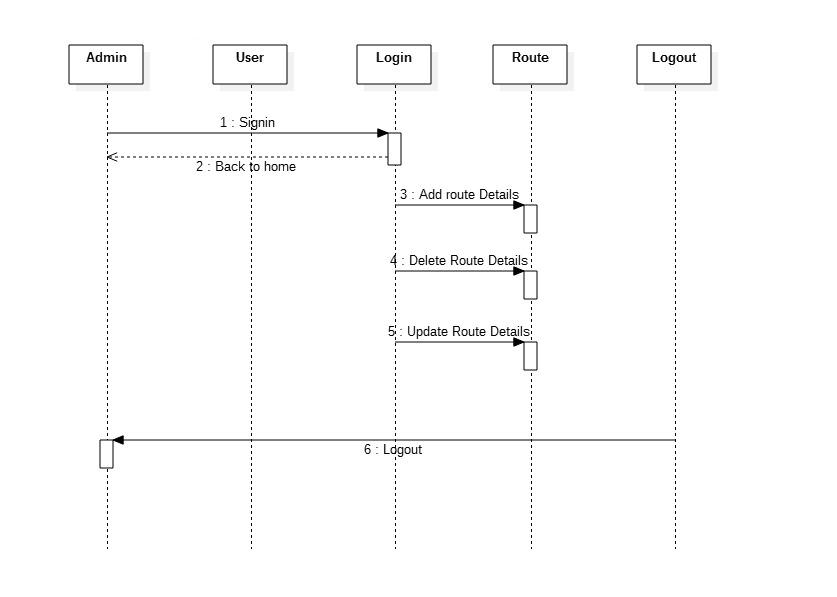
**Sequence Diagram for Administration Module**

****

**Sequence Diagram for Place Module**

****

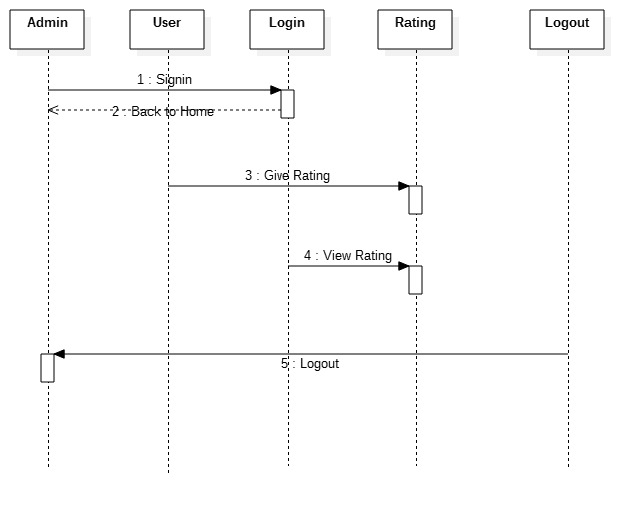
**Sequence Diagram for Route Module**

****

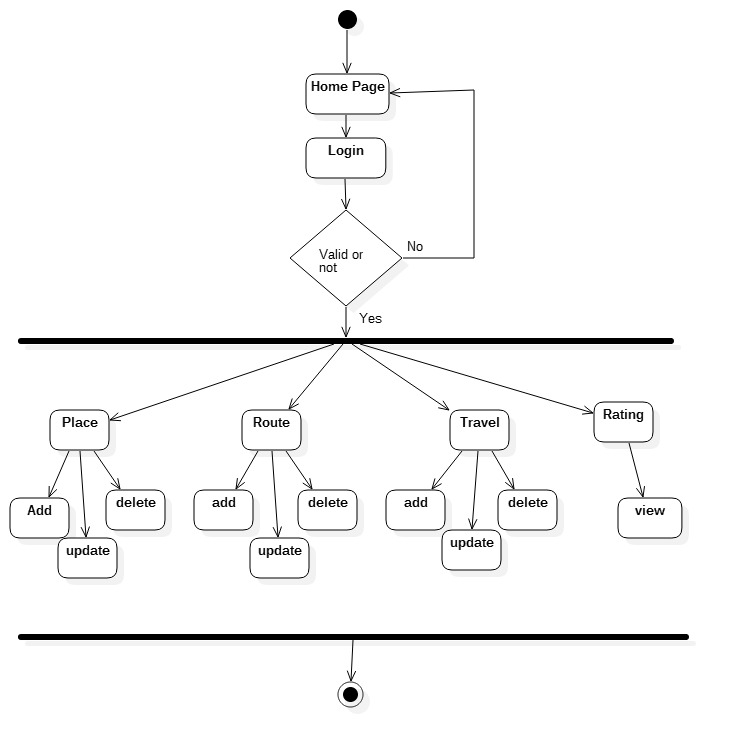
**Sequence Diagram for Travel Module**

****

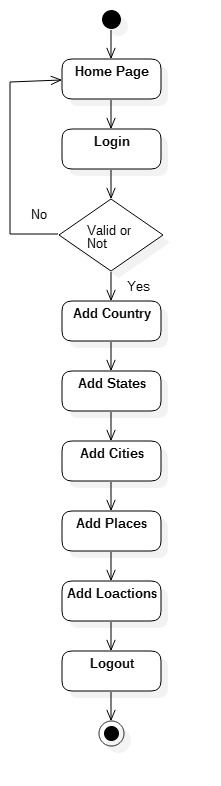
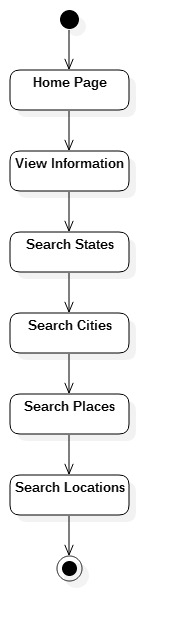
**Sequence Diagram for Rating Module**

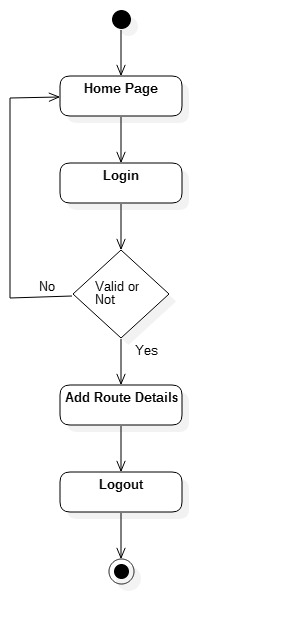
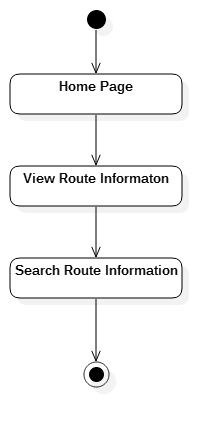
****

**Activity Diagram for Administration Module**

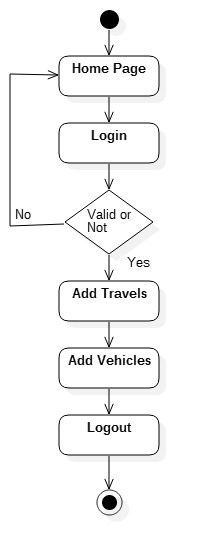
****

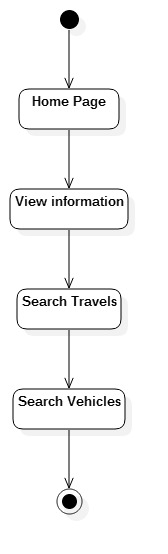
**Activity Diagram for Place Module**

**  
Activity Diagram for Route Module**

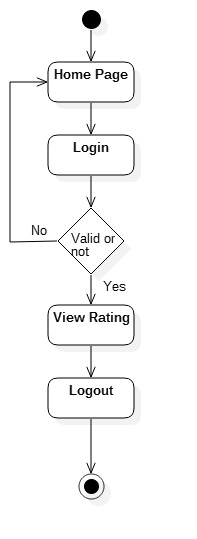
** **

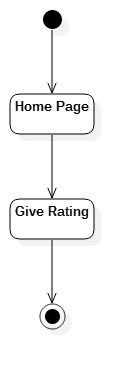
**Activity Diagram for Travel Module**

****

****

**Activity Diagram for Rating Module**

****

****

**3.8 DATA DICTIONARY:**

**NORMALIZATION:**

It is a process of converting a relation to a standard form. The process is used to handle the problems that can arise due to data redundancy i.e. repetition of data in the database, maintain data integrity as well as handling problems that can arise due to insertion, updating, deletion anomalies.

Decomposing is the process of splitting relations into multiple relations to eliminate anomalies and maintain anomalies and maintain data integrity. To do this we use normal forms or rules for structuring relation.

**Normal Forms:**

These are the rules for structuring relations that eliminate anomalies.

**First Normal Form:**

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. By this we mean simply that no attribute value can be set of values or, as it is sometimes expressed, a repeating group.

**Second Normal Form:**

A relation is said to be in second normal form is it is in first normal form and it should satisfy any one of the following rules.

* Primary key is not a composite primary key
* No non key attributes are present
* Every non key attributes is fully functionally dependent on full set of primary key.

**Third Normal Form:**

A relation is said to be in third normal form if their exits no transitive dependencies.

**Transitive Dependency**

If two non key attributes depend on each other as well as on the primary key then they are said to be transitively dependent. The above normalization principles were applied to decompose the data in multiple tables there by making the data to be maintained in a consistent state. In our project we have used three normal forms i.e., first normal form, second normal form, third normal form to decompose tables to avoid inconsistency in them.

Data Dictionaries are integral component of structured analysis. The data dictionary provides additional information about the system. A data dictionary is a catalogue repository provides additional information about the system. The data dictionary contains details of the fields, i .e and descriptions of these elements. After carefully understanding the requirements of the client the entire data storage requirements are divided into tables.

* Name - The primary name of the data or control item.
* Alias - Other names used for the first entry.
* Content description-A notation for representing contents.
* Supplementary information – Other information about data types, present values, restrictions or limitations, etc. The below tables are normalized to avoid any anomalies during the course of data entry.
* ADMINISTRATION TABLE
* STATES TABLE
* CITIES TABLE
* PLACE TABLE
* LOCATION TABLE
* ROUTE TABLE
* RATING TABLE
* TRAVELS TABLE
* VEHICLES TABLE
* TOURISTNEEDS TABLE

**ADMINISTRATION TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| id | VARCHAR2(10) | PRIMARY KEY | ID |
| name | VARCHAR2(50) | NOT NULL | NAME |
| gender | VARCHAR2(10) | NOT NULL | GENDER |
| mobileno | VARCHAR2(20) | NOT NULL | MOBILE NUMBER |
| password | VARCHAR2(10) | NOT NULL | PASSWORD |

This table supports up to 2nd Normal Form.

**STATES TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| statecode | VARCHAR2(10) | PRIMARY KEY | STATE CODE |
| countrycode | VARCHAR2(10) | NOT NULL | COUNTRY CODE |
| statename | VARCHAR2(50) | NOT NULL | STATE NAME |
| remarks | VARCHAR2(50) | NOT NULL | REMARKS |

This table supports up to 2nd Normal Form.

**CITIES TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| citycode | VARCHAR2(10) | PRIMARY KEY | CITY CODE |
| statecode | VARCHAR2(10) | FOREIGN KEY | STATE CODE |
| cityname | VARCHAR2(25) | NOT NULL | CITY NAME |
| distname | VARCHAR2(25) | NOT NULL | DISTRICT NAME |
| remarks | VARCHAR2(50) | NOT NULL | REMARKS |

This table supports up to 3rd Normal Form.

**PLACE TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| placecode | VARCHAR2(10) | PRIMARY KEY | PLACE CODE |
| placename | VARCHAR2(25) | NOT NULL | PLACE NAME |
| citycode | VARCHAR2(10) | FOREIGN KEY | CITY CODE |
| remarks | VARCHAR2(50) | NOT NULL | REMARKS |

This table supports up to 3rd Normal Form.

**LOCATION TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| lcode | VARCHAR2(10) | PRIMARY KEY | LOCATION CODE |
| lname | VARCHAR2(30) | NOT NULL | LOCATION NAME |
| placecode | VARCHAR2(10) | FOREIGN KEY | PLACE CODE |
| famous | VARCHAR2(50) | NOT NULL | FAMOUS |
| description | VARCHAR2(1000) | NOT NULL | DESCRIPTION |
| remarks | VARCHAR2(50) | NOT NULL | REMAERKS |
| citycode | VARCHAR2(10) | FOREIGN KEY | CITY CODE |

This table supports up to 3rd Normal Form.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| routecode | VARCHAR2(300) | PRIMARY KEY | ROUTE CODE |
| citycode | VARCHAR2(10) | FOREIGN KEY | CITY CODE |
| statecode | VARCHAR2(10) | FOREIGN KEY | STATE CODE |
| cityname | VARCHAR2(25) | NOT NULL | CITY NAME |
| distname | VARCHAR2(30) | NOT NULL | DISTRICT NAME |
| source | VARCHAR2(50) | NOT NULL | SOURCE |
| destination | VARCHAR2(50) | NOT NULL | DESTINATION |
| remarks | VARCHAR2(50) | NOT NULL | REMARKS |

**ROUTE TABLE**:

This table supports up to 3rd Normal Form.

**RATING TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| sno | VARCHAR2(10) | PRIMARY KEY | SERIAL NUMBER |
| rating | VARCHAR2(10) | NOT NULL | RATING |

This table supports up to 2nd Normal Form.

**TRAVEL TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| travelid | VARCHAR2(50) | PRIMARY KEY | TRAVEL ID |
| routecode | VARCHAR2(300) | FOREIGN KEY | ROUTE CODE |
| citycode | VARCHAR2(10) | FOREIGN KEY | CITY CODE |
| statecode | VARCHAR2(10) | FOREIGN KEY | STATE CODE |
| cityname | VARCHAR2(25) | NOT NULL | CITY NAME |
| distname | VARCHAR2(30) | NOT NULL | DISTRICT NAME |
| avehicles | VARCHAR2(100) | NOT NULL | AVAILABLE VEHICLES |
| remarks | VARCHAR2(50) | NOT NULL | REMARKS |

This table supports up to 3rd Normal Form.

**VEHICLES TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| vehicleid | VARCHAR2(10) | PRIMARY KEY | VEHICLE ID |
| travelid | VARCHAR2(50) | FOREIGN KEY | TRAVEL ID |
| cabs | VARCHAR2(50) | NOT NULL | CABS |
| ownerno | VARCHAR2(50) | NOT NULL | OWNER NUMBER |
| buses | VARCHAR2(100) | NOT NULL | BUSES |
| trains | VARCHAR2(100) | NOT NULL | TRAINS |
| flights | VARCHAR2(100) | NOT NULL | FLIGHTS |
| auto | VARCHAR2(50) | NOT NULL | AUTO |

This table supports up to 3rd Normal Form.

**TOURISTNEEDS TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| sno | VARCHAR2(10) | PRIMARY KEY | SERIAL NUMBER |
| citycode | VARCHAR2(10) | FOREIGN KEY | CITY CODE |
| statecode | VARCHAR2(10) | FOREIGN KEY | STATE CODE |
| cityname | VARCHAR2(25) | NOT NULL | CITY NAME |
| distname | VARCHAR2(30) | NOT NULL | DISTRICT NAME |
| placename | VARCHAR2(50) | NOT NULL | PLACE NAME |
| resturants | VARCHAR2(200) | NOT NULL | RESTURANTS |
| raddress | VARCHAR2(500) | NOT NULL | RESTURANTS ADDRESS |
| hotels | VARCHAR2(200) | NOT NULL | HOTELS |
| haddress | VARCHAR2(500) | NOT NULL | HOTEL ADDRESS |
| hospitals | VARCHAR2(200) | NOT NULL | HOSPITALS |
| hoaddress | VARCHAR2(500) | NOT NULL | HOSPITAL ADDRESS |
| shoppingmalls | VARCHAR2(200) | NOT NULL | SHOPPINGMALLS |
| saddress | VARCHAR2(500) | NOT NULL | SHOPPINGMALL ADDRESS |
| banks | VARCHAR2(200) | NOT NULL | BANKS |
| baddress | VARCHAR2(500) | NOT NULL | BANK ADDRESS |
| remarks | VARCHAR2(50) | NOT NULL | REMARKS |

This table supports up to 3rd Normal Form.

**SAMPLE CODING**

**4. SAMPLE CODING**

**LOCATION INSERT FORM:**

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@page import="java.sql.\*"%>

<%@include file="header1.jsp"%>

<%@include file="ppLocation.jsp"%>

<%!

Connection conn=null;

Statement st=null;

Statement st1=null;

ResultSet rs=null;

ResultSet rs1=null;

String placecode,citycode;

%>

<%

Class.forName("oracle.jdbc.driver.OracleDriver");

conn=DriverManager.getConnection("jdbc:oracle:thin:@localhost:

1521:XE","tourguide","durga");

st=conn.createStatement();

st1=conn.createStatement();

rs=st.executeQuery("select placecode from tblPlaces");

rs1=st1.executeQuery("select citycode from tblCities");

%>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Location Form</title>

<script src="js/jquery-1.12.0.min.js">

</script>

<script>

$(function(){

$("#btnsave").click(function(){

var lcode=$("#txtlcode").val();

var lname=$("#txtlname").val();

var placecode=$("#optplacecode").val();

var famous=$("#txtfamous").val();

var description=$("#txtdescription").val();

var remarks=$("#txtremarks").val();

var citycode=$("#optcitycode").val();

alert("hiiiiii");

$.post("srcLocationinsert.jsp",{txtlcode:lcode,txtlname:lname,optplacecode:placecode,txtfamous:famous,txtdescription:description,txtremarks:remarks,optcitycode:citycode},function(res,status){

alert(res);

});

});

});

</script>

</head>

<body>

<h1 align="center"><b>Location Information </b></h1>

<form name="frmLocationinsert.jsp" name="frmLocationinsert">

<table align="center" border="1">

<tr><td>Location Code:</td>

<td><input type="text" name="txtlcode" id="txtlcode"></td></tr>

<tr><td>Location Name:</td>

<td><input type="text" name="txtlname" id="txtlname"></td></tr>

<tr><td>Place Code:</td>

<td><select name="optplacecode" id="optplacecode">

<%

while(rs.next())

{

placecode=rs.getString("placecode");

%>

<option value="<%=placecode%>"><%=placecode%></option>

<%

}

%>

</select>

</td></tr>

<tr><td>Famous:</td>

<td><input type="text" name="txtfamous" id="txtfamous"></td></tr>

<tr><td>Description:</td>

<td><input type="text" name="txtdescription" id="txtdescription"></td></tr>

<tr><td>Remarks:</td>

<td><input type="text" name="txtremarks" id="txtremarks"></td></tr>

<tr><td>City Code:</td>

<td><select name="optcitycode" id="optcitycode">

<% while(rs1.next())

{

citycode=rs1.getString("citycode");

%>

<option value="<%=citycode%>"><%=citycode%></option>

<%

}

%>

</select></td></tr>

<tr><td><input type="button" value="save" id="btnsave"></td>

<td><p align="center"> <input type="reset" value="clear"></p> </td></tr>

</table>

</form>

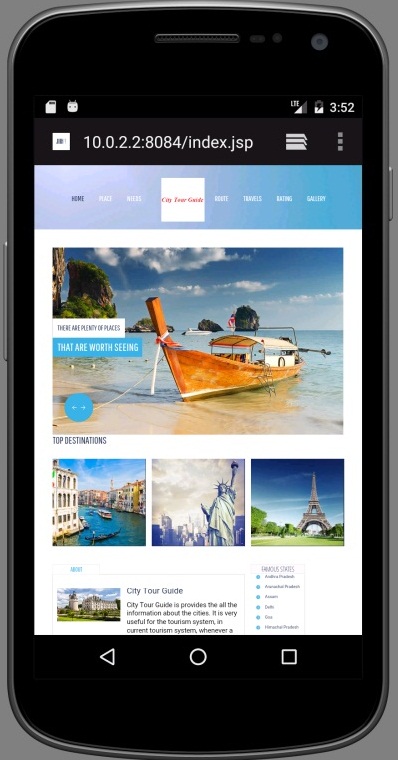
</body>

</html>

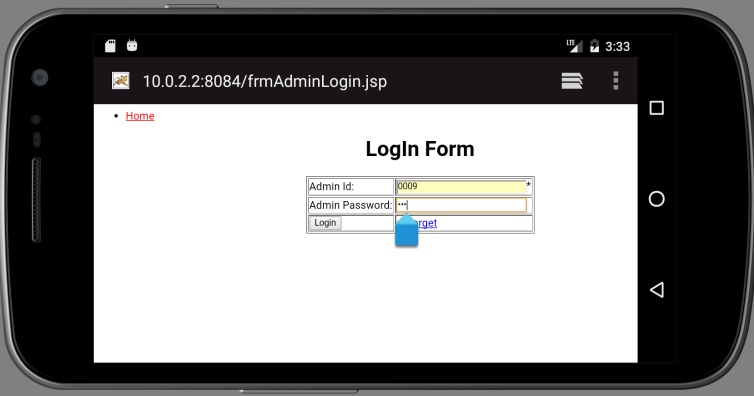
**FORMS**

**5. FORMS**

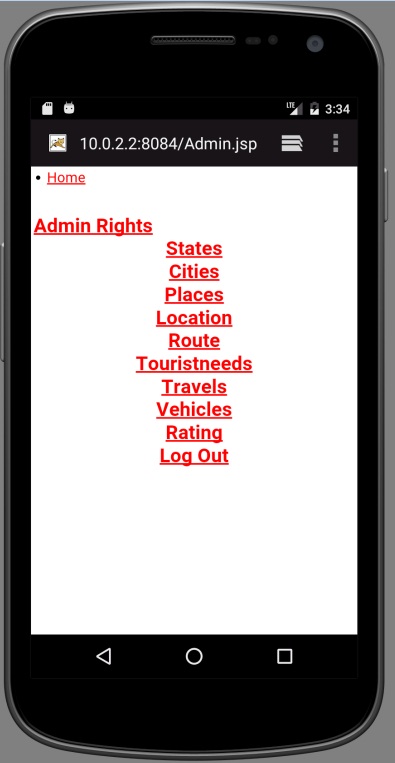
**HOME PAGE**

****

**ADMIN LOGIN FORM**

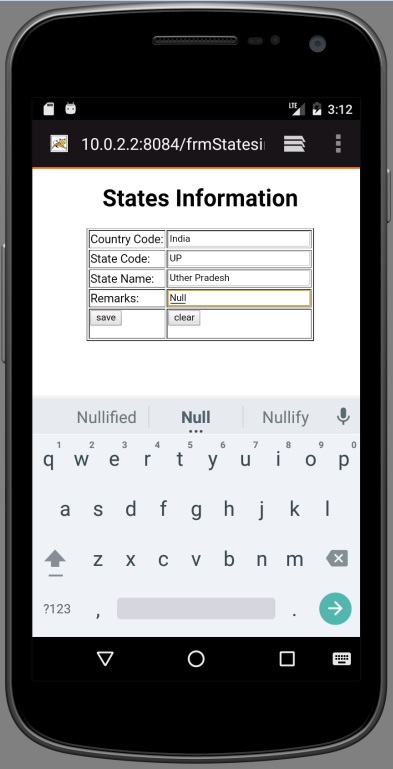
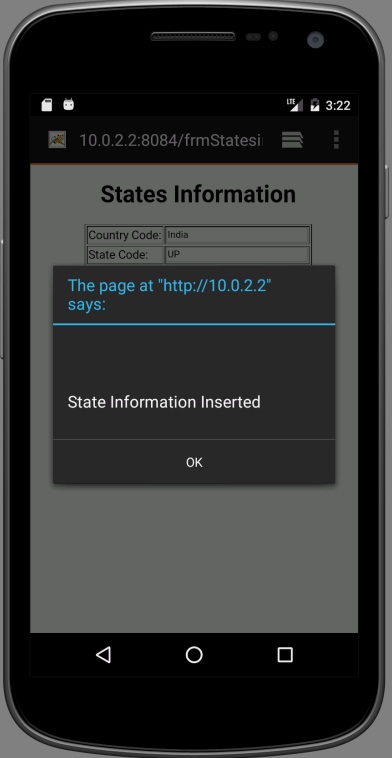
****

**ADMIN HOME PAGE**

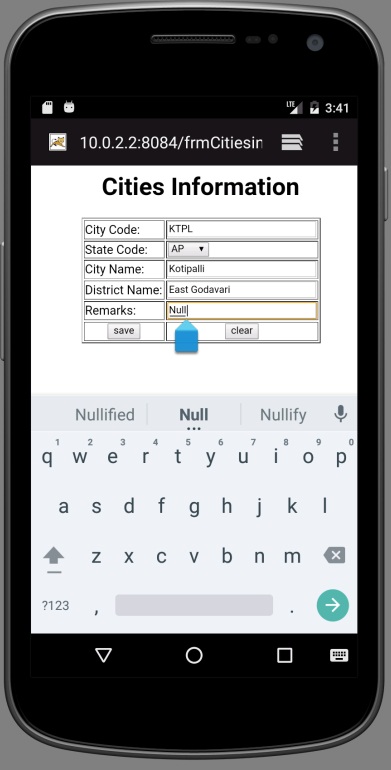
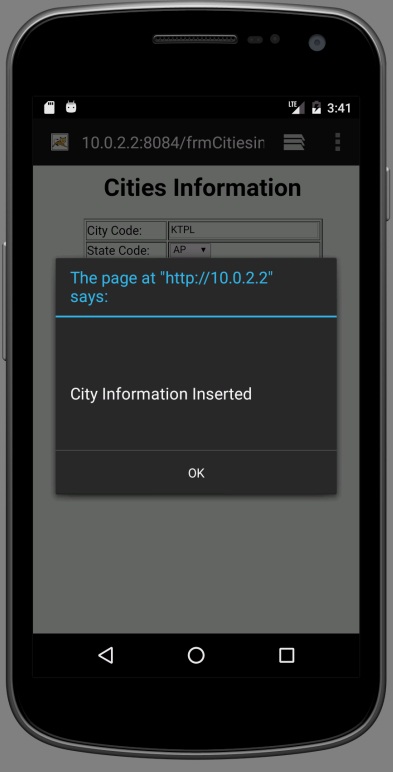
****

**INSERT FORMS**

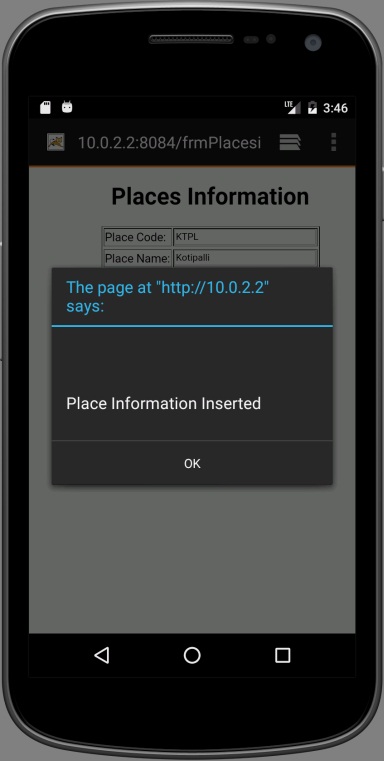
**States:**

** **

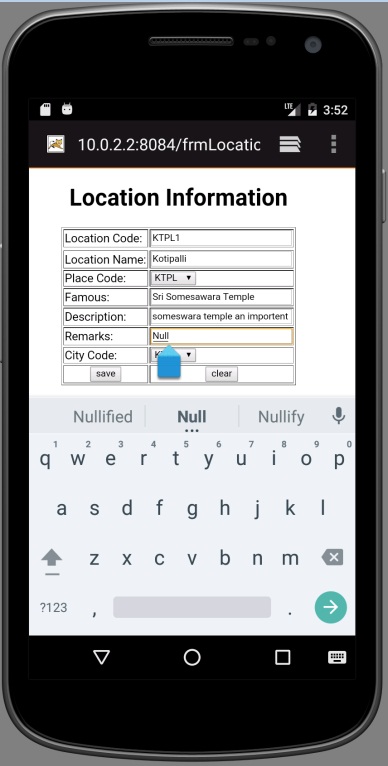
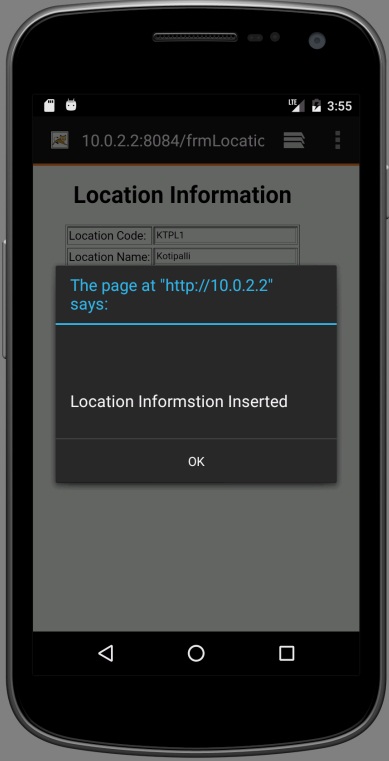
**Cities:**

** **

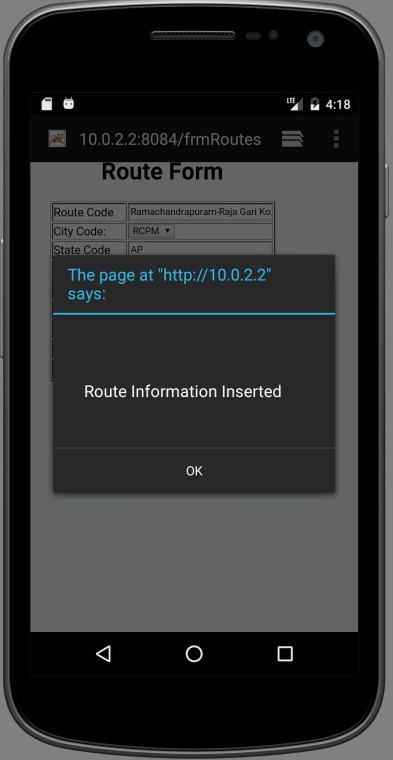
**Places:**

** **

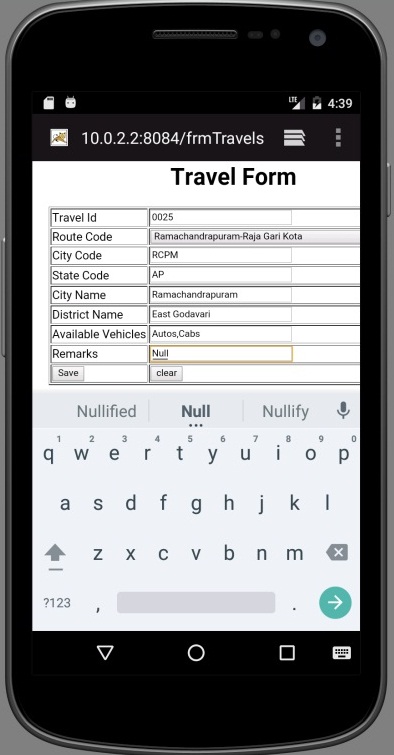
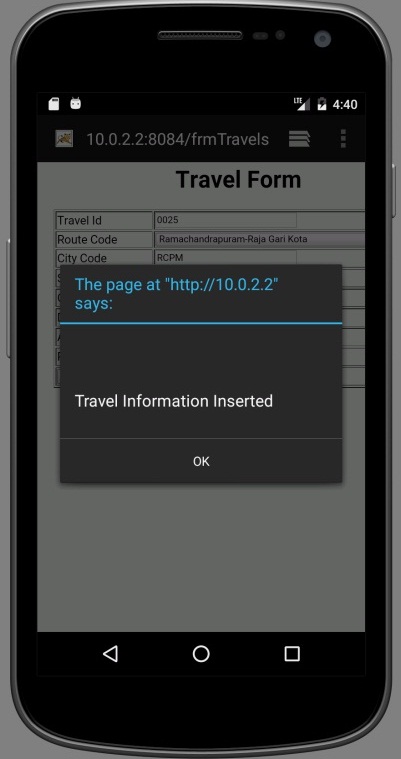
**Locations:**

** **

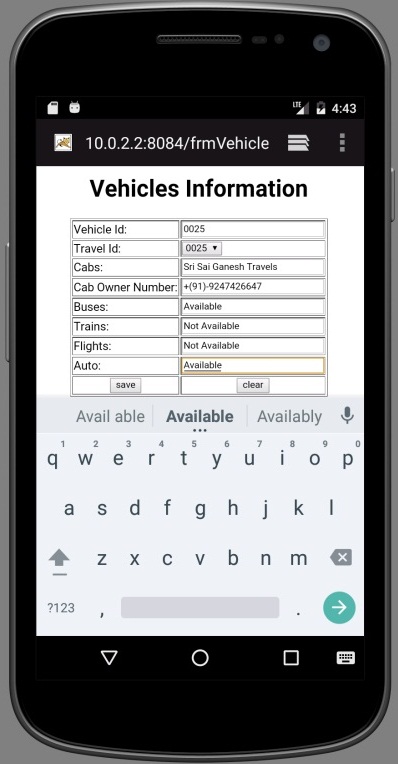
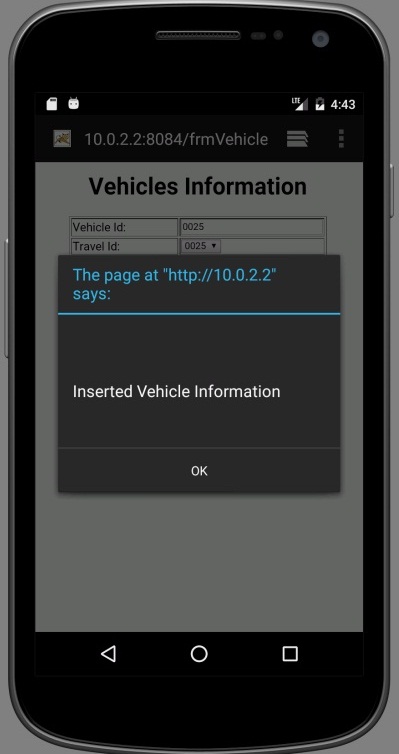
**Routes:**

** **

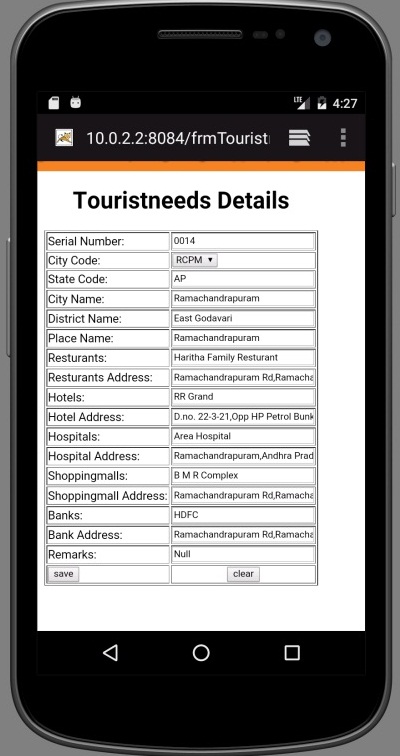
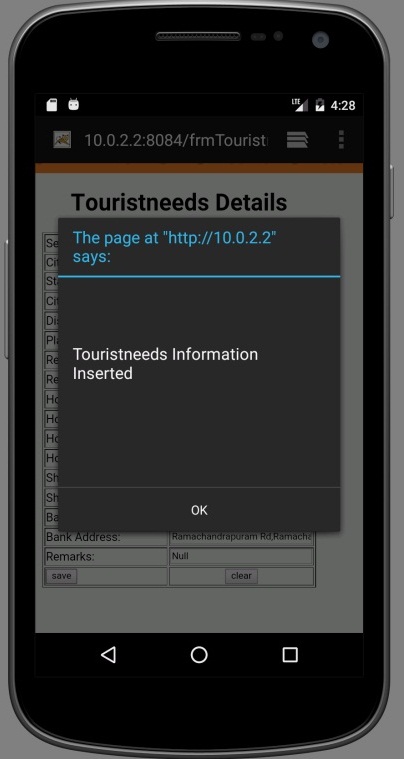
**Travels:**

****

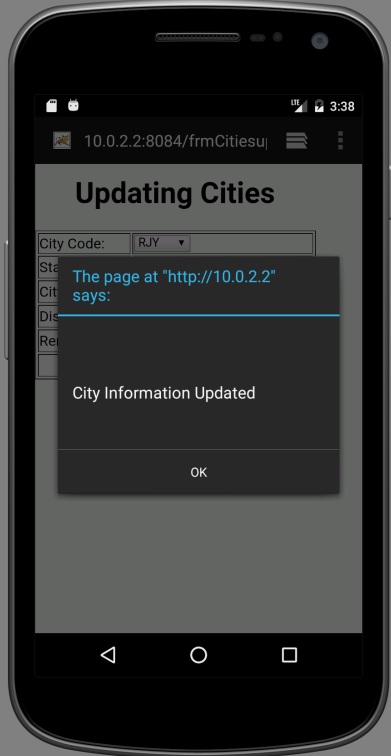
**Vehicles:**

** **

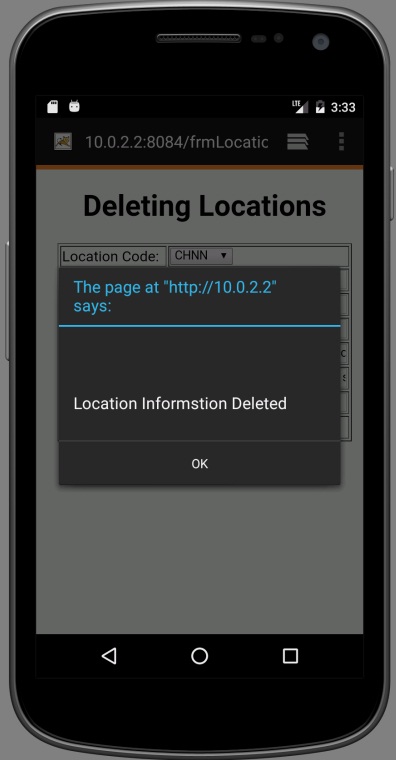
**Tourist needs:**

** **

**Update Form:**

** **

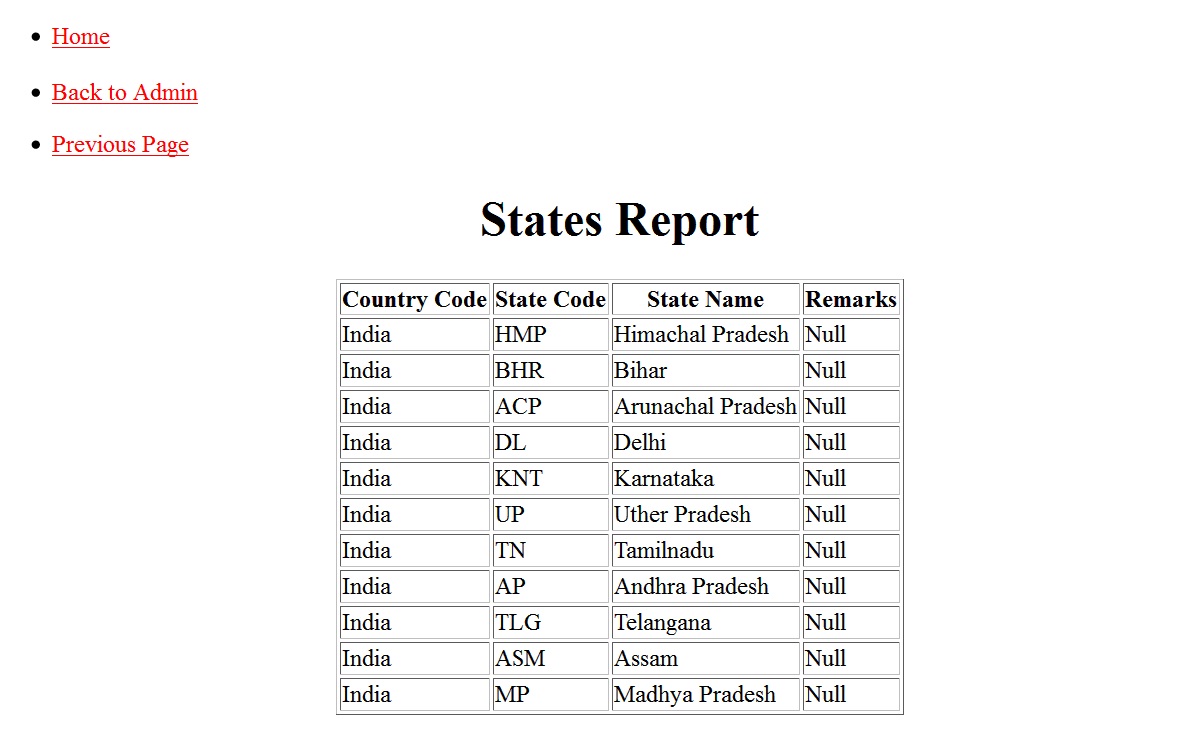
**Delete Form:**

** **

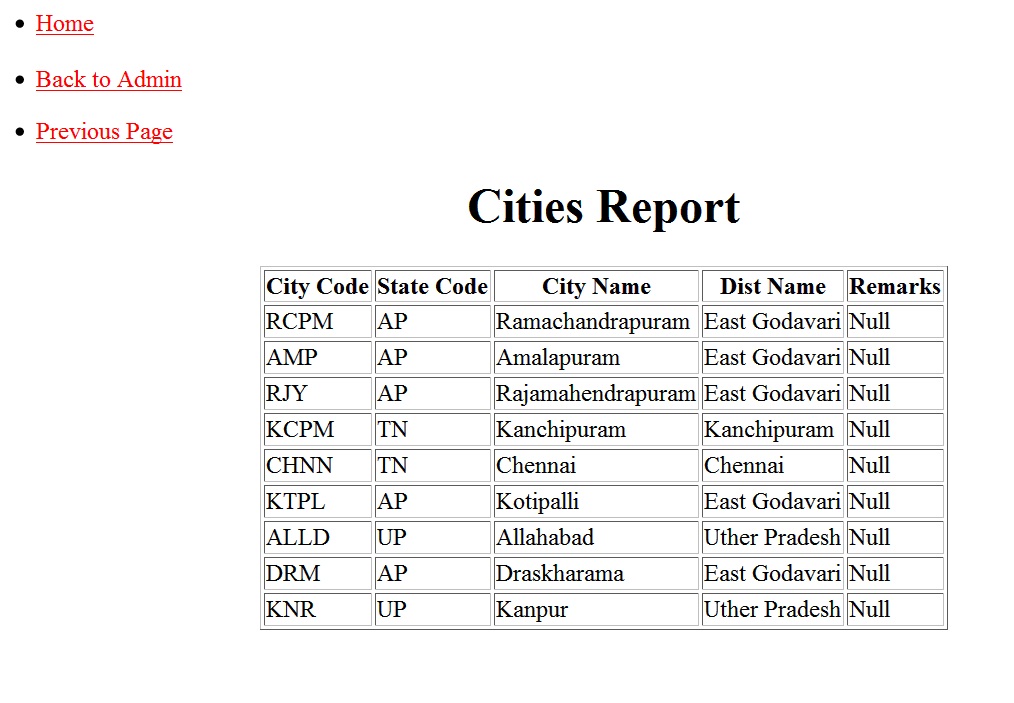
**REPORTS**

**6. REPORTS**

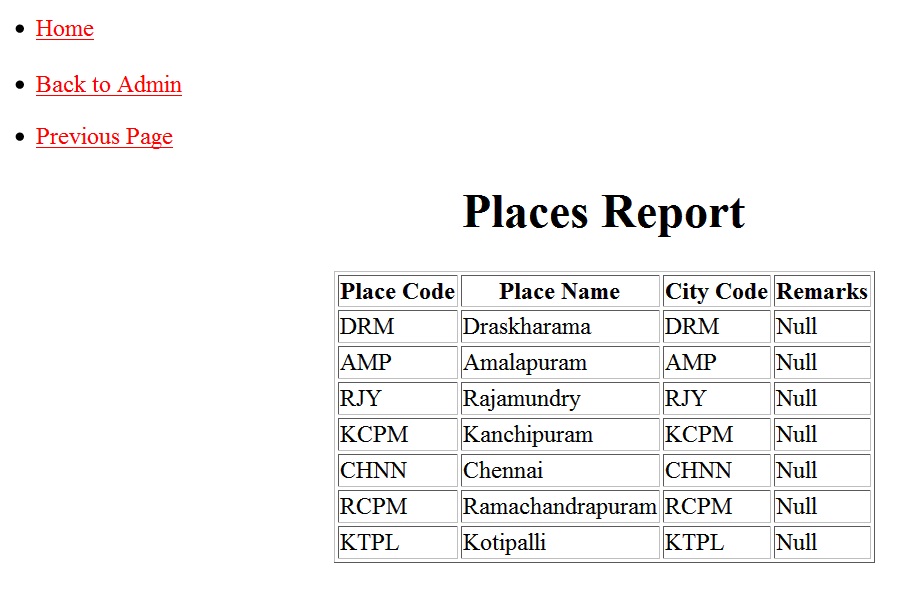
**States Report:**

****

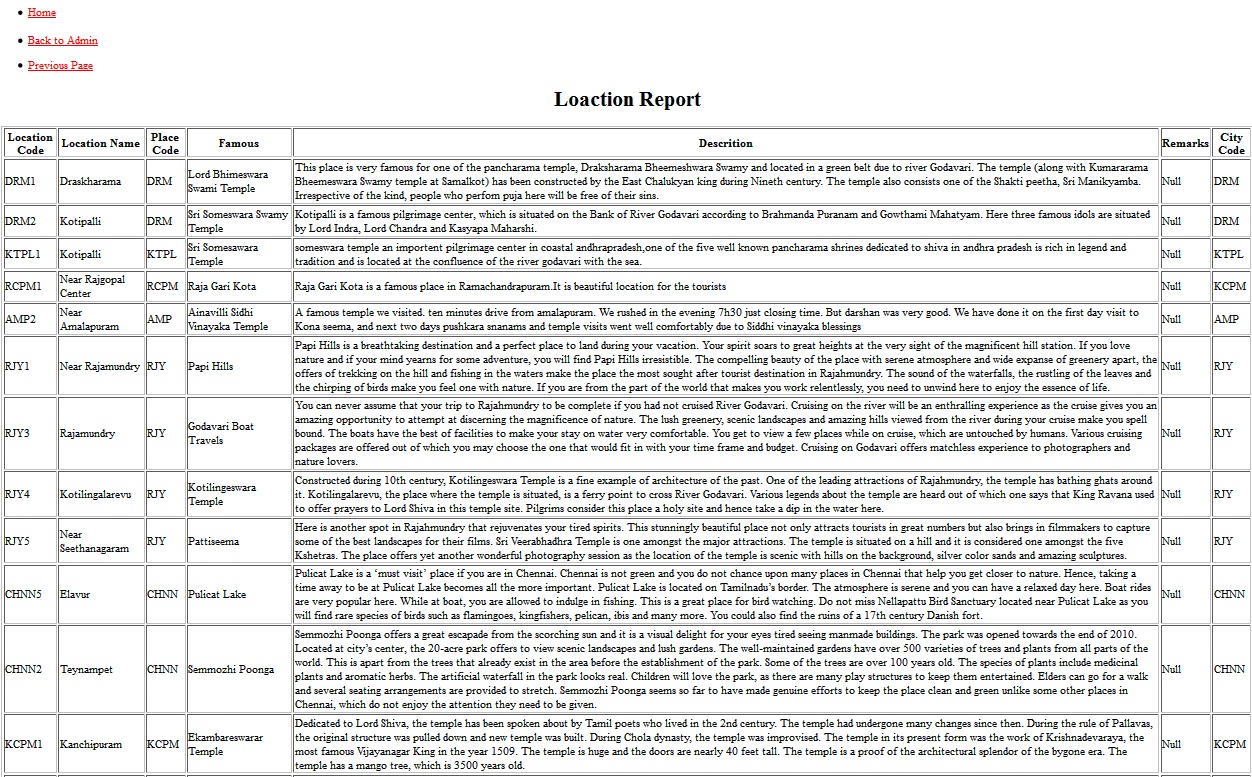
**Cities Report:**

****

**Places Report:**

****

**Location Report:**

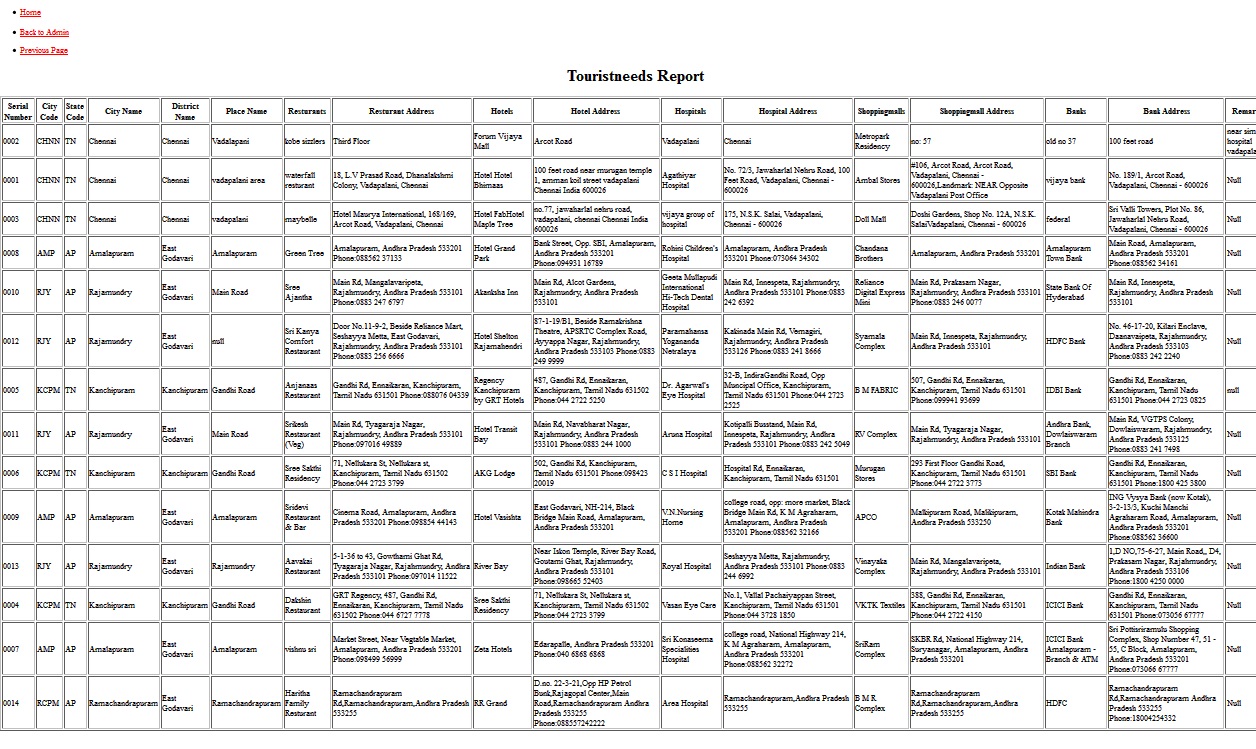
****

****

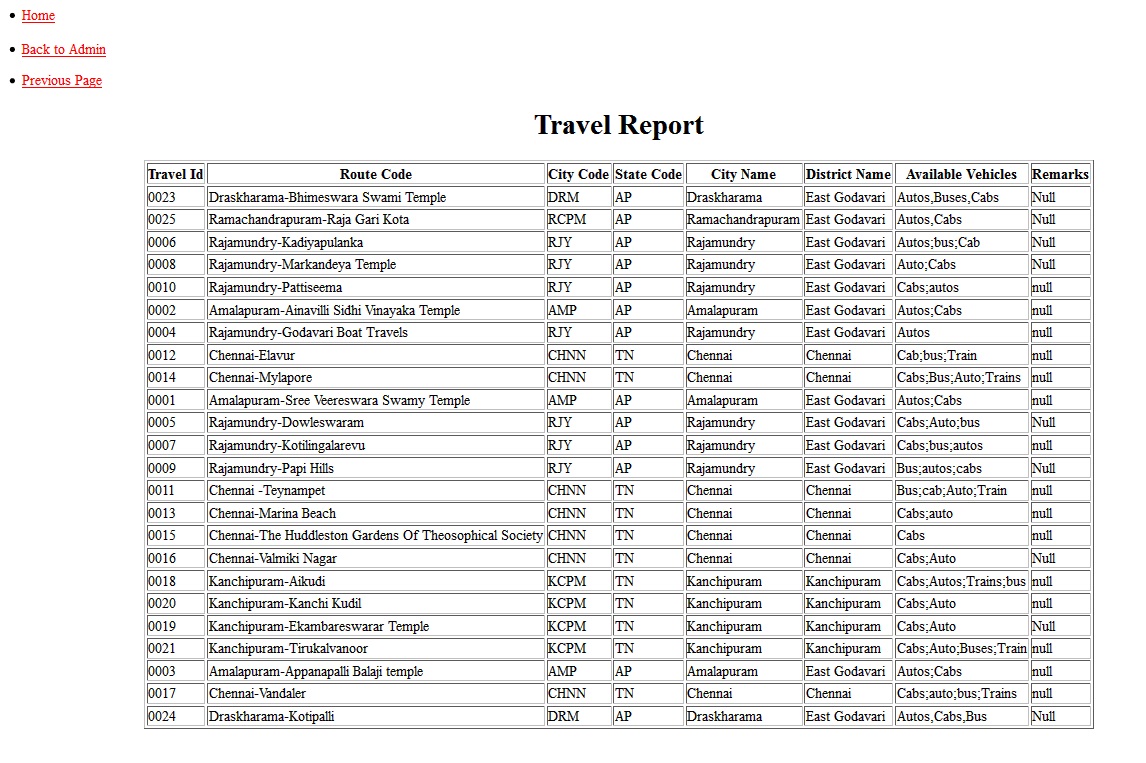
**Route Report:**

****

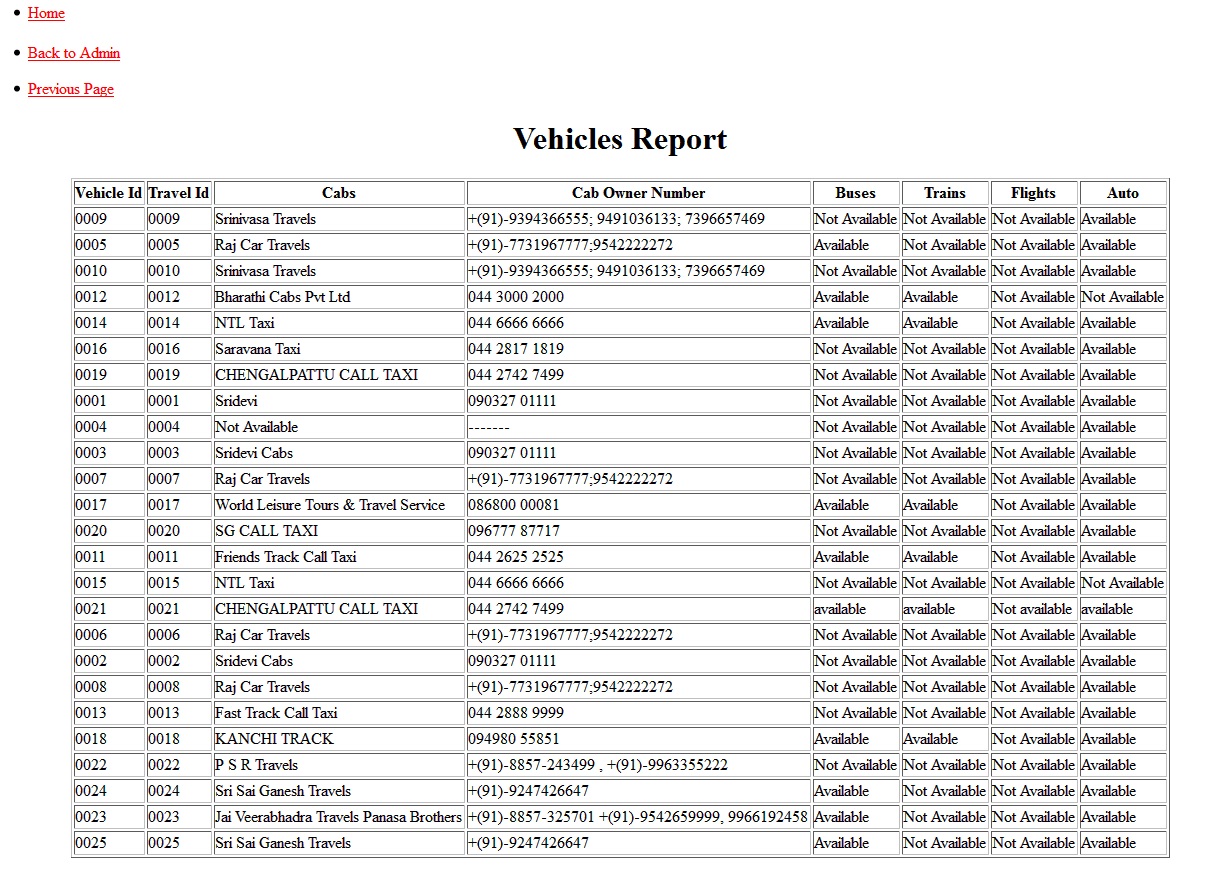
**Tourist Needs Report:**

****

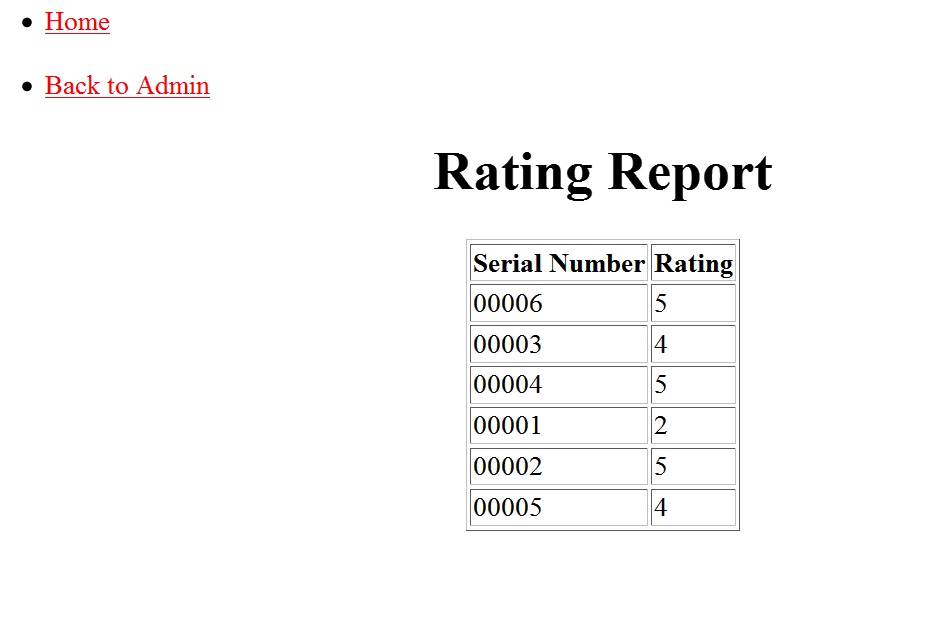
**Travel Report:**

****

**Vehicles Report:**

****

**Rating Report:**

****

**TESTING**

**7. TESTING**

Testing is a process of executing a program with the intent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding. System Testing is an important phase. Testing represents an interesting anomaly for the software. Thus a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

A good test case is one that has a high probability of finding an as undiscovered error. A successful test is one that uncovers an as undiscovered error.

**Testing Objectives:**

1. Testing is a process of executing a program with the intent of finding an error.
2. The tests are inadequate to detect possibly present errors.
3. The software more or less confirms to the quality and reliable standards.
4. A good test case is one that has a probability of finding an as yet undiscovered error.
5. A successful test is one that uncovers an undiscovered error.

The main objective of test in “Online City tour guide Application” is to test the admin id and their password and the access rights given to each authorized user.

**Levels of Testing:**

In order to uncover the errors present in different phases we have the concept of levels of testing. The basic levels of testing are

* Unit testing tests the minimal software component and sub-component or modules by the programmers.
* Integration testing exposes defects in the interfaces and interaction between integrated components (modules).
* Functional testing tests the product according to programmable work.
* System testing tests an integrated system to verify/validate that it meets its requirements.
* Acceptance testing can be conducted by the client. It allows the end-user or customer or client to decide whether or not to accept the product.
* Acceptance testing may be performed after the testing and before the implementation phase. See also Development stage.

Client Needs Acceptance Testing

Requirements System Testing

Design Integration Testing

Code Unit Testing

**7.1 unit testing:**

Unit testing focuses verification effort on the smallest unit of software i.e. the module. Using the detailed design and the process specifications testing is done to uncover errors within the boundary of the module. All modules must be successful in the unit test before the start of the integration testing begins. In this process of unit testing no users are needed.

In this unit testing, five modules are present. Each module provides the information for the users. The modules are

* Administration Module
* Place Module
* Routing Module
* Travelling Module
* Rating Module

The above modules can explains the entire project. Each module must be successful in the unit test. In this unit testing, then we take the Administration module, by this Module admin enter into the system and then he can do his work for data entry or entering to the administrator form. But when he/she enters invalid password then an alert message is displayed as “Password Mismatch”. Then I check the code of the form, and the entered Admin Id and corresponding passwords are checked with the data in the database of this system, after then the error is modified.

**7.2 integration testing:**

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors, associated with interfacing. The objective is to take the unit tested modules and build program structure that has been directed by the design.

After the unit testing, I had performed integration testing .The goal here is to see if modules can be integrated properly, the emphasis is being on testing interfaces between modules. This testing activity can be considered as testing the design and hence the emphasis on testing module interactions.

During integration, I choose home page of this system and administration module for integration. Generally when the administrator opens this system, immediately they find a home page window. After clicking their choosing the login link on the home page then opens the login form of the system which asks for Admin id and password to do their work for data entry or entering to the administrator to insert States details, Cities details Places details, routes details, travel details etc.

In this testing place module contains the states, cities, place, and location information. Routing module explains the famous place of user selected cites. Next travelling module explains the travel and vehicle information’s. Rating module is the application rating user can give the rating about the application.

**7.3 System testing:**

System testing is a method of testing applied on the whole system. The reference document for this process is the requirement document and the code is to see whether the software needs its requirements.

A classic system testing problem is finger pointing. This occurs when defect is uncovered. The software engineer should have potential interfacing problems and design error- handling paths that test all information coming from other demerits of the system conduct a series of tests that simulate bad data or other potential errors. In system testing the entire system was tested for various test cases with various inputs and also tested whether the linking is done properly or not. We found that all the forms are linked properly.

We test the entire system with various inputs and then we observe the behavior of the system, and found that the working of the entire system is proper by all ways.

Here entire system has been tested against requirements of project and it is checked whether all requirements of project have been satisfied or not. In system testing, the entire project is tested i.e. testing is done from the starting page (login page) to the ending page (logout page).In system testing overall system is tested means each and every module of the system is tested, i.e. all forms, hyperlinks of the forms, operations performed (insert, delete, and update) on the forms, reports and outputs of the forms are tested.

**7.4 ACCEPTANCE TESTING:**

We perform this testing with the realistic data of the client to demonstrate that the software is working satisfactorily. In this testing we focus on the external behavior of the system. The internal logic of the program is not emphasized. All my project checked with the acceptance testing the Admin Id and password and get the administrator page .Client tested all the options by clicking the options. He checked the each and every option in the admin and organization modules and accepted.

**Features To Be Tested / Not To Be Tested:**

Here the following features are tested as part of testing which are part of the functional and non-functional requirements of client support optimizer. The testing looks for whether the system

* Identifies administrator based on the authentication.
* Allows Administrator login based on the Admin Id and password.
* Is providing proper communication between various subsystems.
* Is handling errors and extreme conditions.
* Is hardware compatible as specified in the system requirements?

**PASS /FAIL CRETERIA:**

The pass/fail criteria for the above tests are to compare the results with those requirements that are specified in the requirements analysis document of the project.

**TESTING MATERIALS (H/W AND S/W REQUIREMENTS):**

The system needs a class file of the whole system to be tested and testing personal to test the system. The system can be tested on any P.C with the configuration specified in the h/w considerations in software specifications.

**7.5 TEST CASES:**

The testing phase is an important part of software development. It is the process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. A realistic goal for testing is to select as set of test cases. In the functional testing, the test cases are decided solely on the basis of requirements.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S. No** | **Input** | **Condition Being Checked** | **Expected**  **Behavior** | **Observed**  **Behavior** | **Status**  **Pass/Fail** |
| 1. | Enter the correct Admin id and password. | Password and admin id are correct. | Admin Home page should be displayed. | Admin Home  Page is Displayed. | Pass. |
| 2. | Enter the correct admin id and wrong password. | Password and admin id are not correct. | Error message should be displayed. | Error message is displayed. | Pass. |
| 3. | Admin fill the all the fields of the forms. | Validate all the fields. | Information should be Stored. | Information saved. | Pass |
| 4. | Admin fill the all the fields of the forms. | Validate all the fields. | Information should not be Stored. | Information is not saved. | Pass |
| 5. | Admin fill the all the fields of the forms. | Validate all the fields. | Information should be Stored. | Information not saved. | Fail |
| 6. | Search the Place. | Validate the Place. | Place Information should be displayed. | Place Information is  Displayed. | Pass. |
| 7. | Not Search the validate Places. | Validate the Place Information. | Should not Search the Place details. | Cannot search the Place details. | Pass. |
| 8. | Search the Route of the Route form. | Validate the Route. | Route Information should be displayed. | Route Information is  Displayed. | Pass. |
| 9. | Not Search the validate Routes in route form. | Validate the route Information. | Should not Search the route details. | Cannot search the route details. | Pass. |
| 10. | Search the Travels of the Travel form. | Validate the Travels. | Travel Information should be displayed. | Travel Information is  Displayed. | Pass. |
| 11. | Not Search the validate Travels in Travels form. | Validate the Travels Information. | Should not Search the Travels details. | Cannot search the Travels details. | Pass. |
| 12. | Do not Search the all fields of the Home form. | Validate all the fields. | Searching Information should not be displayed. | Searching details are displayed. | Fail. |

**CONCLUSION**

**8. CONCLUSION**

We present the design of android based City Tour Guide system. The system provides the information query of Places, Routes, travels, Live Locations and so on. The system is a combination of Smart-phone and Internet to facilitate the tour of user.

The implementation of this project reduces the over expenditure of such companies and makes economical balance. This project has been greatest knowledge experienced for me, as in a short period of time I have been exposed to the complexities of computer fields. Thus had an opportunity to work in a real time environment and to learn a lot. During this project development, we improved the process by incorporating validations in each level of the system. Based on this, it is simply understand by the end-user.

This project” Online City Tour Guide Application” is very helpful to the Tourists. The reports generated by this project are helpful to the Users. This project is very user friendly, effective and efficient software.

**USER MANUAL**

**8. USER MANUAL**

The user manual serves as a guide for the user to use the system who need not have any idea about project. This user manual is very much useful for large projects. But when it comes to small projects like the present system there is no need for the user’s manual as if the organization wants to computerize the system then they believe that the people working for it are capable of handling the computerization. They also believe that system. This system is developed keeping in view all these things. Still in some there may something about it. In this user manual measures have been taken to explain each and every issue where the users may fall in confusion.

**User Manual for Administration:**

To login into the administration module, the administrator has to select Admin menu then login form is opened. Now administrator has to provide his/her id and password to enter into the administration module. Once these are valid, he entered into the administration module.

The administrator can maintain all the master data details like Place Information details, Route Information details, and Travel Information details.

Admin can generate the all the information. After generation admin can generate all Information details to the User.

**User Manual for User:**

User can access this application after that user can seen home page. In this home page some fields are available to use this application like place, route, travel etc..User must be search the location that is famous tourist places. Once user search the details of the one place, select the city and click the search button after that all the information will be display on the screen. Another part is route and travels with GPS support to user can search the locations. If he wants to know the details of the any particular information, he just clicks on the module. Then the details of the module are opened. When the details of module are opened, he can saw the all the information present in the system.

User must searches his/her place details. After searching the place user can find out the location about the place within the GPS. In this project user can act as a guide.

**BIBLIOGRAPHY**

**10. BIBLIOGRAPHY**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **AUTHOR** | **TITLE** | **PUBLISHER** | **YEAR** | **EDITION** |  |  |  |  |  |
| Bernd Bruegge, Allen H. Dutoit | Object Oriented Software Engineering | Pearson Education | 2006 | 2nd |  |  |  |  |
| Scott Urman | Oracle 10g PL/SQL Programming | Tata McGraw Hill Edition | 2006 | 3rd |  |  |  |  |
| Marty Hall, Larry Brown | Core Servlets and Java Server Pages | Pearson Education | 2006 | 2nd |  |  |  |  |
| James Jaworski | Mastering JavaScript | BPB Publications | 1997 | 1st |  |  |  |  |
| Elmasri Navathe | Fundamentals of Database Systems | Pearson Education | 2000 | 2nd |  |  |  |  |
| Reto Meier | Professional Android 4 Application Development | Wiley India Pvt Ltd | 2012 | 2nd |  |  |  |  |

**Referred Websites:**

[www.wikipedia.org.](http://www.wikipedia.org/)

<www.developers.android.com>

<www.roseindia.net/jsp/>

[www.wikipedia.com](http://www.wikipedia.com/)