2009

One Stop Information Site

Poongkuzhali Dhanasekaran

ONE STOP INFORMATION SITE

A Project

Presented to the

Faculty of

California State University,

San Bernardino

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Computer Science

---

by

Poongkuzhali Dhanasekaran

June 2009

ONE STOP INFORMATION SITE

A Project

Presented to the

Faculty of

California State University,

San Bernardino

by

Poongkuzhali Dhanasekaran

June 2009

Approved by:

[redacted]

Ernesto Gomez, Chair, Computer Science

May 6, 2009

Date

[redacted]

Richard J. Botting

[redacted]

Haiyan Qiao

ABSTRACT

One Stop Information Site is the web application designed to provide users with interactive day to day information like daily weather, traffic alerts, daily news, directions, tourist's attractions and restaurant list for the specified zip code. OSIS is secure, easy to use and also less time consuming website. The feature which differentiates this website from other websites is that, this is a modular site with easy plugability of features and has the facility to view page according to the user's preferred theme and layout. All modules in this page were linked to the map, so that the information can be viewed with the geographic location.

ACKNOWLEDGEMENTS

This report could not have been written without the help and encouragement of my Committee chair Dr. Ernesto Gomez and my committee members Dr. Richard J. Botting and Dr. Haiyan Qiao. I would like to render them special thanks for guiding me throughout the course of my project.

I would also like to thank my family members for their contribution to my project. My Grandparents, Selvaraju and Angammal for their blessings. My Parents, Dhanasekaran and Vijayalakshmi for their encouragement in accomplishing my goals. To my brother Karki for his contribution towards my project banner.

I would like to render my heart full of thanks to my lovable husband Kabilan who served as a ladder in reaching my goal. Last but not least I would like to thanks my smart little son Khaveen for his cooperation in completing this project.

To Kabilan Navanidame

TABLE OF CONTENTS

CHAPTER TWO: INTERFACE DESIGN

CHAPTER THREE: ONE STOP INFORMATION SITE

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

1.1 Purpose of Project

The Purpose of this project is to construct a website to provide users with all superlative and precise information in one easily accessible site.

1.2 Scope of Project

The scope of this project is to develop an easy to use and less time consuming website to provide day-to-day information like weather forecast, current traffic, directions, news, tourist attractions and restaurant lists for the specified zip code. The feature which differentiates this website from other websites is that, this is a pluggable module and this site can be viewed according to the user's preference. All modules are linked to the map, so that the information can be viewed with the geographic location.

1.3 Definitions, Acronyms, and Abbreviations

The following terms are defined as they apply to the project.

- Apache Tomcat: Tomcat is an application server from the Apache Software Foundation that executes Java Servlet and renders Web pages that include Servlet and Java Server Page coding. Described as a "reference implementation" of the Java Servlet and the Java Server Page specifications, Tomcat is the result of an open collaboration of developers and is available from the Apache Web site in both binary and source versions. Tomcat can be used as either a standalone product with its own internal Web server or together with other Web servers, including Apache, Netscape Enterprise Server, Microsoft Internet Information Server (IIS), and Microsoft Personal Web Server. Tomcat requires a Java Runtime Enterprise Environment that conforms to JRE 1.1 or later.
- GUI: GUI stands for Graphical User Interface. GUI allows user to navigate and interact with applications with a simple mouse click. Well-

designed graphical user interface can free the user from learning complex command languages.

- GIMP: GIMP stands for GNU Image Manipulation Program. It is free software, raster graphics editor used to process digital graphics and photographs. This software is used for creating graphics and logos, resizing and cropping photos, altering colors, combining multiple images, removing unwanted image components, and converting between different image formats. GIMP can also be used to create basic animated images in GIF format.
- IP: IP stands for Internet Protocol. IP is an identifier for computer or device on a TCP/IP network. Networks using the TCP/IP protocol route messages based on the IP address of the destination. The format of an IP address is a 32-bit numeric address written as four numbers separated by periods. Each number can be zero to 255. For example, 1.160.10.240 could be an IP address.
- Java: Java is a high-level programming language developed by Sun Microsystems. It is an object-oriented language similar to C++, but simplified to

eliminate language features that cause common programming errors. Java source code files are compiled into a format called bytecode, which. can then be executed by a Java interpreter. Compiled Java code can run on most computers because Java interpreters and runtime environments, known as Java Virtual Machines, exist for most operating systems, including UNIX, the Macintosh OS, and Windows. Bytecode can also be converted directly into machine language instructions by a just-in-time compiler (JIT).

- JDBC: JDBC is an acronym for Java Database Connectivity. JDBC is an application program interface specification for connecting programs written in Java to the data in databases. JDBC specifies a set of object-oriented classes for the programmer to use in building SQL requests. The application program interface lets you encode access request statements in Structured Query Language that are then passed to the program that manages the database.

- DIA: Dia is a gtk+ based diagram creation program. It is inspired by the commercial Windows program 'Visio', though more geared towards informal diagrams for casual use. It can be used to draw many different kinds of diagrams. It currently has special objects to help draw entity relationship diagrams, UML diagrams, flowcharts, network diagrams, and many other diagrams. It is also possible to add support for new shapes by writing simple XML files, using a subset of SVG to draw the shape.
- JUDE: JUDE/Professional is a unique System Design Tool which supports UML, Entity Relationship Diagram, Flowchart, CRUD, Mind Map(R) and Data Flow Diagram.
- TCP: TCP is an acronym for Transmission Control Protocol. It is used to establish connection-oriented, sequenced and error-free sessions over IP networks. TCP is intended for use as a highly reliable transport Protocol between hosts in packet-switched computer communication Networks, and in interconnected systems of such Networks. TCP is a

flow-controlled, connection-oriented, end-to-end reliable Protocol designed to fit into a layered hierarchy of Protocols supporting multi-network applications. TCP provides for reliable interprocess communications between pairs of processes in host computers attached to distinct but interconnected computer communication Networks. Very few assumptions are made as to the reliability of the communication Protocols below the TCP layer. TCP assumes it can obtain a simple, potentially unreliable, datagram service from the lower level Protocols, usually IP.

- UML: UML stands for Unified Modeling Language. It is an object-oriented analysis and design language from the Object Management Group. It is a graphical visualization language. It consists of a series of symbols and connectors that can be used to create process diagrams and is often used to model computer programs and workflows.

- MVC: MVC stands for model view controller. It is a design pattern for the architecture of web applications

- Servlet: A Java application that runs in a Web server or application server and provides server-side processing such as accessing a database and e-commerce transactions. Widely used for Web processing. Servlets are designed to handle HTTP requests.

- XML: XML stands for EXtensible Markup Language. It is used for defining data elements on a Web page. XML uses a similar tag structure as HTML; however, whereas HTML defines how elements are displayed, XML defines what those elements contain. While HTML uses predefined tags, XML allows tags to be defined by the developer of the page.

- SAX: SAX stands for Simple API for XML. SAX is a popular API for manipulating XML documents. SAX does not provide a random access lookup to the document's contents. It scans the document sequentially and presents each item to the application only once.

- XHTML: XHTML is an acronym for EXtensible HTML. XHTML combines HTML and XML into a single format. Like XML, XHTML can be extended with proprietary tags. Also like XML, XHTML must be coded more rigorously than HTML.
- XSLT: XSLT stands for eXtensible Stylesheet Language Transformation. It is software that converts a XML document into another format such as HTML, PDF or text. It may also be used to convert one XML document to another XML document with a different set of XML tags.
- CSS: CSS stands for Cascading Style Sheets. It is a style sheet format for HTML documents endorsed by the World Wide Web Consortium. It is a simple mechanism for adding style (e.g. fonts, colors, spacing) to Web documents.
- AJAX: Ajax stands for Asynchronous JavaScript and XML. It is a web development technique used for creating interactive web applications. The intent is to make web pages feel more responsive by exchanging small amounts of data with the server behind the scenes, so that the entire web page does not have to

be reloaded each time the user requests a change. This is intended to increase the web page's interactivity, speed, functionality, and usability.

- Google Maps API: The Google Maps API lets us to embed Google Maps in our own web pages with JavaScript. The API provides a number of utilities for manipulating maps and adding content to the map through a variety of services, allowing us to create robust map applications on our website.
- Dojo: Dojo is an Open Source DHTML toolkit written in JavaScript. Dojo allows us to easily build dynamic capabilities into web pages and any other environment that supports JavaScript sanely. We can use the components that Dojo provides to make our web sites more usable, responsive, and functional. With Dojo we can build degradable user interfaces more easily, prototype interactive widgets quickly, and animate transitions.

1.4 Overview

This section is organized in such a way to provide an outline for the functional and specificational requirements of One Stop Information Site. Remaining Sections provides a clear view of the project's descriptions and functionalities.

1.5 Overall Description

1.5.1 Project Perspective

One Stop Information Site is able to provide day to day information for the user. If the zip code of the area is provided OSIS will display the required information based on the user requirement. This project will be completely web-based. The following section will describe how One Stop Information Site operates inside various constraints.

1.5.2 System Interface

The client will use web browser as system interface to send request to the server. In order to display the home page of OSIS, the client's browser must support JavaScript.

1.5.3 Software Interface

One Stop Information Site will run on web browsers for Windows, Linux, or Mac OS. The interface will be developed by Java, Servlet, JavaScript, XML, XHTML, XSLT, AJAX, Google API, Dojo and CSS.

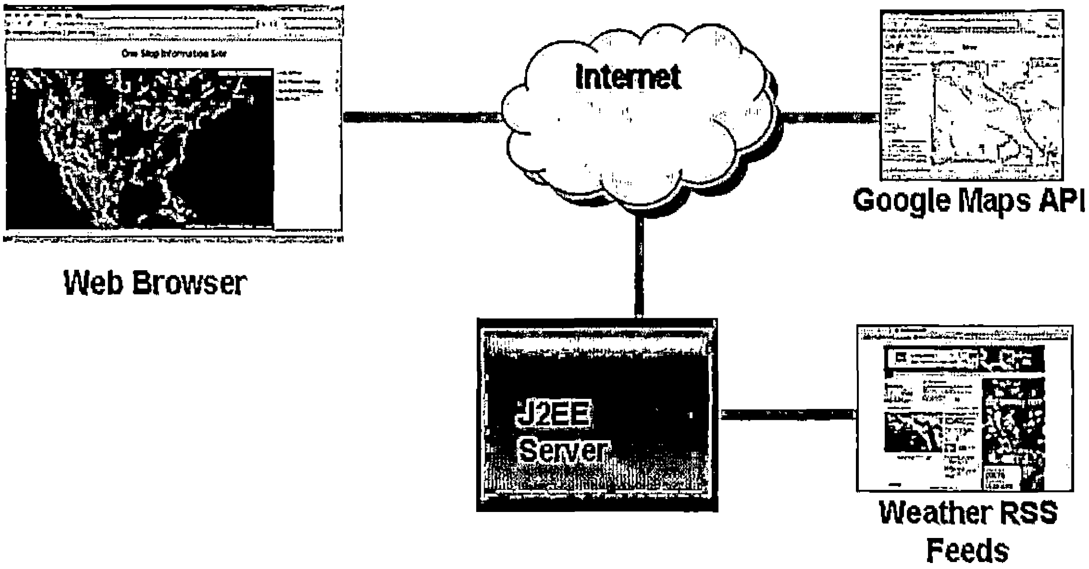


Figure 1. Software Interface Diagram

1.5.4 Communication Interface

The communication interface requires any compatible web browser such as Mozilla, Firefox and Internet Explorer.

1.5.5 Memory Requirement

128MB of RAM is the minimum required for One Stop Information Site.

1.5.6 Project Functions

This section illustrates the interactions between the active users and the system. One Stop Information Site will include several functions: Weather, directions, traffic, news, tourist attractions and restaurant lists.

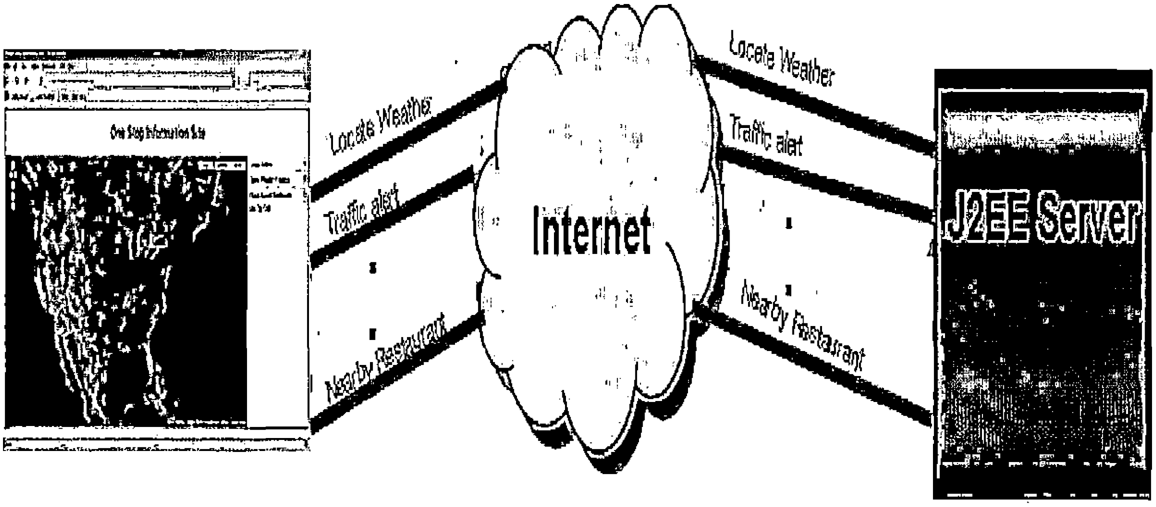


Figure 2. Functional Diagram

1.6 Specifications

1.6.1 Hardware Interfaces

The OSIS must be on a computer connected to the Internet. OSIS will function on Windows XP, Mac OS, and Linux.

1.6.2 Minimum PC Requirements

800 MHz processor and 256 MB of Ram is the minimum requirement to run OSIS.

1.6.3 Software Interfaces

OSIS operates through the user's web browser. The user must use one of the following browsers: Internet Explorer 6.x or higher, Firefox 1.5.x or higher and must have JavaScript enabled.

1.6.4 Communication Interfaces

OSIS requires the user to be connected to the internet with a connection of at least 384 Kbps (DSL). They may be connected wirelessly or through their NIC (Network Interface Card).

CHAPTER TWO

INTERFACE DESIGN

Based on the requirements collected in the previous phase the User interface, Database design and object model were designed in this phase.

2.1 User Interface Design

The design of user interface was based on end-user requirements. During this phase the navigation paths between web pages and the mock-up of the OSIS site is decided.

2.1.1 Structure and Navigation

The overall structure of the application is relatively simple, as shown in the following diagram. There are three main modules in this site and the navigation between these modules is represented below.

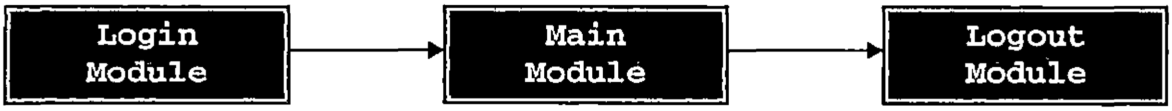


Figure 3. System Navigation

The structure is further broken down in the following diagrams.

2.1.1.1 Navigation Chart For Login Module. The login module handles user validation. In this module the user is allowed to choose his layout and choice of interest for his main page through the registration page.

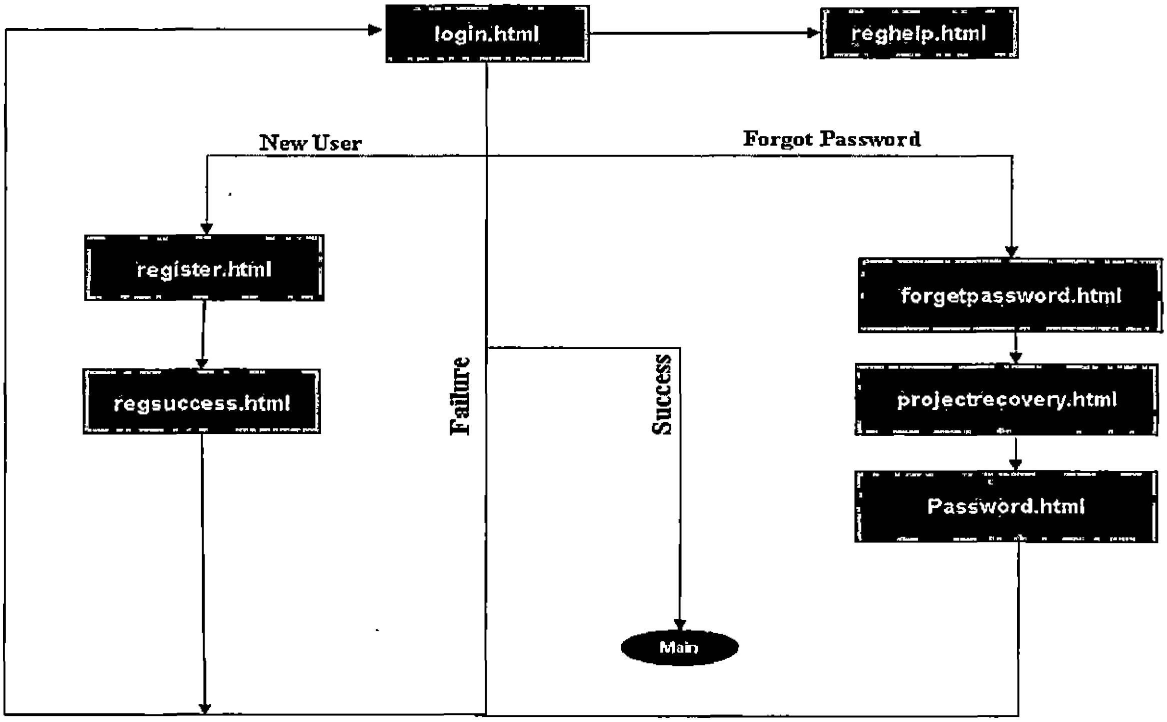


Figure 4. Login Page Navigation

2.1.1.2 Navigation Chart For Main Module. The main module plays the key role in this project. This serves as a personalized page for the user. This page will be served according to the user's choice like layout and choice of interest that they have chosen at the time of registration. The term layout refers to the page design and the choice of interest refers to the functionalities in the main page. Both layout and choice of interest are plugable. This feature is provided to ensure high degree of reusability to the project.

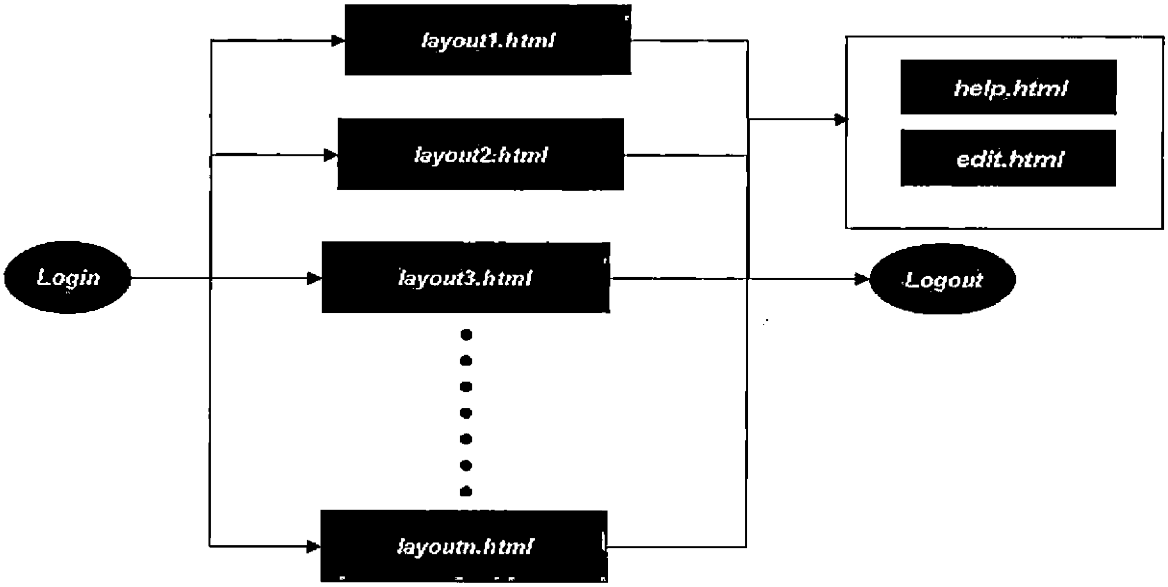


Figure 5. Main Page Navigation

2.1.1.3 Navigation Chart For Logout Module. This module is responsible for invalidating the session.

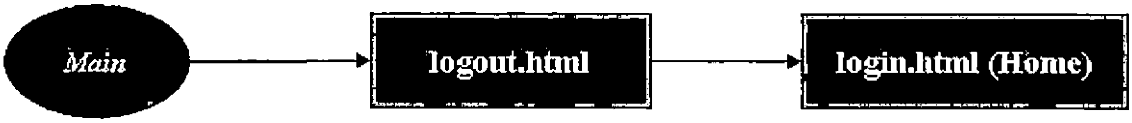
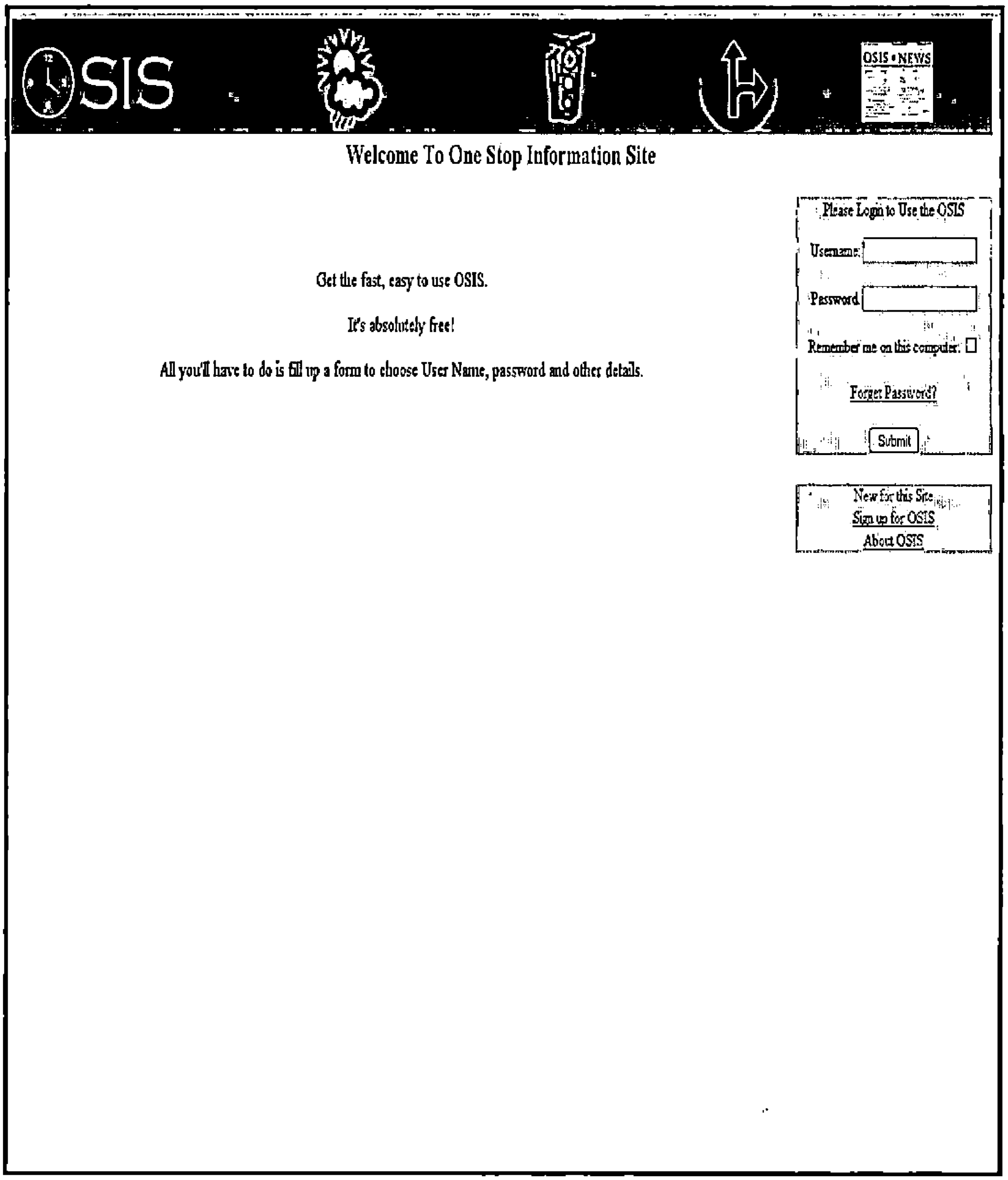


Figure 6. Logout Page Navigation

2.1.2 Page Descriptions

This part of the document contains detailed descriptions of each screen design in OSIS. Unless otherwise noted, no graphic treatment has occurred.

2.1.2.1 Description. The Login Page is used to login to the One Stop Information Site. The following table describes each element or 'item' on the page, top-to-bottom, and left-to-right.



Welcome To One Stop Information Site

Get the fast, easy to use OSIS.

It's absolutely free!

All you'll have to do is fill up a form to choose User Name, password and other details.

Please Login to Use the OSIS

Username:

Password

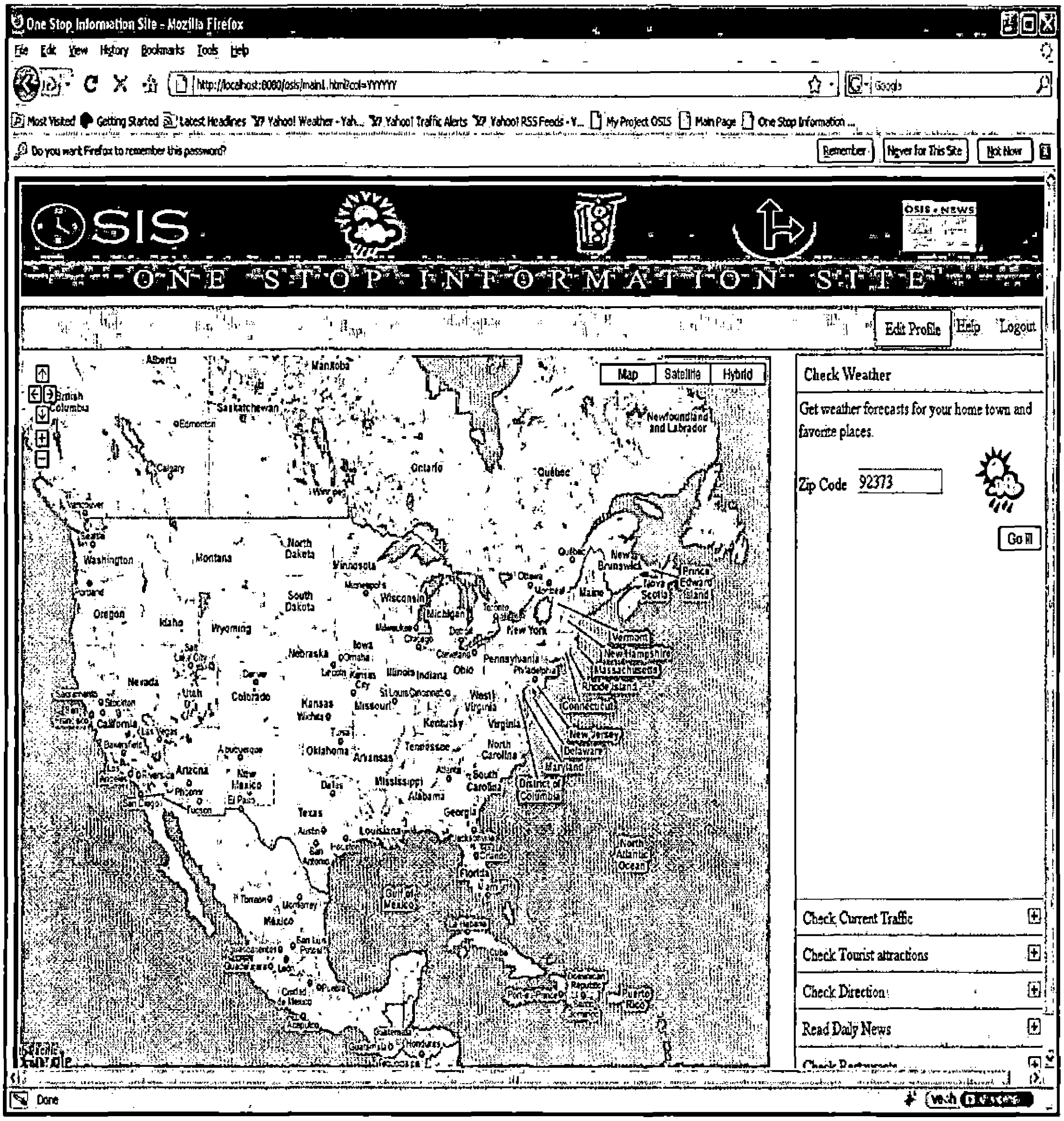
Remember me on this computer: □

Figure 7. Login Page Screen Shot

Table 1. Login Table

Page Item	Description
Username	Enter your user-name. An example user-name is kabilan@gmail.com
Password	Enter your password. An example password is heisgt32
Forgot password?	Select this link if you have forgotten your password. This link directs you to new window, were you will be asked for your Username, On submission of correct user name you will be provided with your password in a new window.
Login Button	Select this button after you have entered your user-name and password to log in to the One Stop Information Site.
Sign up for OSIS	If you are a new user select this button to create an OSIS account
About OSIS	Select this option to know details about the OSIS site providers

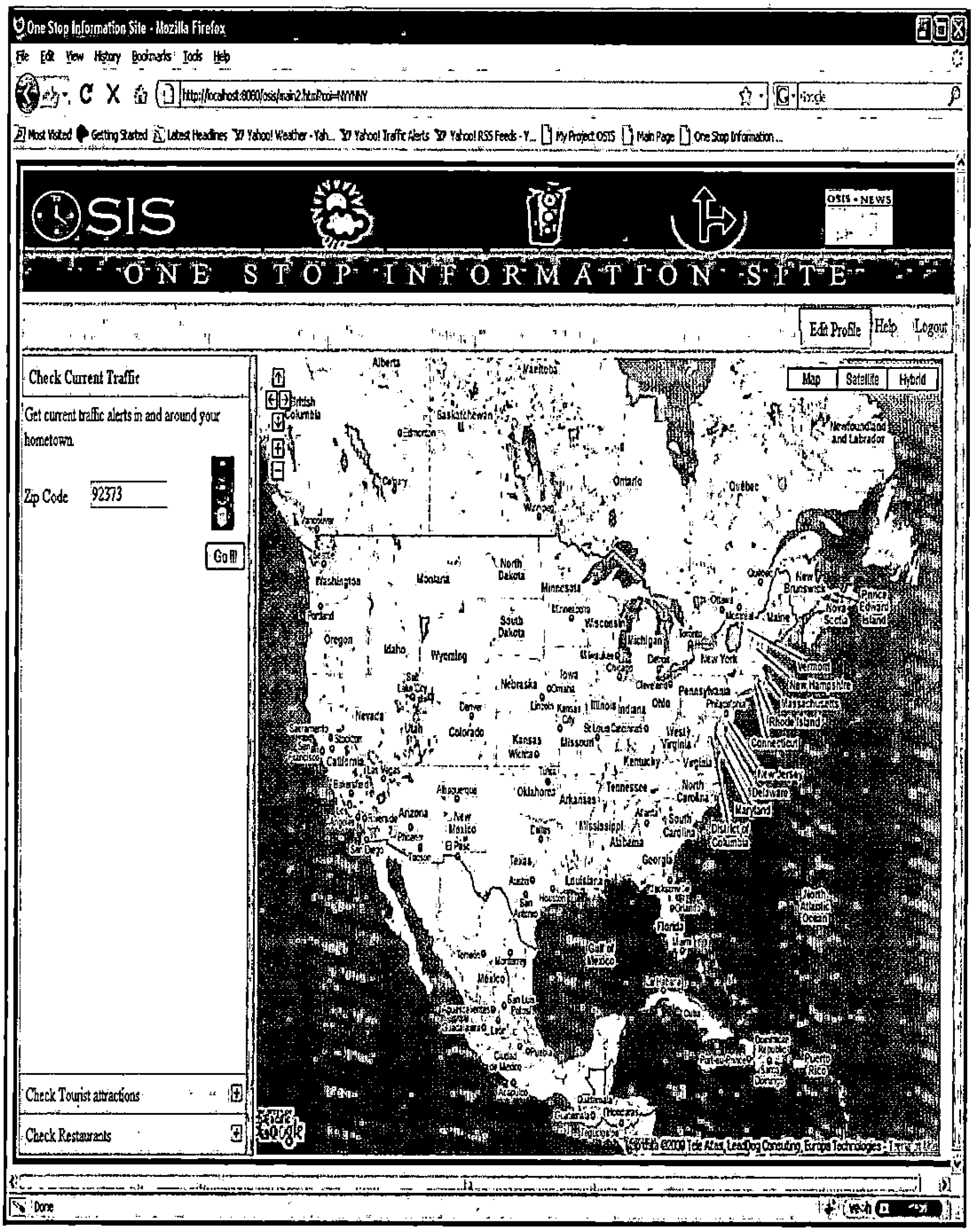
2.1.2.2 Description. The main page is reached by submitting your correct OSIS Username and Password in the login page.


One Stop Information Site - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:8080/osis/main1.html?col=YYYYY
Most Visited Getting Started Latest Headlines Yahoo! Weather - Yah... Yahoo! Traffic Alerts Yahoo! RSS Feeds - Y... My Project OSIS Main Page One Stop Information ...
Do you want Firefox to remember this password?
Remember
Never for This Site
Not Now
OSIS
OSIS NEWS
ONE STOP INFORMATION SITE
Edit Profile
Help
Logout
Map
Satellite
Hybrid
Check Weather
Zip Code
Go !!
Check Current Traffic
Check Tourist attractions
Check Directions
Read Daily News
Done

Get weather forecasts for your home town and favorite places.

Zip Code 92373

Figure 8. Main Page (Layout 1)



Check Current Traffic

Get current traffic alerts in and around your hometown.

Zip Code 92373

Go!!

Check Tourist attractions

Check Restaurants

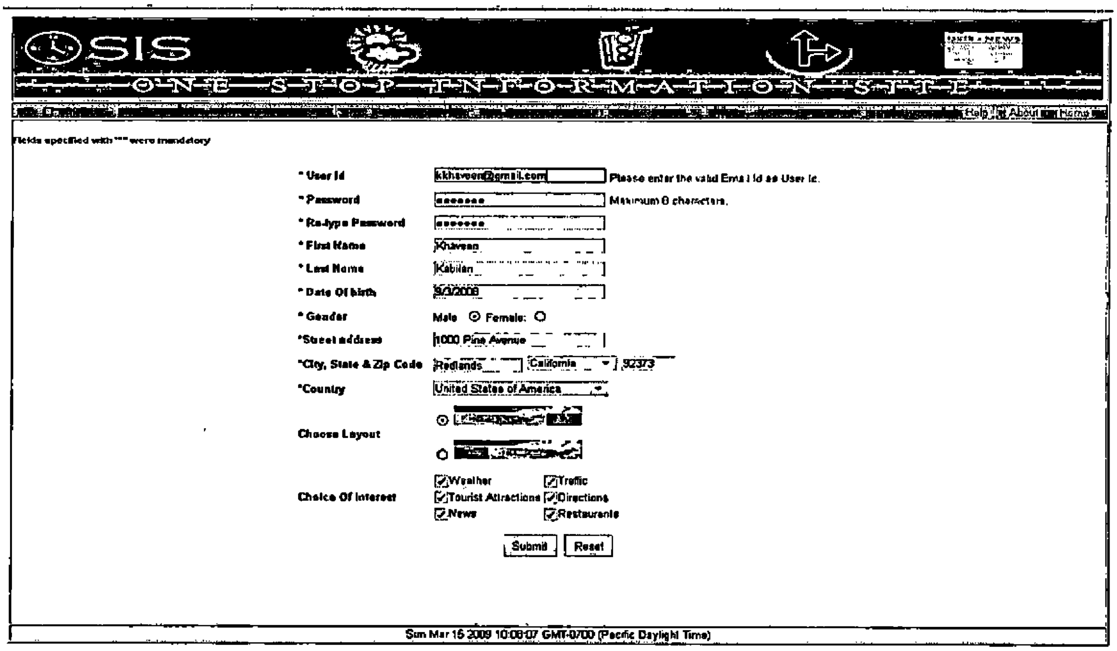
Figure 9. Main Page (Layout2)

Table 2. Main Module Table

Page Item	Description
Check Weather	Clicking this option, the user is able to get the current weather for his place of interest.
Check traffic alerts	Choosing this option, the user is able to get the deals about current traffic in highways around his place of interest.
Read daily News	This option gives user access to Local, National and International news from his choice of News.
Check Direction	User is provided with the route from his place to his choice of destination.
Check Tourist attractions	This option provide user with sight seeing details in and around his place of interest.
Check Restaurant	This option provide user with list of restaurants along with its geographic location.
editProfile	User can edit the details provided during registration.

Help	User can get help regarding the usage of OSIS by clicking this option.
About	By choosing this option the user can view the information about OSIS.
Logout	Logout button is used to exit the OSIS site.

2.1.2.3 Description. The registration page is reached by choosing sign in for OSIS link in the login page.



Fields specified with "*" were mandatory

* User Id: kkhaveen@gmail.com — Please enter the valid Email Id as User Id.

* Password: ••••••• — Maximum 8 characters.

* Re-type Password: •••••••

* First Name: Khaveen

* Last Name: Kabilan

* Date Of birth: 9/3/2008

* Gender: Male ● Female: ○

*Street address: 1000 Pine Avenue

*City, State & Zip Code: Redlands | California | 92373

*Country: United States of America

Choose Layout: ● ○

Choice Of Interest: ☑ Weather ☑ Traffic ☑ Tourist Attractions ☑ Directions ☑ News ☑ Restaurants

Submit | Reset

Sun Mar 15 2009 10:08:07 GMT-0700 (Pacific Daylight Time)

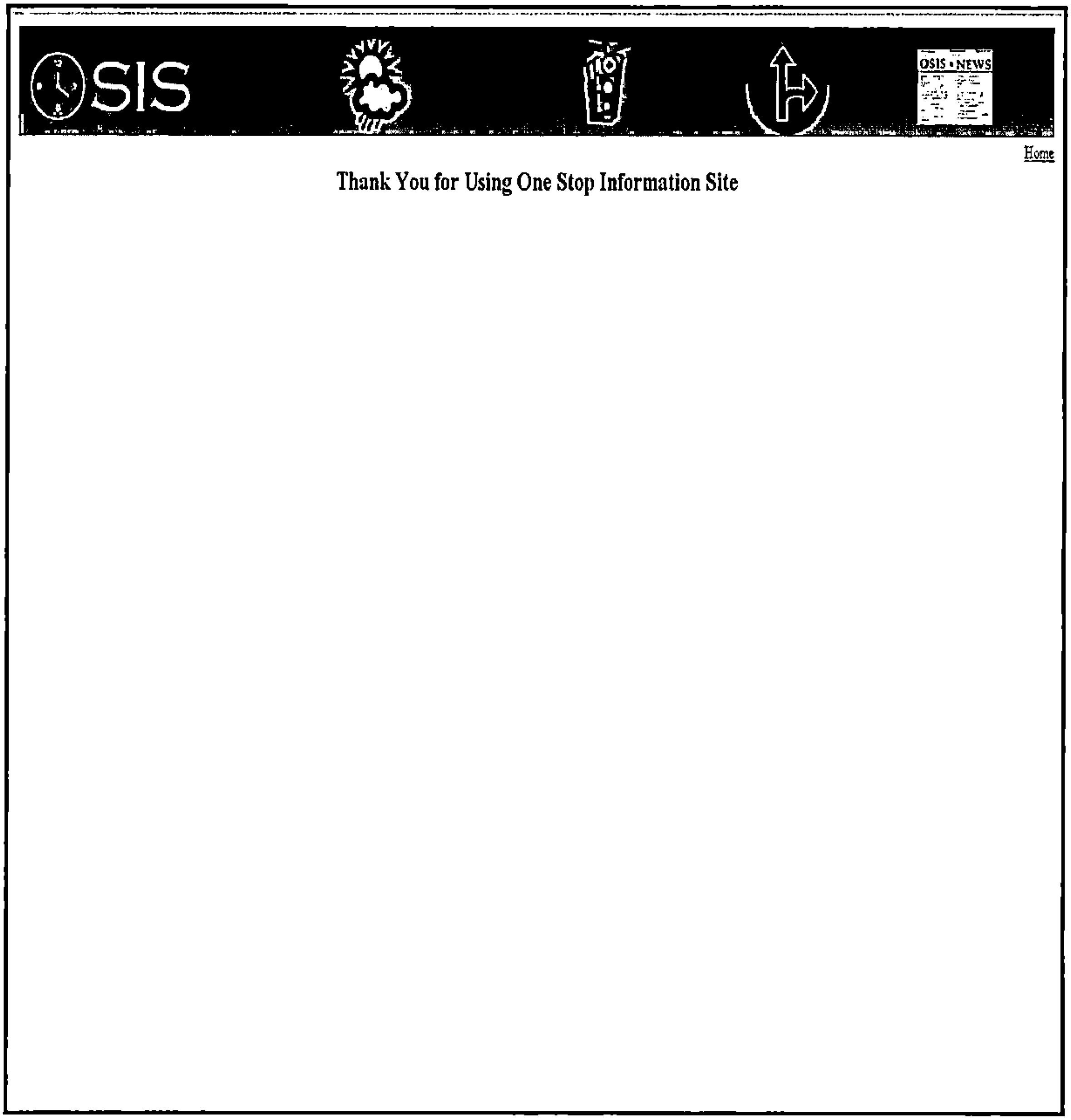
Figure 10. Registration Page

Table 3. Registration Module Table

Page item	Description
UserId	User chooses userid for access OSIS. Email address is used for userid.
Password	User chooses password for access OSIS. Password should be of maximum 8 characters to ensure security.
Retype Password	User types his password again to confirm his selected password.
First Name	User enters his first name which is mandatory.
Last Name	User enters his last name. It is a mandatory field.
Date Of Birth	User enters his date of birth which mandatory.
Gender	User enters his gender in this field which is mandatory.
Street Address	User enters his street address in this section which is mandatory.
City, State	User enters his city and state in this

& Zip Code	field which is mandatory.
Country	User enters his country in this section which is mandatory.
Choose Layout	User sets his layout for main page.
Choice Of Interest	User choices his choice of interests like weather, traffic etc for his main page.
Help	User can get help regarding the usage of OSIS registration page by clicking this option.
About	By choosing this option user can view the information about OSIS.
Home	This button directs user to the login page of OSIS site.

2.1.2.4 Description. This page is reached by clicking the logout option in the main page.



Thank You for Using One Stop Information Site

Figure 11. Logout Page

Table 4. Logout Module Table

Page item	Description
Home	This button directs user to the login page of OSIS site.

2.2 Database Design

The objective of database in this project is to store the user details along with his personalized detail.

2.2.1 Conceptual Design

The database requirements to support the functional requirements for OSIS is reviewed and understood. Based on the analysis MySQL is chosen.

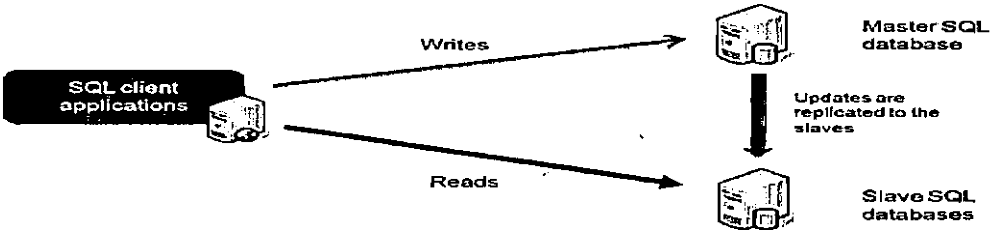


Figure 12. Database Conceptual Diagram

2.2.2 Physical Design

The file formats and access paths are selected. Relational tables for OSIS are designed as following.

Table 5. User Table

User	Data Type	Max Length
userid	varchar	30 char
layoutid	int	1
password	varchar	30 char
Retype Password	varchar	30 char
firstname	varchar	30 char
lastname	varchar	30 char
dateofbirth	date	8 bytes
gender	enum	<M, F>
address	varchar	30 char
city	varchar	30 char
state	varchar	30 char
zipcode	integer	5
country	varchar	30 char

Table 6. Layout Table

Layout	Data Type	Length
layoutid	integer	1
layoutname	varchar	30 char

2.3 Object Model Diagram

This Object Design defines the object-level design of the OSIS to be developed. It is based on the requirements collected in the requirement phase. DIA and JUDE tools are used for designing the class diagram.

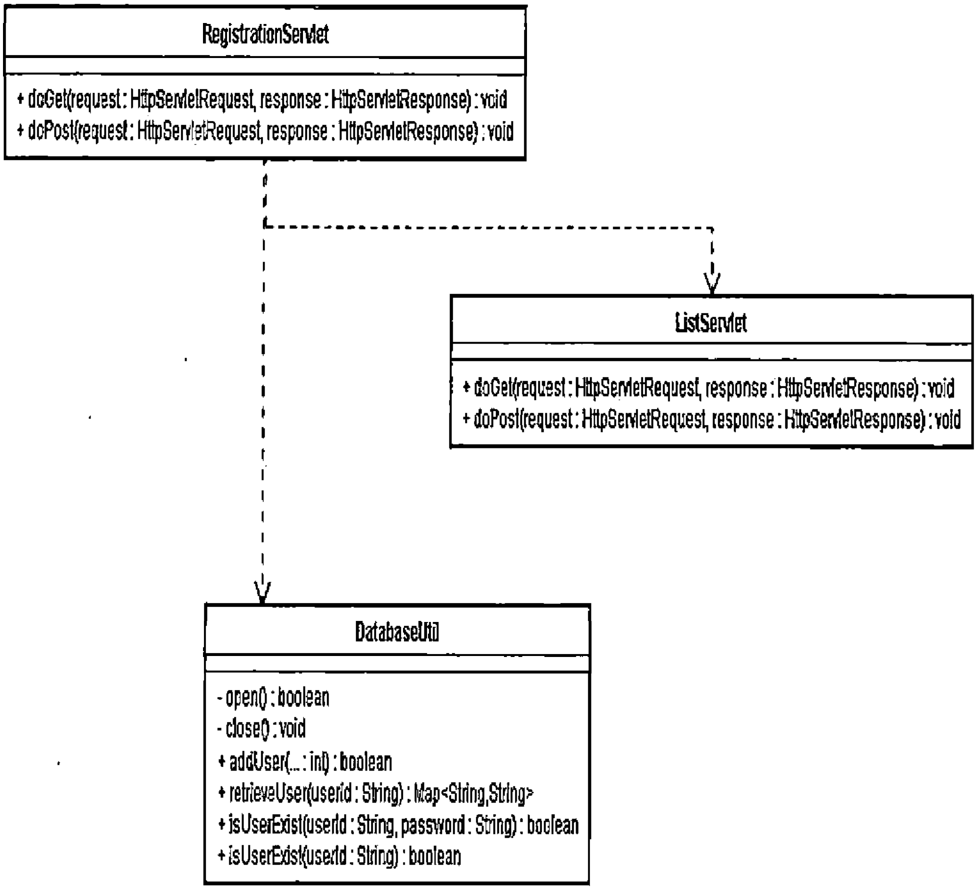

RegistrationServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
ListServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
DatabaseUtil
- open() : boolean
- close() : void
+ addUser(... : int) : boolean
+ retrieveUser(userid : String) : Map<String,String>
+ isUserExist(userId : String, password : String) : boolean
+ isUserExist(userId : String) : boolean

Figure 13. Registration Module

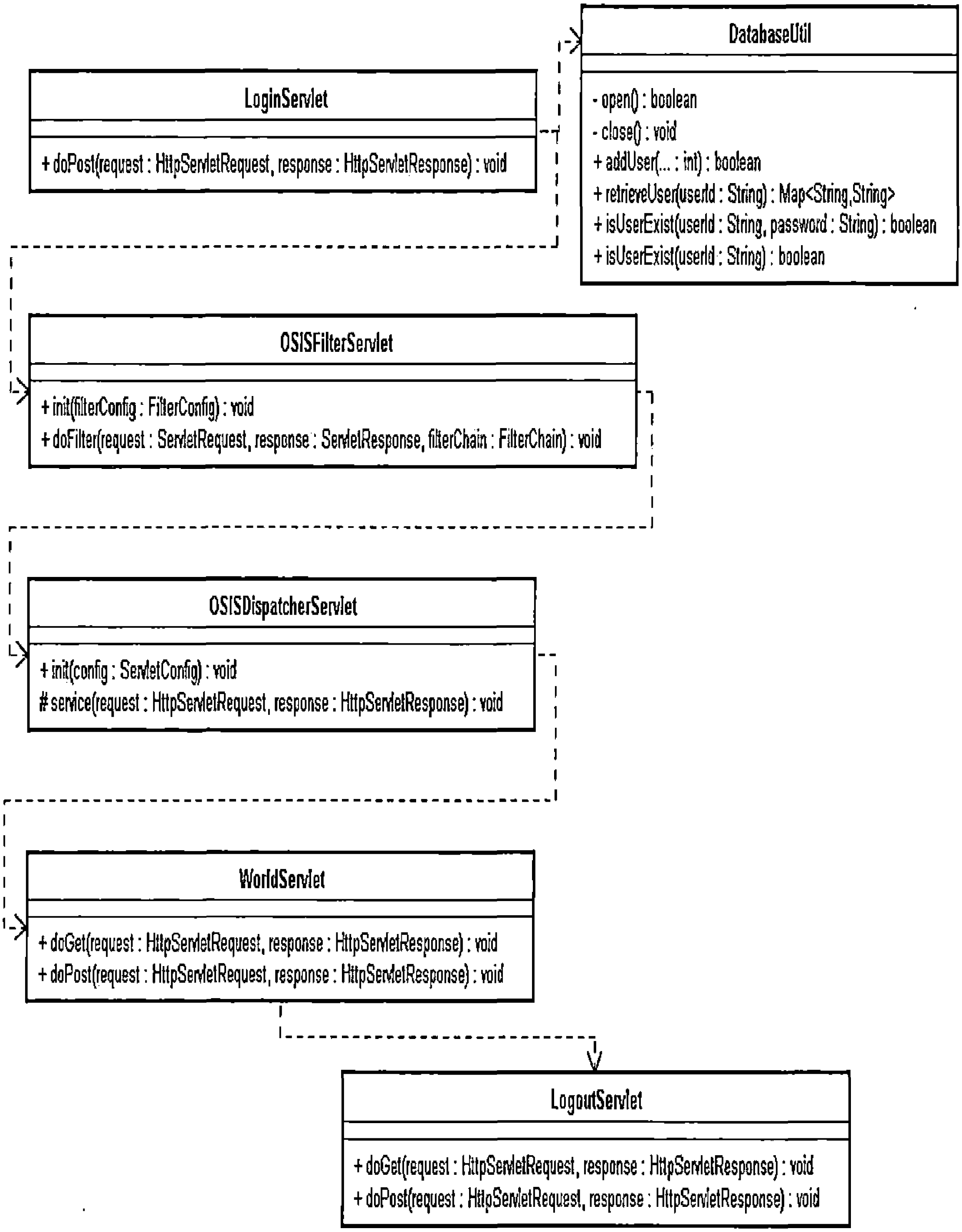

DatabaseUtil
- open() : boolean
- close() : void
+ addUser(... : int) : boolean
+ retrieveUser(userId : String) : Map<String,String>
+ isUserExist(userId : String, password : String) : boolean
+ isUserExist(userId : String) : boolean
LoginServlet
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
OSISFilterServlet
+ init(filterConfig : FilterConfig) : void
+ doFilter(request : ServletRequest, response : ServletResponse, filterChain : FilterChain) : void
OSISDispatcherServlet
+ init(config : ServletConfig) : void
# service(request : HttpServletRequest, response : HttpServletResponse) : void
WorldServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
LogoutServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void

Figure 14. Class Diagram Representing System Flow

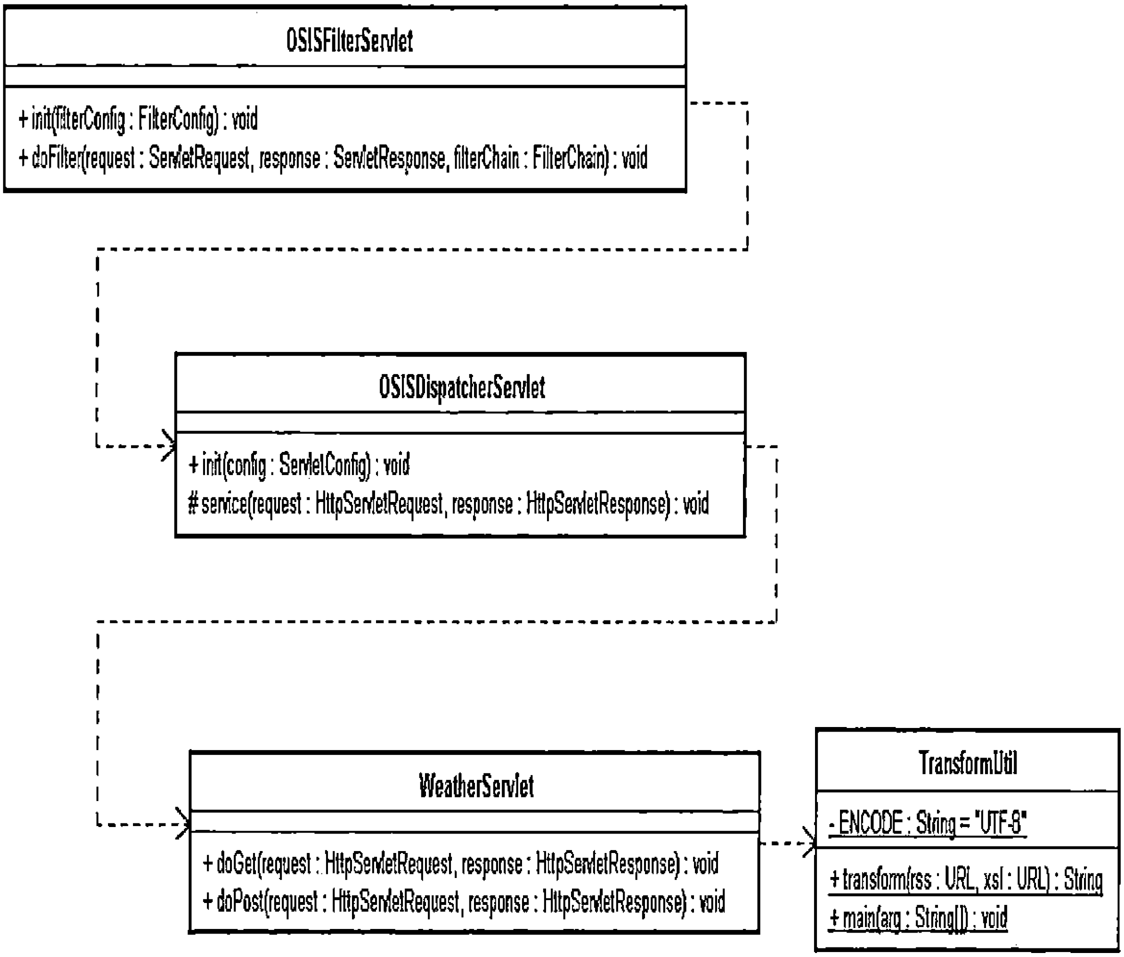

OSISFilterServlet
+ init(filterConfig : FilterConfig) : void
+ doFilter(request : ServletRequest, response : ServletResponse, filterChain : FilterChain) : void
OSISDispatcherServlet
+ init(config : ServletConfig) : void
# service(request : HttpServletRequest, response : HttpServletResponse) : void
WeatherServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
TransformUtil
- ENCODE : String = "UTF-8"
+ transform(rss : URL, xsl : URL) : String
+ main(arg : String[]) : void

Figure 15. Class Diagram for Weather Module

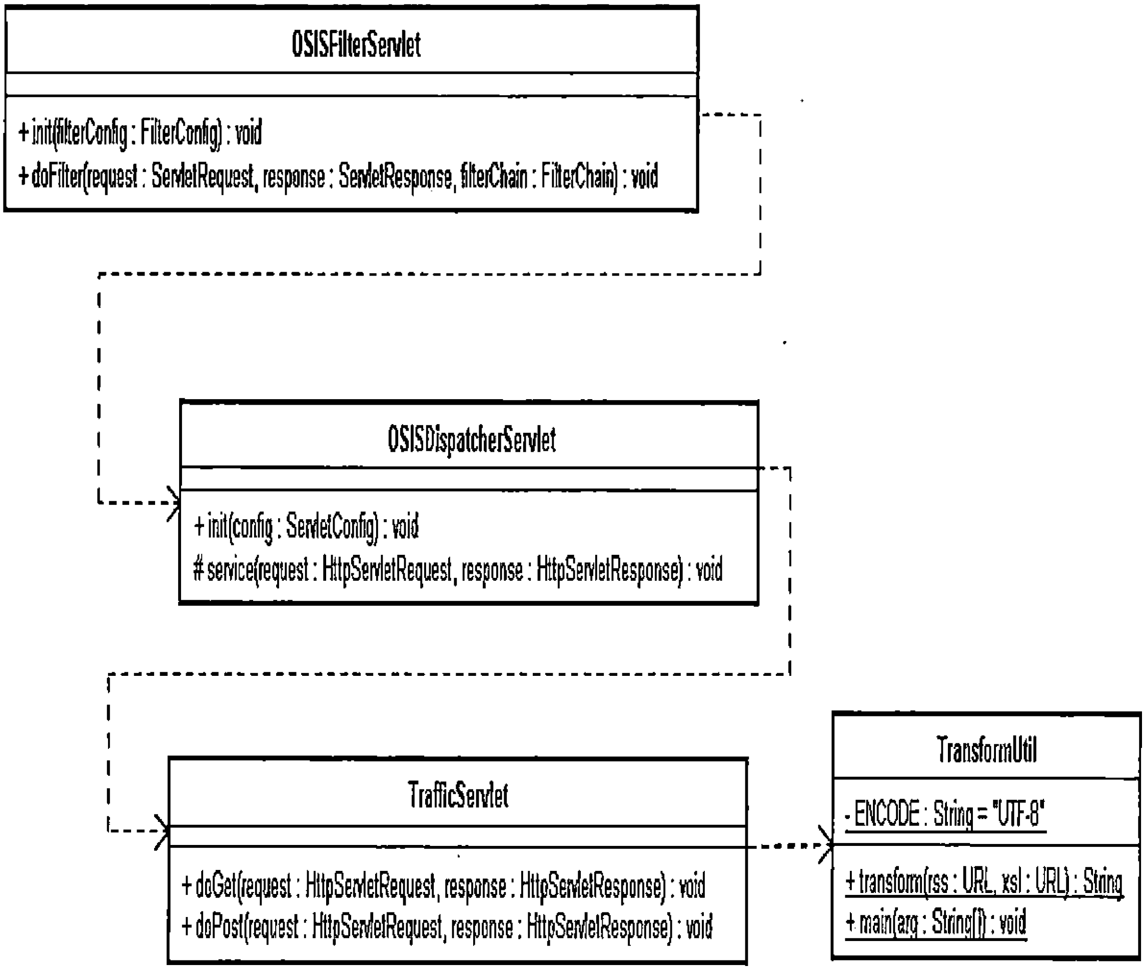

OSISFilterServlet
+ init(filterConfig : FilterConfig) : void
+ doFilter(request : ServletRequest, response : ServletResponse, filterChain : FilterChain) : void
OSISDispatcherServlet
+ init(config : ServletConfig) : void
# service(request : HttpServletRequest, response : HttpServletResponse) : void
TrafficServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
TransformUtil
- ENCODE : String = "UTF-8"
+ transform(rss : URL, xsl : URL) : String
+ main(arg : String[]) : void

Figure 16. Class Diagram for Traffic Module

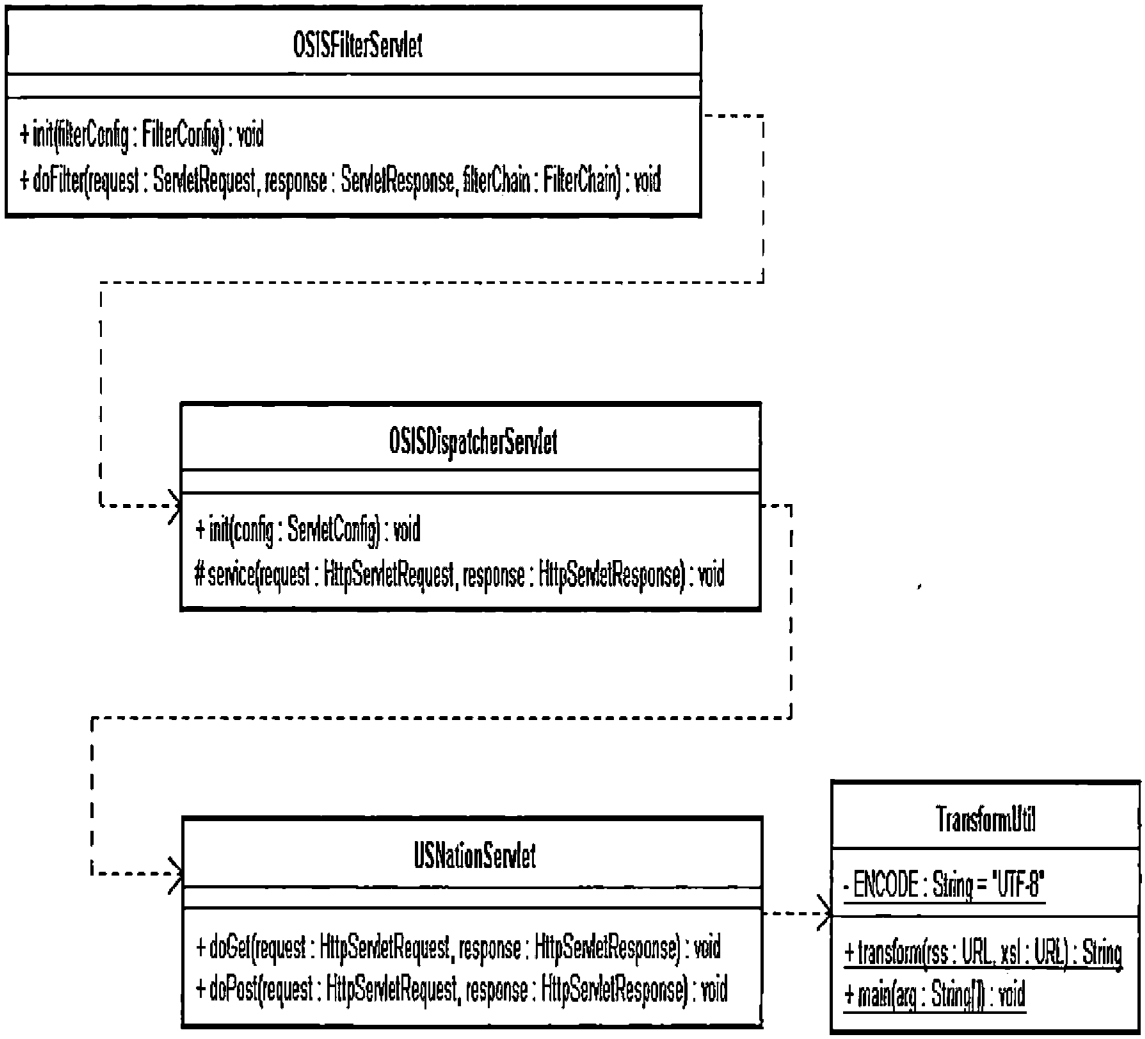

OSISFilterServlet
+ init(filterConfig : FilterConfig) : void
+ doFilter(request : ServletRequest, response : ServletResponse, filterChain : FilterChain) : void
OSISDispatcherServlet
+ init(config : ServletConfig) : void
# service(request : HttpServletRequest, response : HttpServletResponse) : void
USNationServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
TransformUtil
- ENCODE : String = "UTF-8"
+ transform(rss : URL, xsl : URL) : String
+ main(arg : String[]) : void

Figure 17. Class Diagram for United Nation Servlet

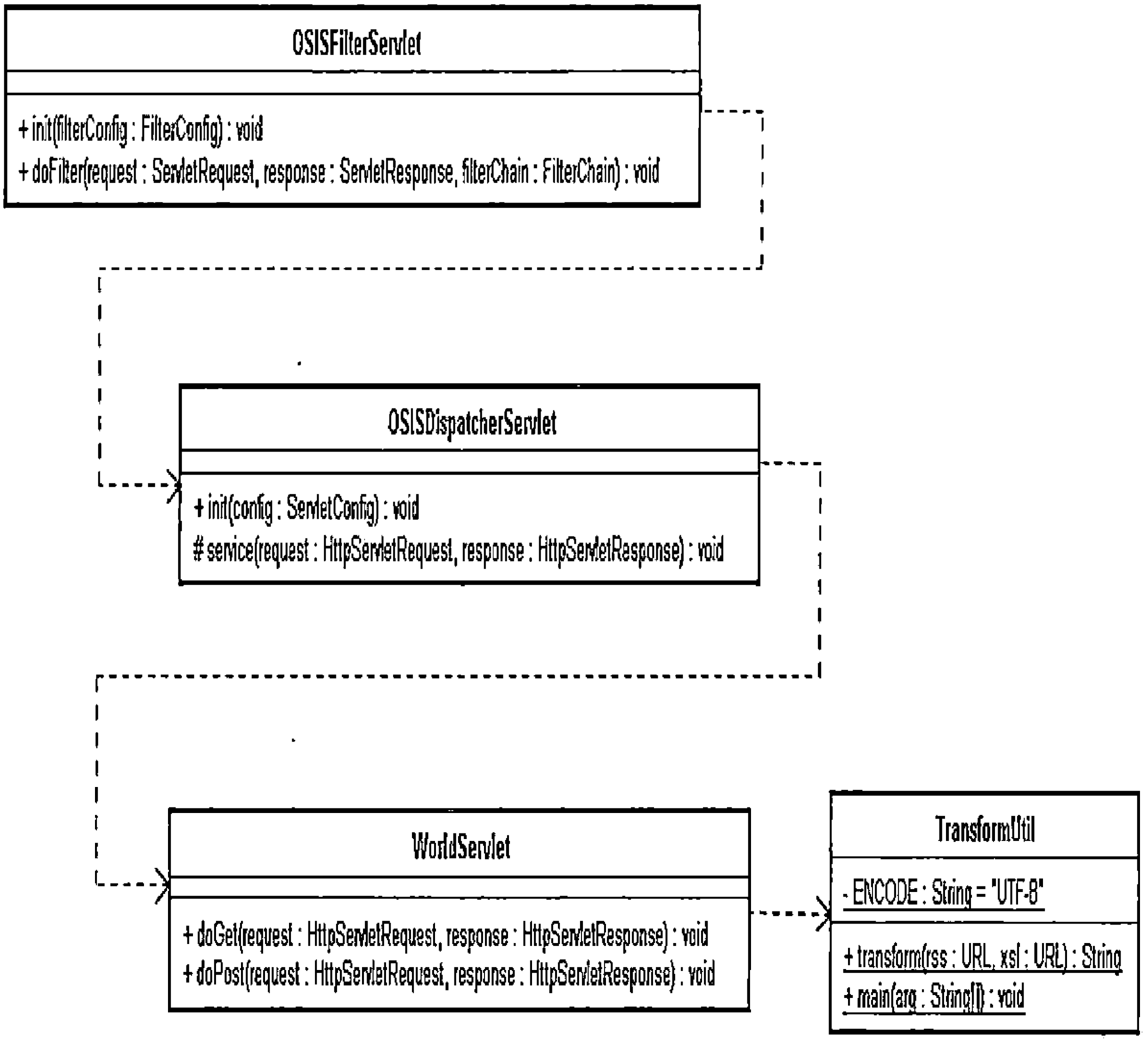

OSISFilterServlet
+ init(filterConfig : FilterConfig) : void
+ doFilter(request : ServletRequest, response : ServletResponse, filterChain : FilterChain) : void
OSISDispatcherServlet
+ init(config : ServletConfig) : void
# service(request : HttpServletRequest, response : HttpServletResponse) : void
WorldServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
TransformUtil
- ENCODE : String = "UTF-8"
+ transform(rss : URL, xsl : URL) : String
+ main(arg : String[]) : void

Figure 18. Class Diagram for World Servlet

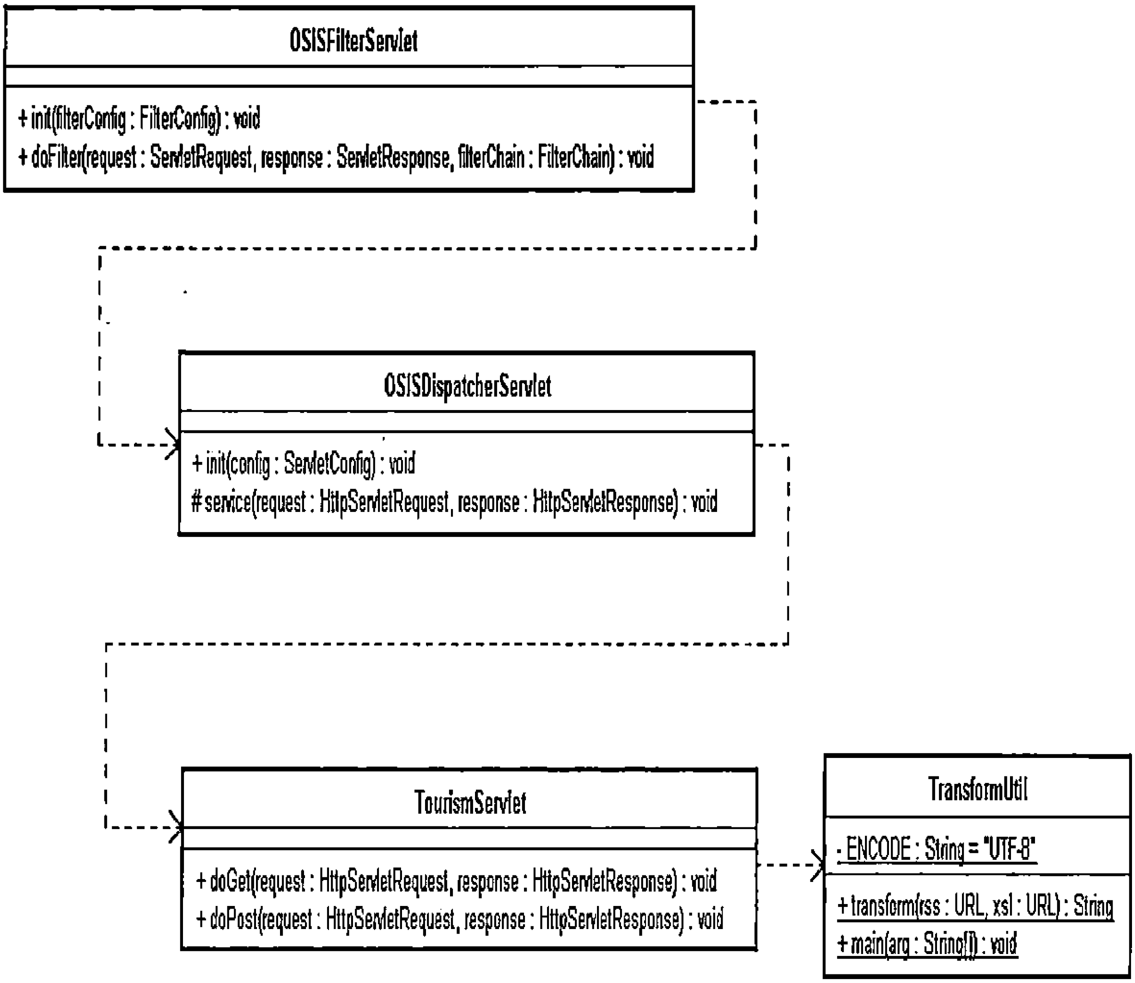

OSISFilterServlet
+ init(filterConfig : FilterConfig) : void
+ doFilter(request : ServletRequest, response : ServletResponse, filterChain : FilterChain) : void
OSISDispatcherServlet
+ init(config : ServletConfig) : void
# service(request : HttpServletRequest, response : HttpServletResponse) : void
TourismServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
TransformUtil
- ENCODE : String = "UTF-8"
+ transform(rss : URL, xsl : URL) : String
+ main(arg : String[]) : void

Figure 19. Class Diagram for Tourism Servlet

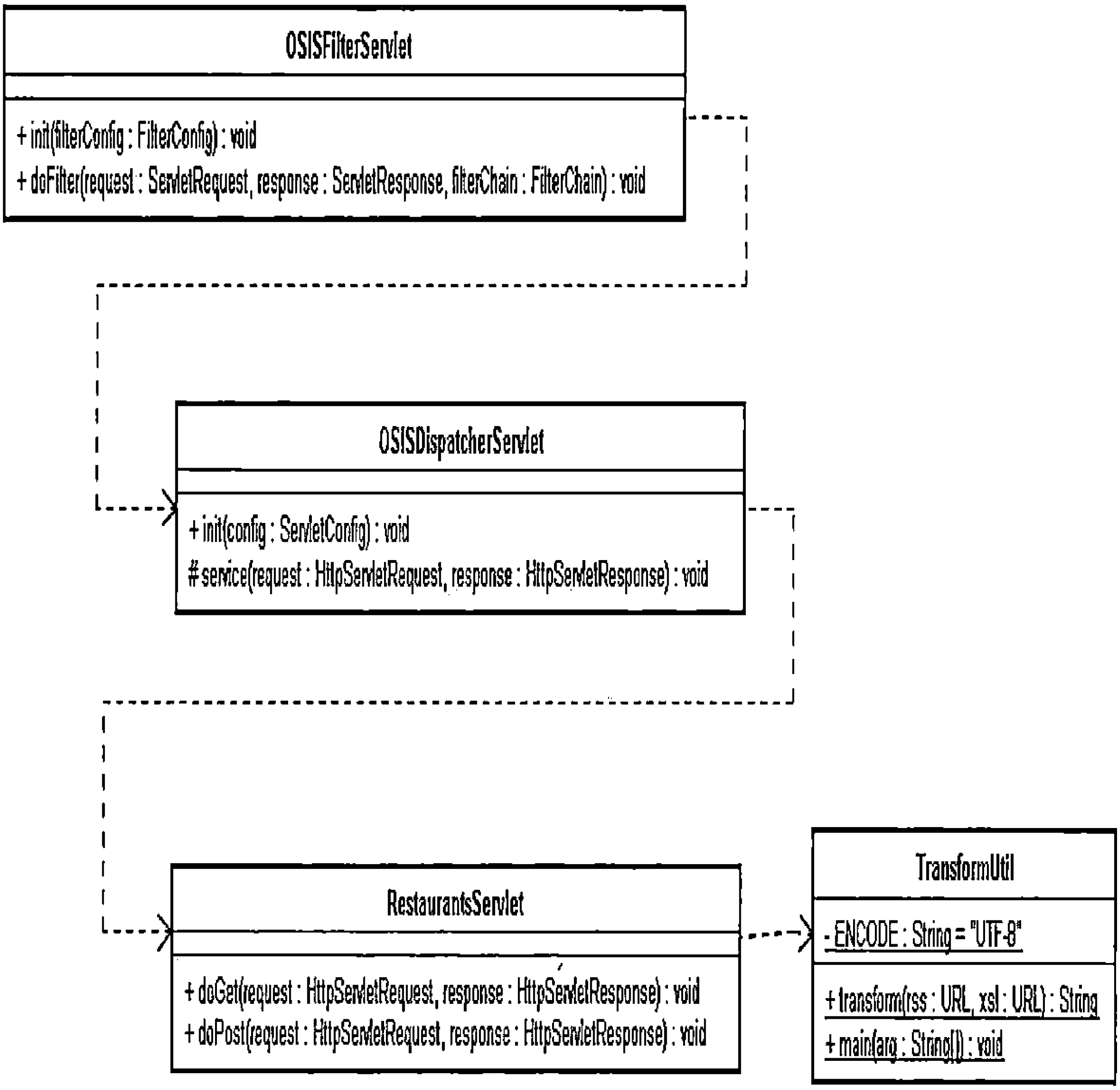

OSISFilterServlet
+ init(filterConfig : FilterConfig) : void
+ doFilter(request : ServletRequest, response : ServletResponse, filterChain : FilterChain) : void
OSISDispatcherServlet
+ init(config : ServletConfig) : void
# service(request : HttpServletRequest, response : HttpServletResponse) : void
RestaurantsServlet
+ doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+ doPost(request : HttpServletRequest, response : HttpServletResponse) : void
TransformUtil
- ENCODE : String = "UTF-8"
+ transform(rss : URL, xsl : URL) : String
+ main(arg : String[]) : void

Figure 20. Class Diagram for Restaurant Servlet

CHAPTER THREE

ONE STOP INFORMATION SITE

3.1 Design of Code

In software development, design pattern is a written document that describes a design problem, its proposed solution, and any other factors that might affect the problem or the solution. In this chapter the design pattern details are discussed.

3.2 Design Pattern

MVC pattern is followed in this project for the separation of user interface logic from business logic. This pattern is followed to achieve a clean separation between three components such as model, view and controller.

3.2.1 Model Demonstration

Model represents business logic & processing. The model manages the behavior and data of the application domain, responds to requests for information about its state usually from the view, and responds to instructions to change state usually from the controller. The model has no specific knowledge of either its controllers or its views. The system itself maintains links between model and views and notifies the views when the model changes state. The view is the piece that manages the visual display of the state represented by the model. A model can have more than one view. JDBC which is used for enabling the java program to manipulate data stored into the database falls under model. Java uses methods of its JDBC API for querying and updating data in the database.

3.2.2 View Demonstration

View represent user interface. In my project View part is played by XSL which parses the XML RSS feed and displays the output in HTML format. The view represents or serves up the data manipulated by the controller from the model. The view is responsible for mapping graphics onto a device. The view renders the contents of a model. It accesses enterprise data through the model and specifies how that data should be presented. The view manages the graphical and/or textual output to the portion of the bitmapped display that is allocated to its application.

3.2.3 Controller Demonstration

The controller interprets the mouse and keyboard inputs from the user, informing the model and/or the view to change as appropriate. Controller part is played by the Servlet. The controller also sends data to the view. The controller is the piece that manages user interaction with the model. It provides the mechanism by which changes are made to the state of the model.

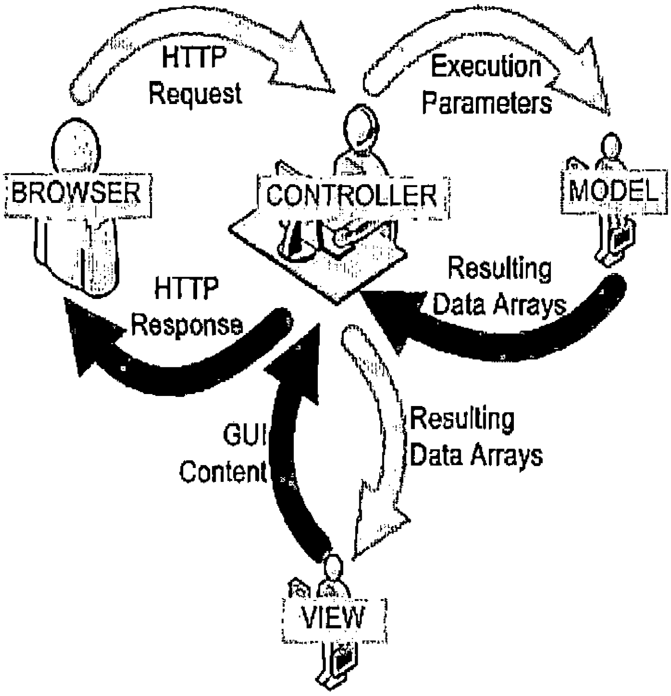


Figure 21. Model View Controller Relationship (1)

3.2.3.1 Relating To OSIS. In this project each component references one or more controller that is Servlet and one view that is XSL style sheet and one model which has database in it.

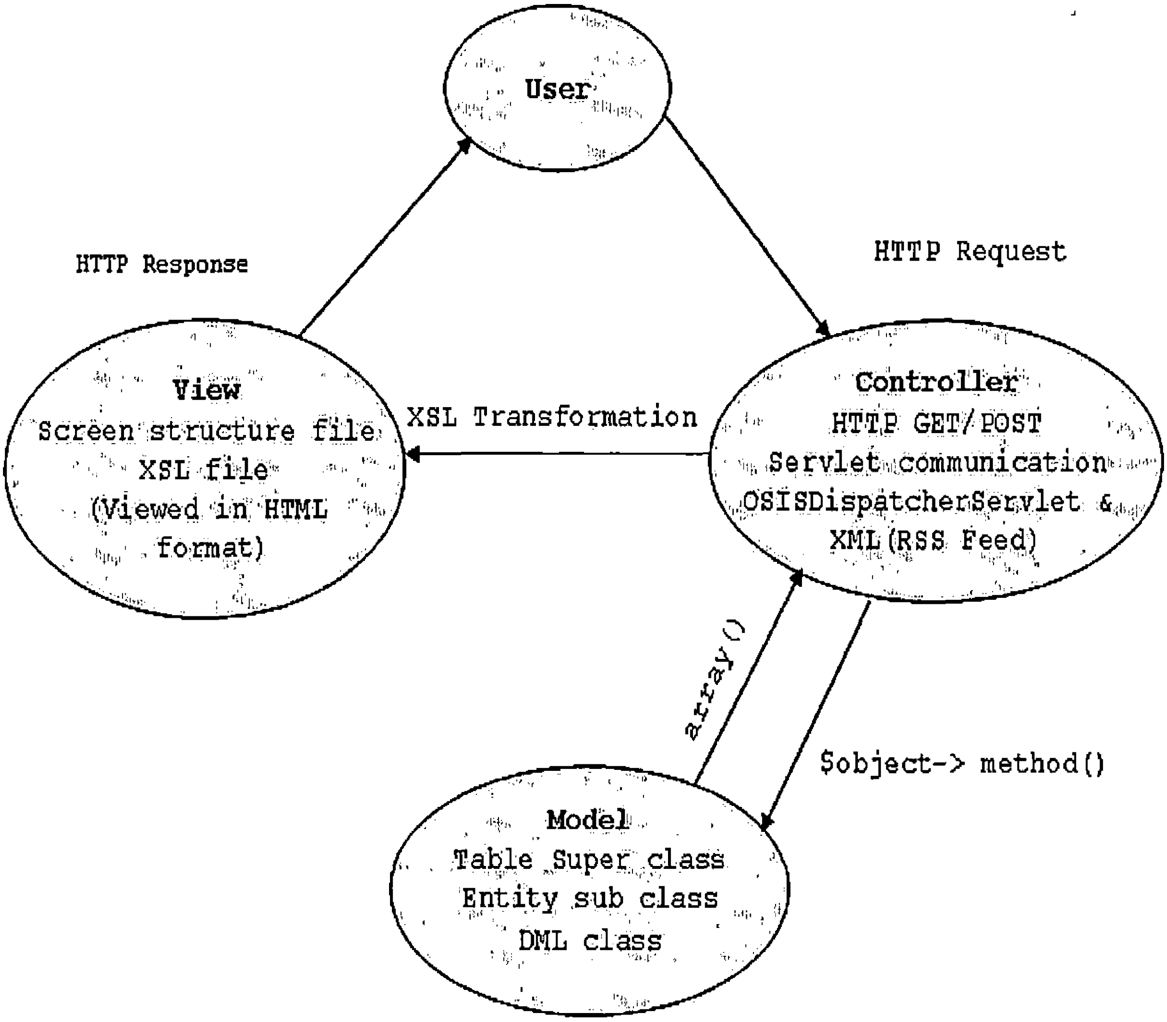


Figure 22. Model View Controller Relationship (2)

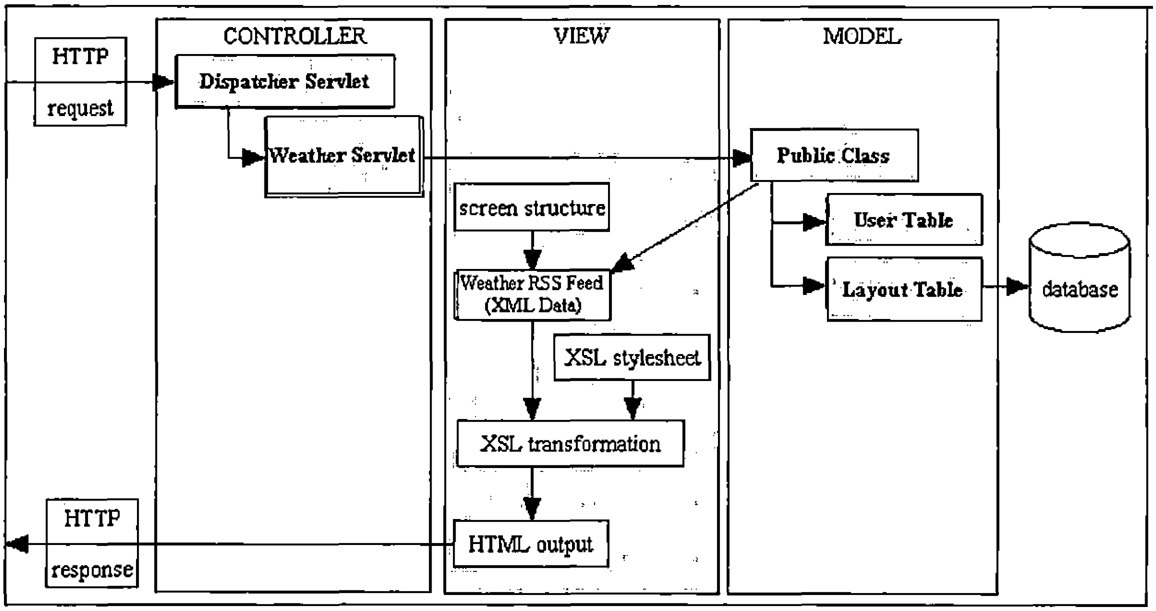


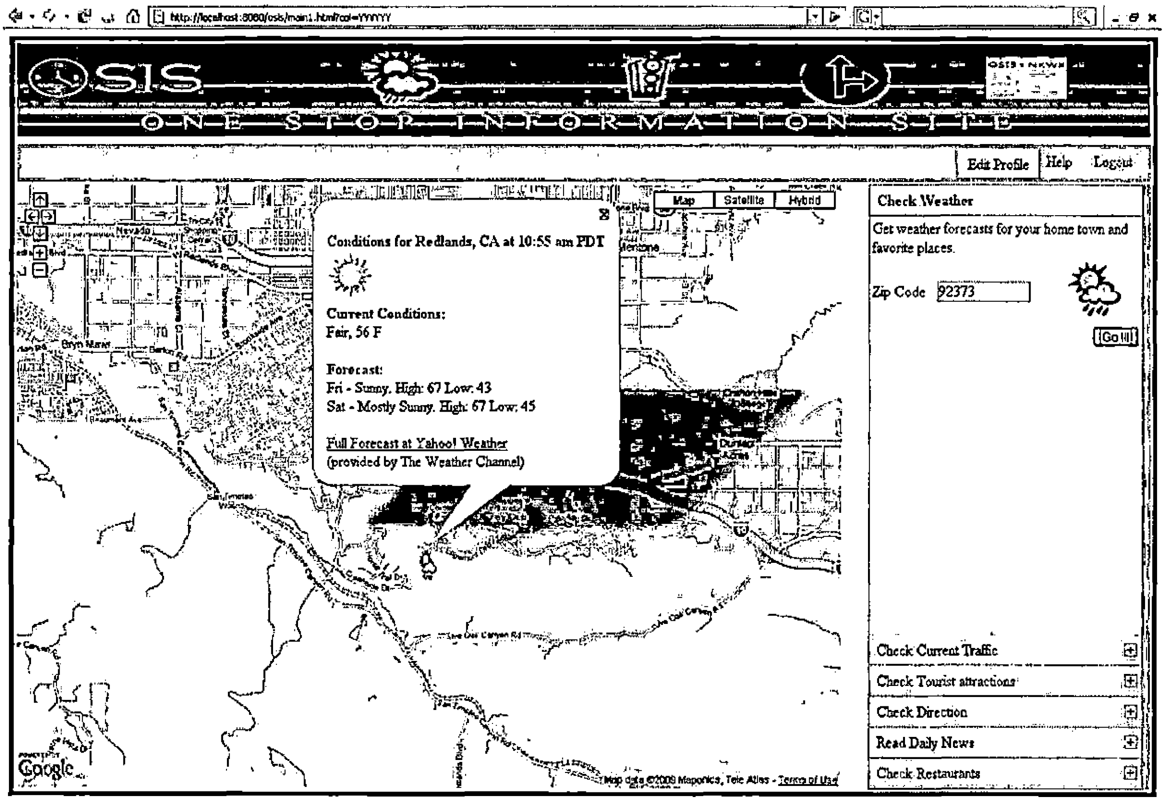
Figure 23. Design Pattern for Weather Module

3.3 Functionalities and Screen Shots

This project has six modules in it, such as check weather, check traffic alert, read daily news, check direction, check tourist attractions and check restaurants. Brief descriptions and screenshots of these modules were as following.

3.3.1 Check Weather

This module displays the current weather condition for the specified zip code. User's request (zip code) is send to the Servlet by Ajax's weatherAjax function. The request will be sent first to the main Servlet that is the dispatcher Servlet which forwards the requests to the corresponding weather Servlet. TransformUtil class transforms the file path for the XSL file that is weather.xsl to the weatherServlet. The weather Servlet has the URL for the weather RSS feed (yahoo RSS feed). XSL file is designed is a way to parse the XML file of that RSS feed to produce HTML output. Google map object is used to show the output in the map. GIMP software is used to design the images for this module.



Conditions for Redlands, CA at 10:55 am PDT

Current Conditions:
Fair, 56 F

Forecast:
Fri - Sunny. High: 67 Low: 43
Sat - Mostly Sunny. High: 67 Low: 45

Full Forecast at Yahoo! Weather
(provided by The Weather Channel)

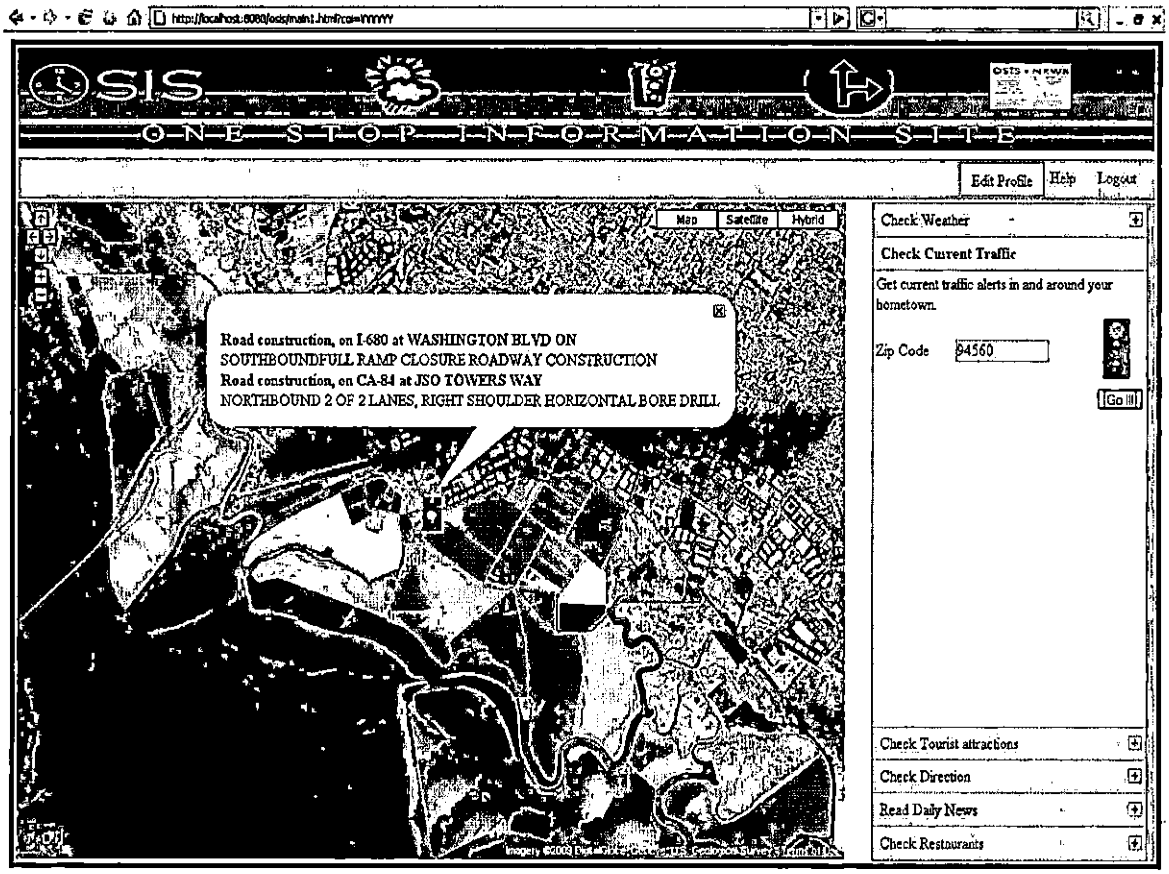
Get weather forecasts for your home town and favorite places.

Zip Code 92373

Figure 24. Check Weather Module Screen Shot

3.3.2 Check Traffic Alerts

This module displays the current traffic alerts for the specified zip code. User's request (zip code) is send to the Servlet by Ajax's trafficAjax function. The request will be sent first to the main Servlet that is the dispatcher Servlet which forwards the requests to the corresponding traffic Servlet. TransformUtil class transforms the file path for the XSL file that is traffic.xsl to the trafficServlet. The traffic Servlet has the URL for the traffic RSS feed (yahoo RSS feed). XSL file is designed is a way to parse the XML file of that RSS feed to produce HTML output. Google map object is used to show the output in the map. GIMP software is used to design the images for this module.



Road construction, on I-680 at WASHINGTON BLVD ON SOUTHBOUNDFULL RAMP CLOSURE ROADWAY CONSTRUCTION

Road construction, on CA-84 at JSO TOWERS WAY NORTHBOUND 2 OF 2 LANES, RIGHT SHOULDER HORIZONTAL BORE DRILL

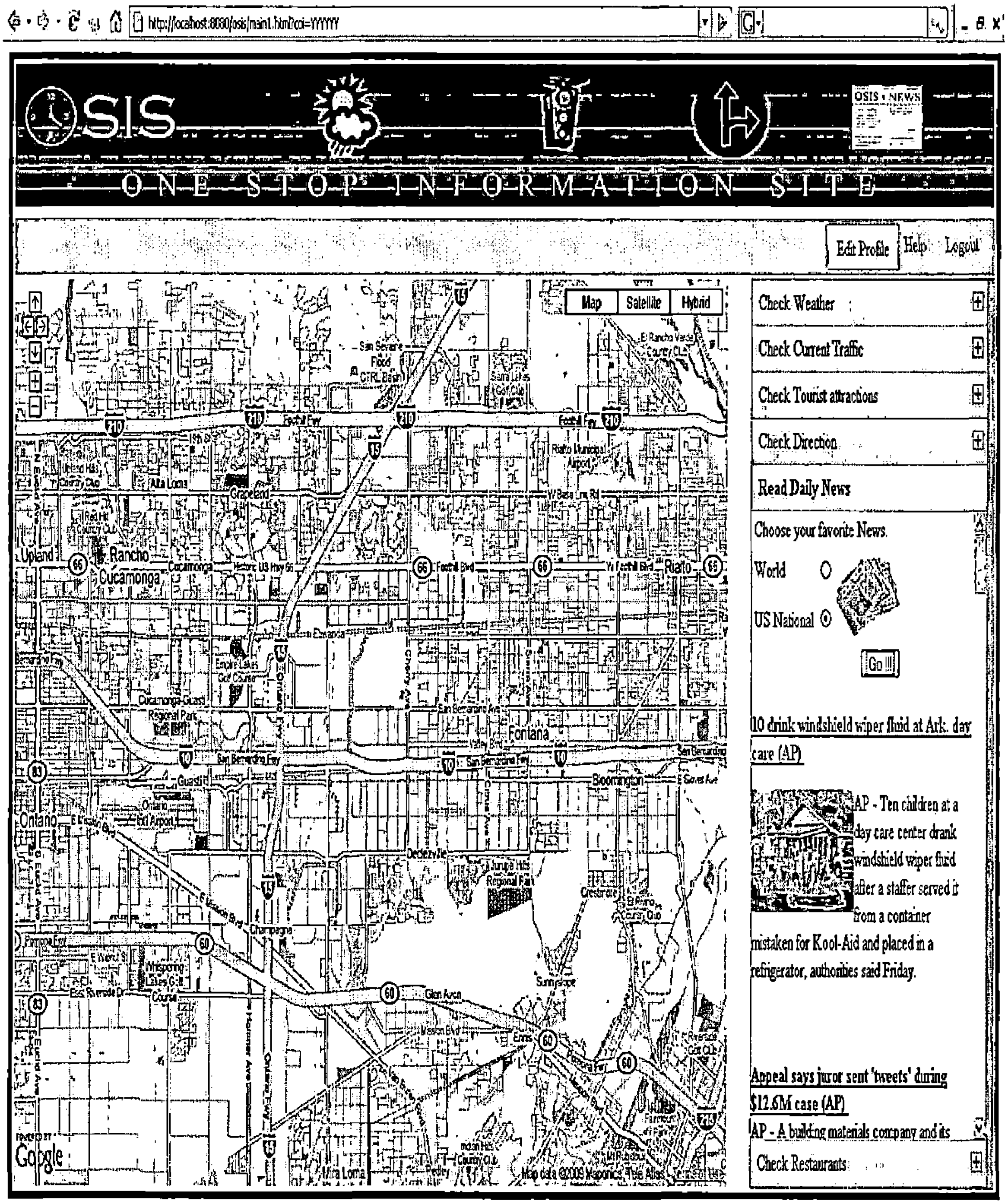
Get current traffic alerts in and around your hometown.

Zip Code 94560

Figure 25. Traffic Alert Module Screen Shot

3.3.3 Read Daily News

This module displays the world news or US News based on user's request. User's request is send to the Servlet by Ajax's newsAjax function. The request will be sent first to the main Servlet that is the dispatcher Servlet which forwards the requests to the corresponding news Servlets that is worldServet or usnationServlet. TransformUtil class transforms the file path for the XSL file that is usnation.xsl or world.xsl to the newsServlet based on the radio button selection. The news Servlet has the URL for the news RSS feed (yahoo RSS feed). XSL file is designed is a way to parse the XML file of that RSS feed to produce HTML output. XSL file is designed in a way to display the news in new window.



Choose your favorite News.

World ○

US National ●

Go !!!

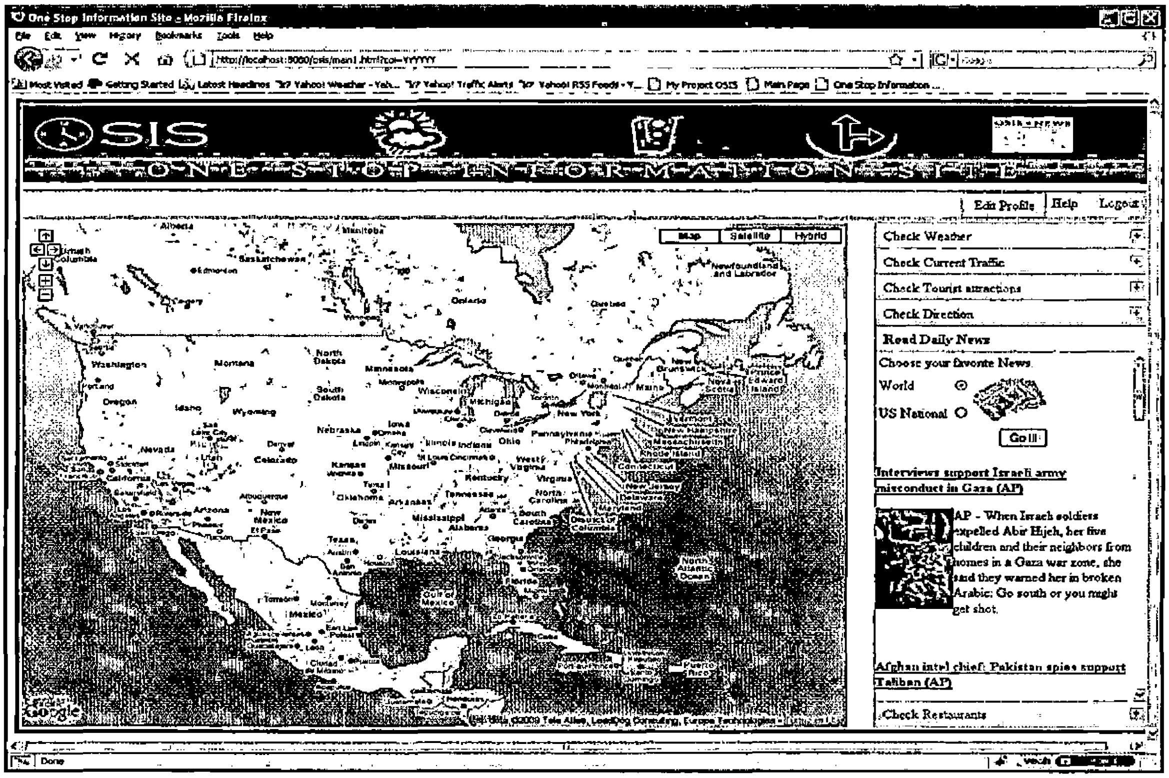
10 drink windshield wiper fluid at Ark. day care (AP)

AP - Ten children at a day care center drank windshield wiper fluid after a staffer served it from a container mistaken for Kool-Aid and placed in a refrigerator, authorities said Friday.

Appeal says juror sent 'tweets' during $12.6M case (AP)

AP - A building materials company and its

Figure 26. News Module Screen Shot (US News)



Interviews support Israeli army misconduct in Gaza (AP)

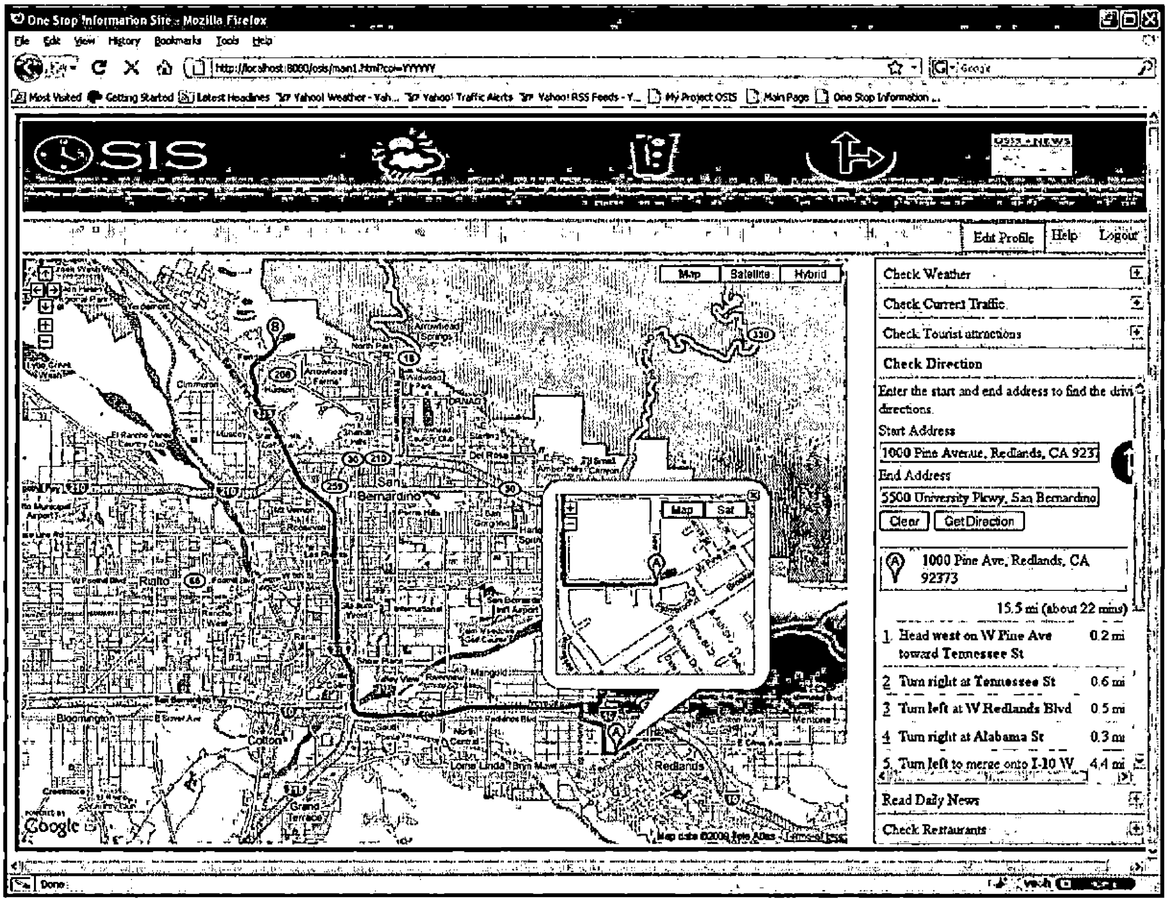
AP - When Israeli soldiers expelled Abir Hijeh, her five children and their neighbors from homes in a Gaza war zone, she said they warned her in broken Arabic: Go south or you might get shot.

Afghan intel chief: Pakistan spies support Taliban (AP)

Figure 27. News Module Screen Shot (World News)

3.3.4 Check Direction

This module displays the direction between the specified "From" and "To" address. Google map's Gdirection object is used to show the direction in the map.


One Stop Information Site - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:8080/osis/main1.html?co=YYYYY
Most Visited Getting Started Latest Headlines Yahoo! Weather - Yah... Yahoo! Traffic Alerts Yahoo! RSS Feeds - Y... My Project OSIS Main Page One Stop Information ...
OSIS
OSIS NEWS
Edit Profile
Help
Logout
Map
Satellite
Hybrid
Check Weather
Check Current Traffic
Check Tourist attractions
Check Direction
Clear
GetDirection
Read Daily News
Check Restaurants
Done

Enter the start and end address to find the drivi directions.

Start Address

1000 Pine Avenue, Redlands, CA 923

End Address

5500 University Pkwy, San Bernardino

1000 Pine Ave, Redlands, CA 92373

15.5 mi (about 22 mins)

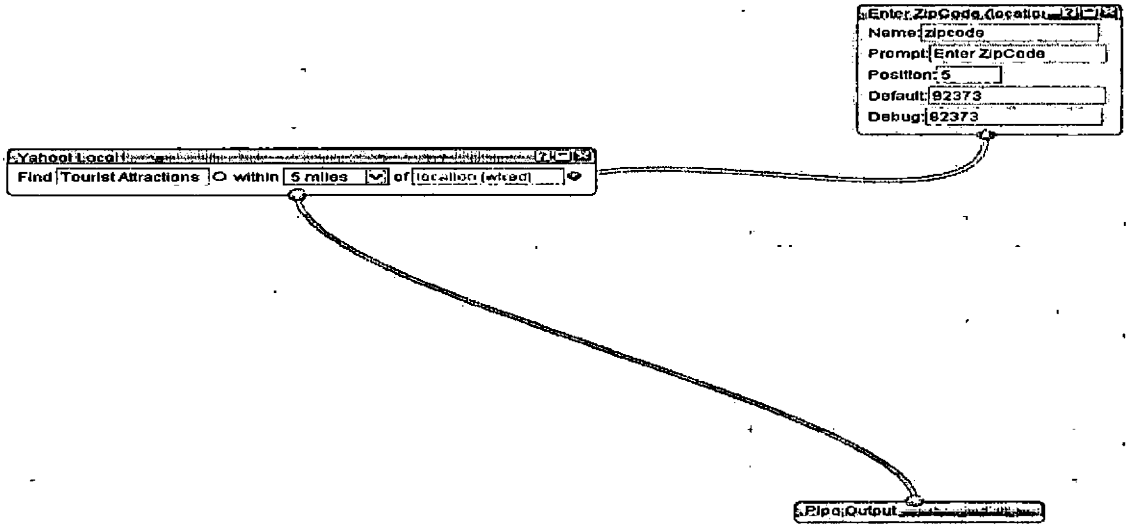
1. Head west on W Pine Ave toward Tennessee St 0.2 mi
2. Turn right at Tennessee St 0.6 mi
3. Turn left at W Redlands Blvd 0.5 mi
4. Turn right at Alabama St 0.3 mi
5. Turn left to merge onto I-10 W 4.4 mi

Figure 28. Check Direction Module Screen Shot

3.3.5 Check Tourist Attractions

This module displays the tourist attractions for the specified zip code. User's request (zip code) is sent to the Servlet by Ajax's tourismAjax function. The request will be send first to the main Servlet that is the dispatcher Servlet which forwards the requests to the corresponding tourism Servlet. The traffic Servlet has the

URL for the tourism pipe and the file path for the XSL file that is tourism.xsl is transformed to it by transformUtil. Tourism pipe is designed using yahoo pipes. XSL file is designed in a way to produce HTML output. It is designed in a way to display the attractions in new window. Google map object is used to show the output in the map. GIMP software is used to design the images for this module.



Enter ZipCode (location

Name: zipcode

Prompt: Enter ZipCode

Position: 5

Default: 92373

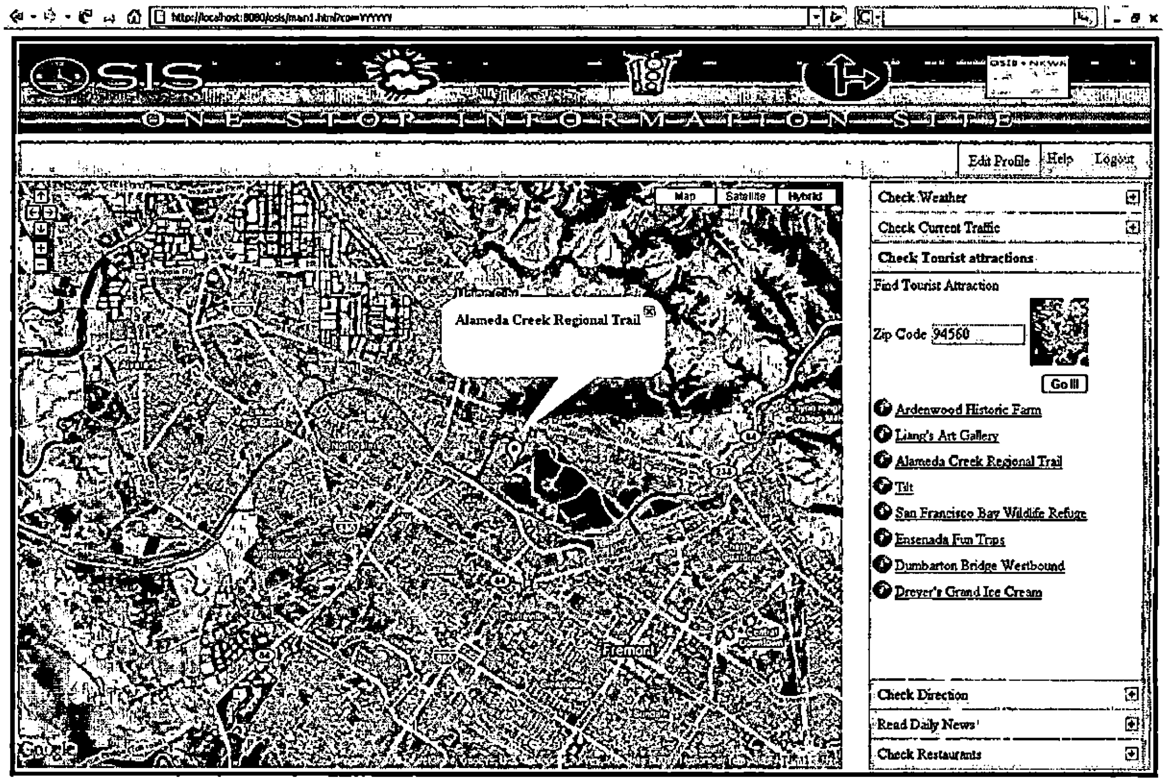
Debug: 92373

Yahoo! Local

Find Tourist Attractions ○ within 5 miles of location (wired)

Pipe Output

Figure 29. Tourism Pipe



Find Tourist Attraction

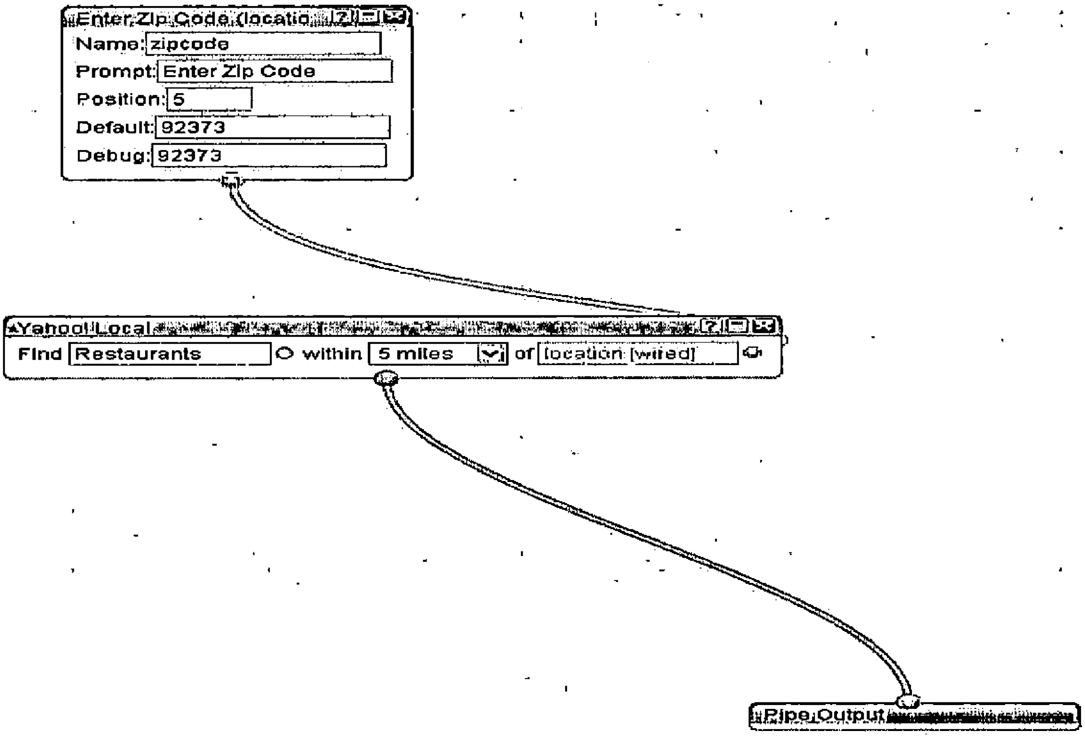
Zip Code 94560

- Ardenwood Historic Farm
- Liang's Art Gallery
- Alameda Creek Regional Trail
- Tilt
- San Francisco Bay Wildlife Refuge
- Ensenada Fun Trips
- Dumbarton Bridge Westbound
- Dreyer's Grand Ice Cream

Figure 30. Tourist Attraction Module Screen Shot

3.3.6 Check Restaurants

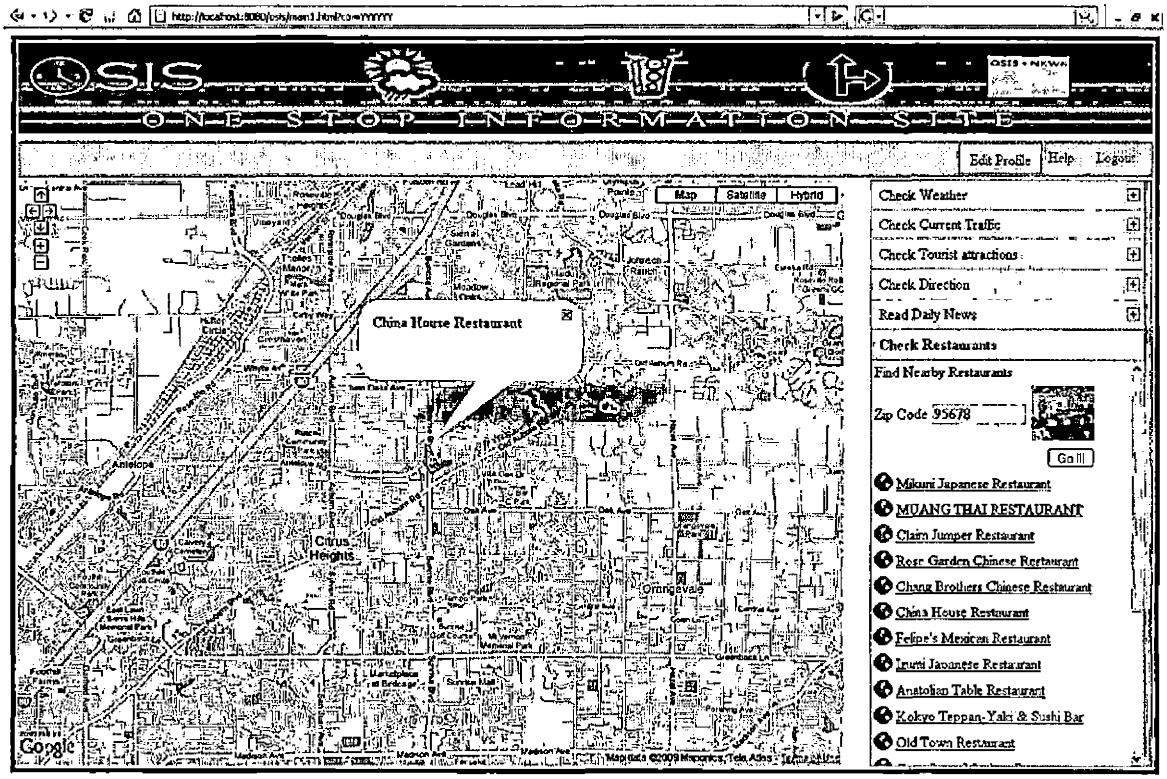
This module displays the restaurant list for the specified zip code. User's request (zip code) is sent to the Servlet by Ajax's restarantAjax function. The request will be send first to the main Servlet that is the dispatcher Servlet which forwards the requests to the corresponding restaurantsServlet. The restaurants Servlet has the URL for the restaurants pipe and the file path for the XSL file that is restaurants.xsl is transformed to it by transformUtil. Restaurants pipe is designed using yahoo pipes. XSL file is designed in a way to produce HTML output. It is designed in a way to display the restaurants details in new window. Google map object is used to show the output in the map.



Name: zipcode
Prompt: Enter Zip Code
Position: 5
Default: 92373
Debug: 92373

Find Restaurants ○ within 5 miles of location [wired]

Figure 31. Restaurant Pipe



Zip Code 95678

- Mikuni Japanese Restaurant
- MUANG THAI RESTAURANT
- Claim Jumper Restaurant
- Rose Garden Chinese Restaurant
- Chang Brothers Chinese Restaurant
- China House Restaurant
- Felipe's Mexican Restaurant
- Izumi Japanese Restaurant
- Anatolian Table Restaurant
- Kokyo Teppan-Yaki & Sushi Bar
- Old Town Restaurant

Figure 32. Restaurant Module Screen Shot

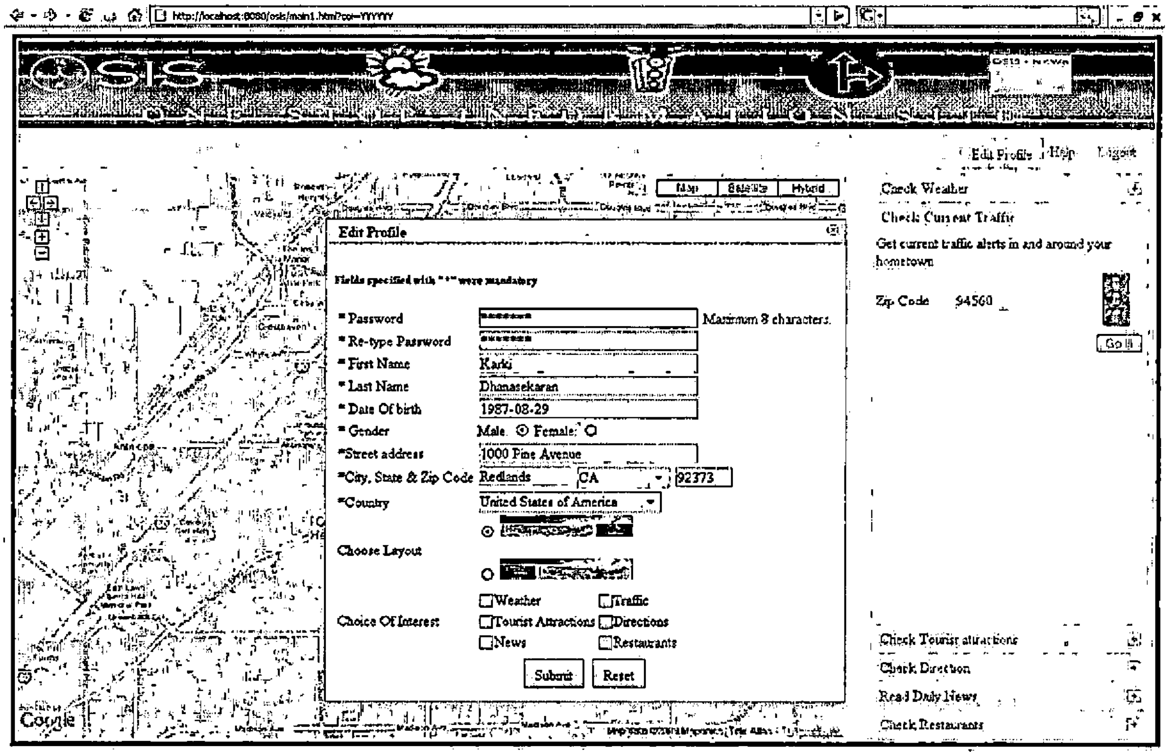
3.4 Security Aspects

OSIS is designed to be a highly secure site with the help of Servlet filter. When the user calls the main page without login into the OSIS the site is designed in a way to redirect him to the login page.

3.5 Friendly Features

3.5.1 Edit Profile

With edit profile option in main page user can change his options like layout and the theme at any instance of time. This feature is designed with Dojo functionality.



Edit Profile

Fields specified with "*" were mandatory

*Password	*******	Maximum 8 characters.
*Re-type Password	*******	
*First Name	Karki	
*Last Name	Dhanasekaran	
*Date Of birth	1987-08-29	
*Gender	Male ● Female ○	
*Street address	1000 Pine Avenue	
*City, State & Zip Code	Redlands CA 92373	
*Country	United States of America	
Choose Layout	● ○	
Choice Of Interest	□Weather □Traffic □Tourist Attractions □Directions □News □Restaurants	

Submit Reset

Check Weather

Check Current Traffic

Get current traffic alerts in and around your hometown

Zip Code 94560

Check Tourist attractions

Check Direction

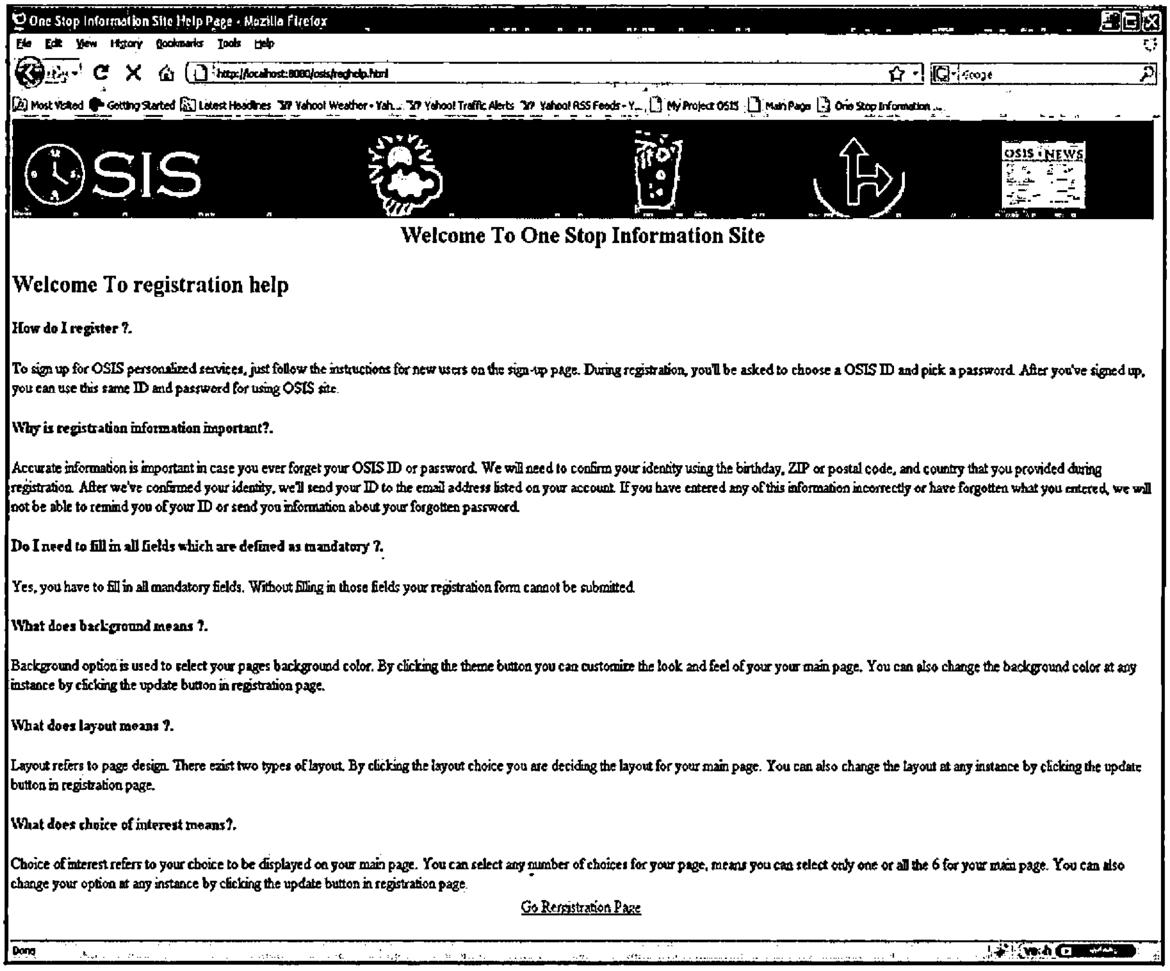
Read Daily News

Check Restaurants

Figure 33. Edit Profile Screen Shot

3.5.2 User Assistance

OSIS has help for both registration module and main module.



One Stop Information Site Help Page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:8080/osis/reghelp.html
Most Visited Getting Started Latest Headlines Yahoo! Weather - Yah... Yahoo! Traffic Alerts Yahoo! RSS Feeds - Y... My Project OSIS Main Page One Stop Information ...
OSIS
OSIS NEWS
Done

Welcome To One Stop Information Site

Welcome To registration help

How do I register ?.

To sign up for OSIS personalized services, just follow the instructions for new users on the sign-up page. During registration, you'll be asked to choose a OSIS ID and pick a password. After you've signed up, you can use this same ID and password for using OSIS site.

Why is registration information important?.

Accurate information is important in case you ever forget your OSIS ID or password. We will need to confirm your identity using the birthday, ZIP or postal code, and country that you provided during registration. After we've confirmed your identity, we'll send your ID to the email address listed on your account. If you have entered any of this information incorrectly or have forgotten what you entered, we will not be able to remind you of your ID or send you information about your forgotten password.

Do I need to fill in all fields which are defined as mandatory ?.

Yes, you have to fill in all mandatory fields. Without filling in those fields your registration form cannot be submitted.

What does background means ?.

Background option is used to select your pages background color. By clicking the theme button you can customize the look and feel of your your main page. You can also change the background color at any instance by clicking the update button in registration page.

What does layout means ?.

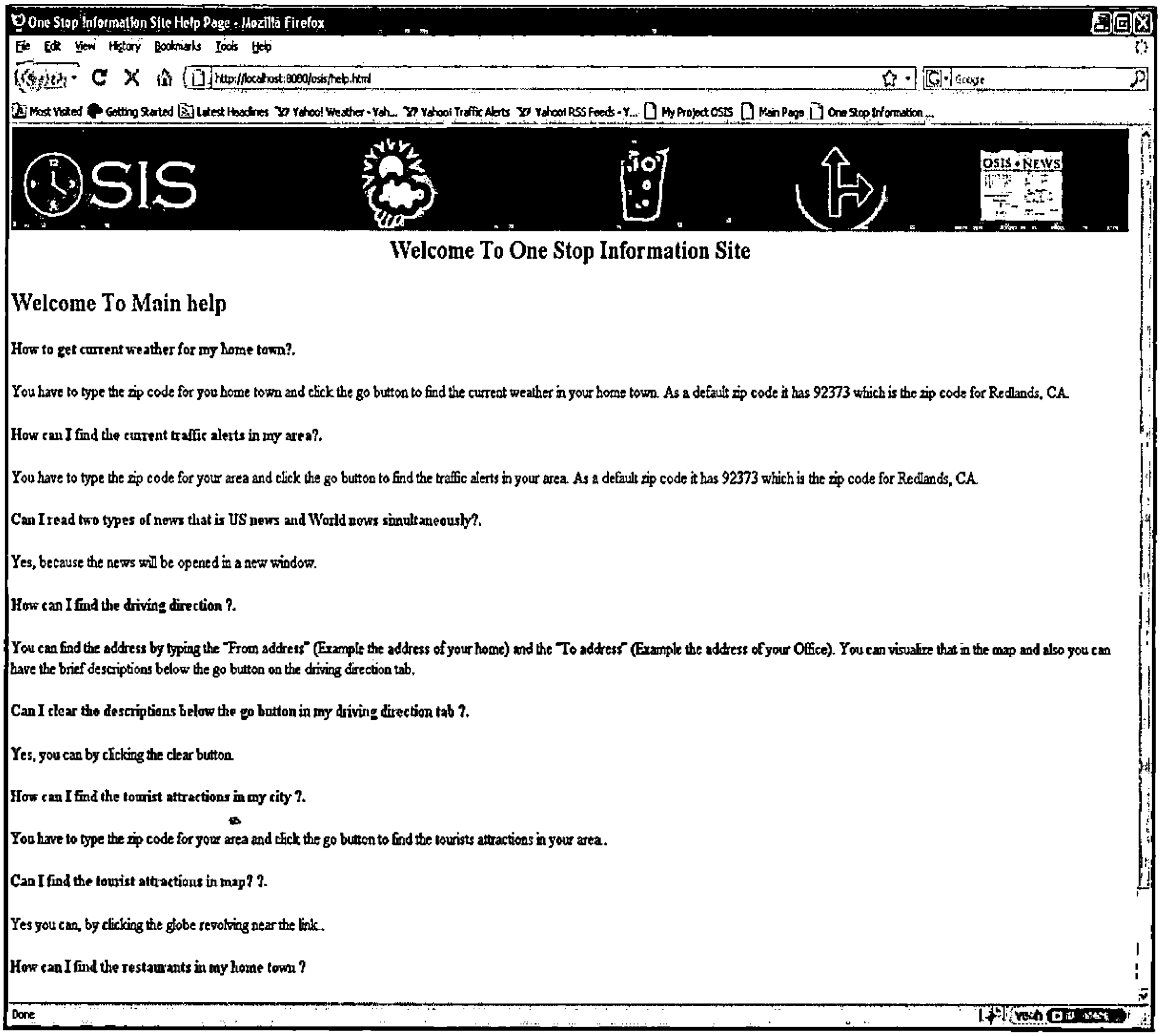
Layout refers to page design. There exist two types of layout. By clicking the layout choice you are deciding the layout for your main page. You can also change the layout at any instance by clicking the update button in registration page.

What does choice of interest means?.

Choice of interest refers to your choice to be displayed on your main page. You can select any number of choices for your page, means you can select only one or all the 6 for your main page. You can also change your option at any instance by clicking the update button in registration page.

Go Registration Page

Figure 34. Registration Help


One Stop Information Site Help Page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:8080/osis/help.html
Google
Most Visited Getting Started Latest Headlines Yahoo! Weather - Yah... Yahoo! Traffic Alerts Yahoo! RSS Feeds - Y... My Project OSIS Main Page One Stop Information ...
OSIS
OSIS • NEWS
Done

Welcome To One Stop Information Site

Welcome To Main help

How to get current weather for my home town?.

You have to type the zip code for you home town and click the go button to find the current weather in your home town. As a default zip code it has 92373 which is the zip code for Redlands, CA.

How can I find the current traffic alerts in my area?.

You have to type the zip code for your area and click the go button to find the traffic alerts in your area. As a default zip code it has 92373 which is the zip code for Redlands, CA.

Can I read two types of news that is US news and World news simultaneously?.

Yes, because the news will be opened in a new window.

How can I find the driving direction ?.

You can find the address by typing the "From address" (Example the address of your home) and the "To address" (Example the address of your Office). You can visualize that in the map and also you can have the brief descriptions below the go button on the driving direction tab.

Can I clear the descriptions below the go button in my driving direction tab ?.

Yes, you can by clicking the clear button.

How can I find the tourist attractions in my city ?.

You have to type the zip code for your area and click the go button to find the tourists attractions in your area.

Can I find the tourist attractions in map? ?.

Yes you can, by clicking the globe revolving near the link..

How can I find the restaurants in my home town ?

Figure 35. Main Help

3.5.3 Forget Password

OSIS helps to retrieve user's password in a secure way when the user forgets his password.

3.6 Benefits of Design Pattern

By using MVC pattern the following benefits were achieved.

3.6.1 Multiple Views Support

Separation of view from the model allows the user interface to display multiple views of the same data at the same time.

3.6.2 Change Accommodation

Independency between model and view supports adding new types of views to the system.

3.6.3 Separation Of Concerns

Independency between model, view and controller ensures reusability of business-logic across applications and development of multiple UI without disturbing business logic codebase.

3.7 Downside

3.7.1 Complexity

The MVC pattern introduces new levels of indirection and therefore increases the complexity of the solution slightly. It also increases the event-driven nature of the user-interface code, which can become more difficult to debug.

CHAPTER FOUR

SOFTWARE MAINTENANCE

4.1 Test Approach

This section describes the approach and methodologies used for testing the application. To test the functionality of OSIS black box test is performed. The description and result of testing is discussed below.

4.2 Black Box Testing

Black-box testing treats the system as an "opaque-box", so it doesn't explicitly use knowledge of the internal structure. It is usually used to test the functional requirements of the system. For OSIS Selenium is used for functional testing.

4.2.1 Testing Using Selenium

Using Selenium IDE all modules in OSIS including features like edit profile, forget password were tested. Test scripts, results and screen shots are as following. For test case refer to APPENDIX B. Refer APPENDIX C for Test results.

CHAPTER FIVE

MAINTENANCE MANUAL

In this manual you will find the details about the system requirements, OSIS WAR file structure, and details about database configuration and deployment of WAR file in Tomcat.

5.1 Running the Application

For running OSIS Software the following Software has to be installed and configured in the machine.

- JavaSE 1.5 Standard Edition or higher version
- Apache Tomcat 5.5.26 or higher version
- MySQL Database 5.1
- Eclipse 3.3 (optional)

5.1.1 JavaSE Installation

The procedure to download the JRE broadly consists of the following 3 steps:

- Download and Install JRE
- Enable and Configure
- Test Installation

The following software will be installed on the computer:

- The Java 2 Runtime Environment, Standard Edition (JRE), which includes the Java Plug-in software and the Java Virtual Machine components needed to run Java technology based applets in a web browser.
- Java Web Start software which simplifies launching and running web-based programs that use Java technology based software.

The JavaSE software and the brief installation instructions are available at the Sun site (http://java.com/en/download/index.jsp)

5.1.2 Apache Tomcat Installation

To install apache server the following steps should be followed:

- Unzip tomcat-5.5.26-preconfigured.zip into the top level of the C drive.
- Set JAVA_HOME by setting the environment variable to point at the top-level of the Java installation directory (e.g., C:\Program Files\Java\jdk1.6.0_10

- Set CLASSPATH by setting the environment variable to include ".", C:\Servlets+JSP, and the Servlet/JSP JAR files.
- Test the server.

The Apache Tomcat server and the brief installation instructions are available at the Apache web site (http://tomcat.apache.org)

5.1.3 MySQL Installation

Download and Install MySQL 5.1 version or higher version. MySQL installation for windows operating system is discussed below. Run the MSI file and use the following settings:

- Typical Setup
- Skip Sign-Up
- Make sure "Configure the MySQL Server now" is checked
- Detailed Configuration
- Developer Machine
- Multifunctional Database
- InnoDB Tablespace Settings - leave everything default
- Decision Support (DSS)/OLAP

- Make sure "Enable TCP/IP Networking" is checked and leave the port number at 3306
- Standard Character Set
- Check Install As Windows Service
- Enter your root password
- Hit "execute" to install.

The MySQL database and the brief installation instructions for all the operating system are available at the MySQL web site (http://www.mysql.com/).

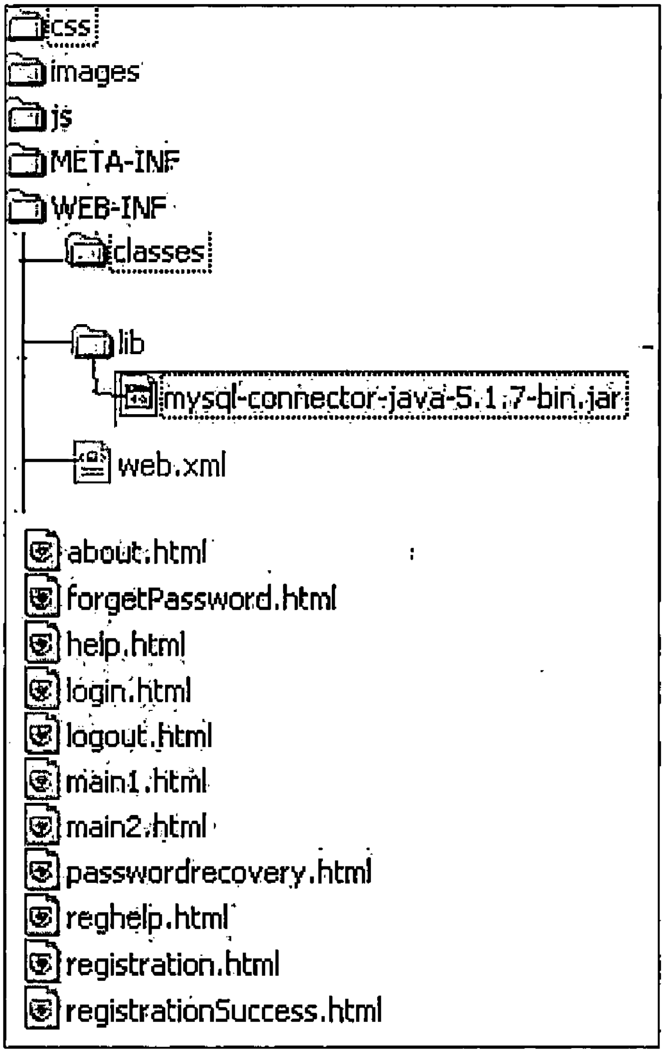
5.1.4 Eclipse Installation

Download Eclipse zip file. Unzip the Eclipse SDK into a directory (default c:\eclipse) and execute the *eclipse.exe* program. If Eclipse does not start the most probable reason is that the *javaw.exe* program cannot be found. Make sure that the PATH environment variable has the path to the JRE/JDK binary directory that contains javaw.exe.

Brief installation instructions and troubleshooting steps were available at the IBM web site (http://www.ibm.com/us/en).

5.2 Web Application Archive Structure

Once you are done with your prerequisite software installed as specified above, copy the OSIS WAR file. The directory structure shown below illustrates the way in which the WAR file is organized.



```
css
images
js
META-INF
WEB-INF
  classes
  lib
    mysql-connector-java-5.1.7-bin.jar
  web.xml
about.html
forgetPassword.html
help.html
login.html
logout.html
main1.html
main2.html
passwordrecovery.html
reghelp.html
registration.html
registrationSuccess.html
```

Figure 36. Web Application Archive Structure

5.3 Deploying Web Application Archive File

To deploy OSIS WAR file in Tomcat the following procedure is followed:

- Stop Tomcat.
- Delete existing deployment.
- Copy OSIS WAR file to TOMCAT_HOME/webapps/.
- Start Tomcat.

5.4 Database Configuration

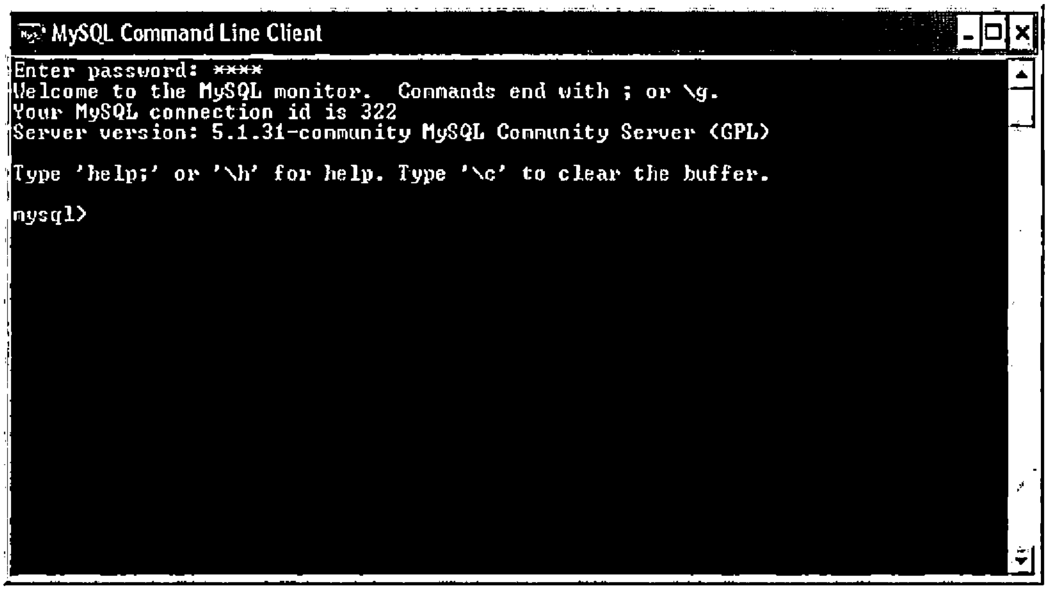
After deploying OSIS Software in Tomcat, the next step is to create tables in MySQL database. In order to create tables follow the steps outlined below:

- Using command line MySQL client create new database called "**osis**".
- Exit from the command line client.
- Go to Windows Command Line prompt "C:\Program Files\osis\database" and run the script file "tables.sql". For example,

```
C:\MyProject mysql -u osis -p osis < tables.sql
```

- This script will create the Layout and User tables in the database.
- In order to make sure the tables are

created correctly, connect to MySQL command line client and enter the root password.



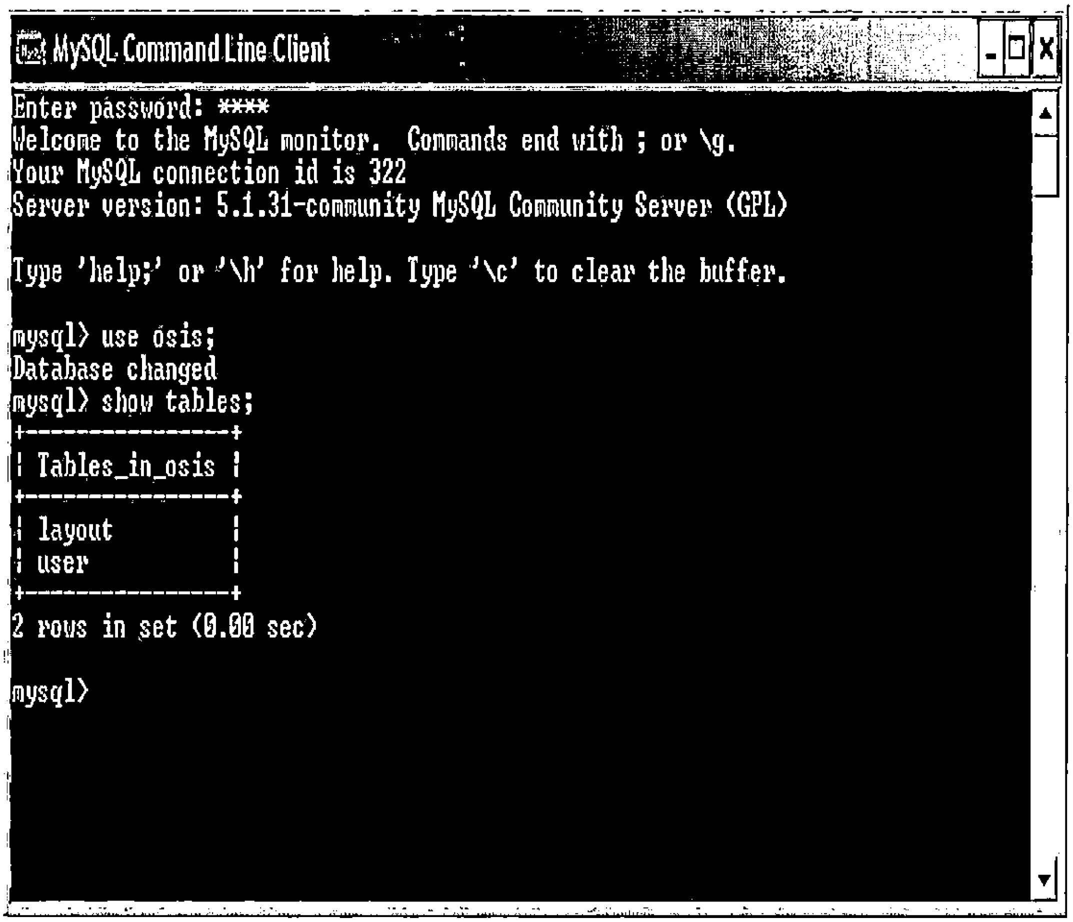
```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 322
Server version: 5.1.31-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

Figure 37. Database Screen Shot

- Connect the osis database and list all the tables and make sure the listed tables are same as shown in above step.



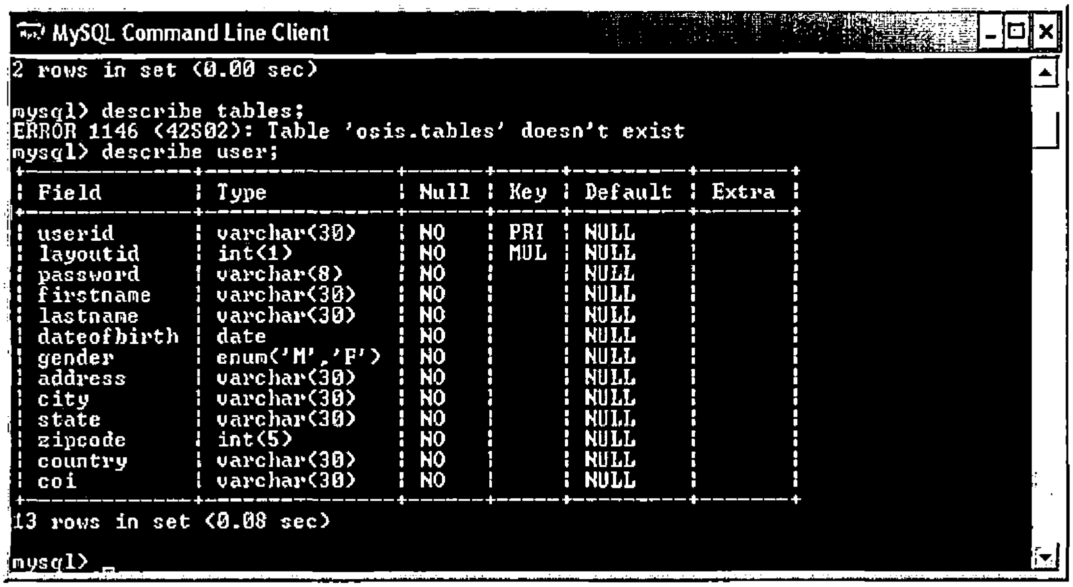
```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 322
Server version: 5.1.31-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use osis;
Database changed
mysql> show tables;
+----------------+
| Tables_in_osis |
+----------------+
| layout         |
| user           |
+----------------+
2 rows in set (0.00 sec)

mysql>
```

Figure 38. Tables Screen Shot



```
2 rows in set (0.00 sec)

mysql> describe tables;
ERROR 1146 (42S02): Table 'osis.tables' doesn't exist
mysql> describe user;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| userid      | varchar(30) | NO   | PRI | NULL    |       |
| layoutid    | int(1)      | NO   | MUL | NULL    |       |
| password    | varchar(8)  | NO   |     | NULL    |       |
| firstname   | varchar(30) | NO   |     | NULL    |       |
| lastname    | varchar(30) | NO   |     | NULL    |       |
| dateofbirth | date        | NO   |     | NULL    |       |
| gender      | enum('M','F') | NO |     | NULL    |       |
| address     | varchar(30) | NO   |     | NULL    |       |
| city        | varchar(30) | NO   |     | NULL    |       |
| state       | varchar(30) | NO   |     | NULL    |       |
| zipcode     | int(5)      | NO   |     | NULL    |       |
| country     | varchar(30) | NO   |     | NULL    |       |
| coi         | varchar(30) | NO   |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
13 rows in set (0.08 sec)

mysql> _
```

Figure 39. Table Description Screen Shot

- Now required tables for OSIS Software are successfully imported in MySQL database.

CHAPTER SIX

CONCLUSION AND FURTHER DIRECTIONS

6.1 Enhancements/ Improvements

OSIS is a complete software, like any piece of software, but there is always room to extend and enhance. OSIS was designed using MVC pattern to simplify the job of further enhancement. Up to my knowledge the following were several features which can be added to augment OSIS. Other then this there may be more and more features that can be added to enhance OSIS.

6.1.1 User's Reminder

Can store user's rememberable dates and alert or remind him if he enters into his account.

By doing this OSIS can act as user's dynamic date book.

6.1.2 More Features

Can plug in more features like hot job, live traffic, finance etc., which has to be specified with its geographic location. Can tune OSIS to show user's default zip code in all the modules, by retrieving his home town zip code from registration page.

6.2 Conclusion

OSIS serves as a personalized page for the user, providing him with interactive day to day information. Weather affects almost every aspect of our daily life, from our daily commute to our recreational activities. With OSIS we can have access to local forecasts 24 hours a day, 7 days a week.

With OSIS traffic alert and direction we can have the pulse of the road in our fingertips, when and where we need. OSIS helps us to take control of our commute with real time traffic information for our daily route. Daily news option provides us with local news, sports, classified ads, obituaries, and editorials.

With tourist attraction we have access to all the best tourist attraction sites in the US, allowing us to tailor our own unforgettable experience.

With restaurant lists we have the list of all best restaurants in the US along with its geographic location, allowing us to have a most unforgettable dining experience.

APPENDIX A

SAMPLE SOURCE CODE

```
import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import edu.cs.osis.util.TransformUtil;

/**

 * <b>OSIS</b>

 * This Servlet is designed for OSIS weather module.

 * @author Poongkuzhali Dhanasekaran

 * </p>Date : 2nd January 2009.

 */

public class WeatherServlet extends HttpServlet {

  private static final long serialVersionUID = 1L;


  public static final String REQUEST_PARAM_NAME =

"zipcode";


  public void doGet(HttpServletRequest request,

HttpServletResponse response)

      throws ServletException, IOException {

    response.setContentType("text/html");

    PrintWriter out = response.getWriter();
try {
      URL weatherRSS = new
URL("http://weather.yahooapis.com/forecastrss?p="
          + request.getParameter(REQUEST_PARAM_NAME));
      URL xslFileURL =
TransformUtil.class.getClassLoader().getResource("xsl/weath
er.xsl");
      out.write(TransformUtil.transform(weatherRSS,
xslFileURL));
    }
    catch (Exception e) {
      out.write("error");
    }


    out.close();
    out.flush();
  }
  public void doPost(HttpServletRequest request,
HttpServletResponse response)
      throws ServletException, IOException {
    doGet(request, response);  }
}
```

Transform Util:

```
package edu.cs.osis.util;

import java.io.File;

import java.io.IOException;

import java.io.StringWriter;

import java.net.URL;

import java.util.Properties;

import javax.xml.transform.OutputKeys;

import javax.xml.transform.Source;

import javax.xml.transform.Transformer;

import javax.xml.transform.TransformerException;

import javax.xml.transform.TransformerFactory;

import javax.xml.transform.URIResolver;

import javax.xml.transform.stream.StreamResult;

import javax.xml.transform.stream.StreamSource;

/**

 * <b>OSIS</b>

 * This java class is responsible for transforming the

 * URL for rss feed and xsl to the weather servlet.

 * @author Poongkuzhali Dhanasekaran

 * </p>Date : 29nd January 2009.

 */
```

```
public class TransformUtil {
  private static boolean debug = false;
  private static TransformerFactory tFactory =
TransformerFactory.newInstance();
  private static final String ENCODE = "UTF-8";
  public static String transform(URL rss, URL xsl) throws
TransformerException, IOException {
    tFactory.setURIResolver(new URIResolver() {
      public Source resolve(String href, String base) {
        if (href == null) {
          return null;
        }
        File file = new File(href);
        if (file.isAbsolute()) {
          return null;
        }
        try {
          URL myURL = new URL(href);
          if (myURL.toExternalForm() != null) {
            return null;
          }
        }
catch (Exception _) {
        }
        return new
StreamSource(TransformUtil.class.getClassLoader().getResour
ceAsStream(
            "xsl/" + href));
      }
    });
    Transformer transformer = null;
    StreamSource xmlSource = new
StreamSource(rss.openStream());

    StringWriter writer = new StringWriter();
    transformer = tFactory.newTransformer(new
StreamSource(xsl.openStream()));
    Properties prop = new Properties();
    prop.setProperty(OutputKeys.CDATA_SECTION_ELEMENTS,
"description");
    prop.setProperty(OutputKeys.ENCODING, ENCODE);
    transformer.setOutputProperties(prop);
    transformer.transform(xmlSource, new
StreamResult(writer));
```

```
String transformResult = writer.toString();
    transformResult = transformResult.replaceAll("&lt;",
"<");
    transformResult = transformResult.replaceAll("&gt;",
">");
    if (debug) {
      System.out.println("Transform Result : " +
transformResult);
    }
    return transformResult;
  }

  // For Testing Purpose Only
  public static void main(String arg[]) {
    try {
      String rssURL =
"http://weather.yahooapis.com/forecastrss?p=92373";
      String xslPath = "xsl/weather.xsl";
    rssURL =
"http://local.yahooapis.com/MapsService/rss/trafficData.xml
?appid=YdnDemo&zip=92654";
      xslPath = "xsl/traffic.xsl";
```

```
 rssURL = "http://rss.news.yahoo.com/rss/world";
      xslPath = "xsl/world.xsl";
      rssURL =
"http://pipes.yahoo.com/pipes/pipe.run?_id=dabafc8cbd6c2d1d
8cbee80aae96c4fd&_render=rss&zipcode=92373";
      rssURL =
"http://pipes.yahoo.com/pipes/pipe.run?_id=8twsDYsE3hG9t77P
3rVd_w&_render=rss&zipcode=92373";
      xslPath = "xsl/tourism.xsl";
      URL weatherRSS = /*
TransformUtil.class.getClassLoader().getResource(rssURL);*/
new URL(
          rssURL);
      URL xslFileURL =
TransformUtil.class.getClassLoader().getResource(xslPath);
      TransformUtil.transform(weatherRSS, xslFileURL);
    }
    catch (Exception e) {
      e.printStackTrace();
  }
  }
}
```

APPENDIX B

SAMPLE TEST CASE

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
<head profile="http://selenium-
ide.openqa.org/profiles/test-case">
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8" />
<link rel="selenium.base" href="" />
<title>TestOne</title>
</head>
<body>
<table cellpadding="1" cellspacing="1" border="1">
<thead>
<tr><td rowspan="1" colspan="3">TestOne</td></tr>
</thead><tbody>
<tr>
     <td>open</td>
     <td>/osis/login.html</td>
     <td></td>
</tr>
```

```
<tr>
    <td>type</td>
    <td>userId</td>
    <td>kkhaveen@gmail.com</td>
</tr>
<tr>
    <td>type</td>
    <td>password</td>
    <td>khaveen</td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>//input[@value='Submit']</td>
    <td></td>
</tr>
<tr>
    <td>click</td>
    <td>dijit_form_Button_0</td>
    <td></td>
</tr>
<tr>
    <td>click</td>
```

```
<td>//div[@id='editProfileDialog']/div[1]/span[2]</td>

     <td></td>

</tr>

<tr>

     <td>type</td>

     <td>password</td>

     <td>khaveen</td>

</tr>

<tr>

     <td>click</td>

     <td>//input[@value='Go !!!']</td>

     <td></td>

</tr>

<tr>

     <td>click</td>

     <td>//div[@id='tourist']/div[1]/div[4]</td>

     <td></td>

</tr>

<tr>

     <td>click</td>
```

```
<td>//input[@value='Go !!!' and @type='button' and
@onclick=&quot;touristAjax(document.getElementById('tourist
zipcode').value);&quot;]</td>
     <td></td>
</tr>
<tr>
     <td>click</td>
     <td>//img[@onclick='showMarker(34.056862, -117.221786,
&quot;&lt;b&gt;Bracken Bird
Farm&lt;/b&gt;&lt;br/&gt;&quot;);']</td>
     <td></td>
</tr>
<tr>
     <td>click</td>
     <td>//div[@id='restaurants']/div[1]/div[4]</td>
     <td></td>
</tr>
<tr>
     <td>click</td>
     <td>//input[@value='Go !!!' and @type='button' and
@onclick=&quot;restaurantsAjax(document.getElementById('res
taurantszipcode').value);&quot;]</td>
```

```
<td></td>
</tr>
<tr>
     <td>click</td>
     <td>//img[@onclick='showMarker(34.063149, -117.21606,
&quot;&lt;b&gt;Thai House
Restaurant&lt;/b&gt;&lt;br/&gt;&quot;,
&quot;R&quot;);']</td>
     <td></td>
</tr>
<tr>
     <td>click</td>
     <td>//img[@onclick='showMarker(34.056273, -117.181159,
&quot;&lt;b&gt;The Gourmet Pizza
Shoppe&lt;/b&gt;&lt;br/&gt;&quot;, &quot;R&quot;);']</td>
     <td></td>
</tr>
<tr>
     <td>click</td>
     <td>//img[@onclick='showMarker(34.05383, -117.174542,
&quot;&lt;b&gt;Tokyo Restaurant&lt;/b&gt;&lt;br/&gt;&quot;,
&quot;R&quot;);']</td>
```

```
<td></td>
</tr>
<tr>
    <td>click</td>
    <td>//div[@id='tourist']/div[1]/div[4]</td>
    <td></td>
</tr>
<tr>
    <td>type</td>
    <td>touristzipcode</td>
    <td>92374</td>
</tr>
<tr>
    <td>click</td>
    <td>//input[@value='Go !!!' and @type='button' and
@onclick=&quot;touristAjax(document.getElementById('tourist
zipcode').value);&quot;]</td>
    <td></td>
</tr>
<tr>
    <td>click</td>
```

```
<td>//input[@value='Go !!!' and @type='button' and
@onclick=&quot;touristAjax(document.getElementById('tourist
zipcode').value);&quot;]</td>
     <td></td>
</tr>
<tr>
     <td>click</td>
     <td>//div[@id='map_canvas']/div[5]/div[1]/div</td>
     <td></td>
</tr>
<tr>
     <td>click</td>
     <td>//div[@id='map_canvas']/div[5]/div[2]/div</td>
     <td></td>
</tr>
<tr>
     <td>click</td>
     <td>//div[@id='map_canvas']/div[5]/div[3]/div</td>
     <td></td>
</tr>
<tr>
     <td>clickAndWait</td>
```

```
<td>link=Logout</td>
     <td></td>
</tr>
<tr> <td>clickAndWait</td>
     <td>link=Home</td>
     <td></td>
</tr>
</tbody></table>
</body>
</html>
```

APPENDIX C

SAMPLE TEST RESULT

```
[info] Executing: |open | /osis/login.html | |

[info] Executing: |type | userId | kkhaveen@gmail.com |

[info] Executing: |type | password | khaveen |

[info] Executing: |clickAndWait | //input[@value='Submit']
| |

[info] Executing: |click | dijit_form_Button_0 | |

[info] Executing: |click |
//div[@id='editProfileDialog']/div[1]/span[2] | |

[info] Executing: |type | password | khaveen |

[info] Executing: |click | //input[@value='Go !!!'] | |

[info] Executing: |click |
//div[@id='tourist']/div[1]/div[4] | |

[info] Executing: |click | //input[@value='Go !!!' and
@type='button' and
@onclick="touristAjax(document.getElementById('touristzipco
de').value);"] | |
[info] Executing: |click |
//img[@onclick='showMarker(34.056862, -117.221786,
"<b>Bracken Bird Farm</b><br/>");'] | |

[info] Executing: |click |
//div[@id='restaurants']/div[1]/div[4] | |

[info] Executing: |click | //input[@value='Go !!!' and
@type='button' and
@onclick="restaurantsAjax(document.getElementById('restaura
ntszipcode').value);"] | |

[info] Executing: |click |
//img[@onclick='showMarker(34.063149, -117.21606, "<b>Thai
House Restaurant</b><br/>", "R");'] | |

[info] Executing: |click |
//img[@onclick='showMarker(34.056273, -117.181159, "<b>The
Gourmet Pizza Shoppe</b><br/>", "R");'] | |

[info] Executing: |click |
//img[@onclick='showMarker(34.05383, -117.174542, "<b>Tokyo
Restaurant</b><br/>", "R");'] | |
[info] Executing: |click |
//div[@id='tourist']/div[1]/div[4] | |

[info] Executing: |type | touristzipcode | 92374 |

[info] Executing: |click | //input[@value='Go !!!' and
@type='button' and
@onclick="touristAjax(document.getElementById('touristzipco
de').value);"] | |

[info] Executing: |click | //input[@value='Go !!!' and
@type='button' and
@onclick="touristAjax(document.getElementById('touristzipco
de').value);"] | |

[info] Executing: |click |
//div[@id='map_canvas']/div[5]/div[1]/div | |

[info] Executing: |click |
//div[@id='map_canvas']/div[5]/div[2]/div | |

[info] Executing: |click |
//div[@id='map_canvas']/div[5]/div[3]/div | |

[info] Executing: |clickAndWait | link=Logout | |
[info] Executing: |clickAndWait | link=Home | |
```

REFERENCES

[1] David Flanagan, "JavaScript: The Definitive Guide" O'Reilly Media, Incorporated, August 2006.

[2] Fowler, M., "UML DISTILLED", (Third Edition) - Covers through Version 2.0 OMG UML Standard. Addison-Wesley, 2004.

[3] IEEE Recommended Practice for Software Engineering Process, Addison-Wesley,2000.

[4] Ian Llyod,"Build Your Own Website Using HTML &CSS", SitePoint Pty,Limited, November 2008.

[5] Joshua Bloch,"Effective Java-Programming Language Guide", Addison-Wesley, February 2007.

[6] Ken Arnold, James Gosling, David Holmes, " The Java Programming Language",Addison-Wesley,June 2006.

[7] Marty Hall, Larry Brown,"Core Servlet and JavaServerPages", Prentice Hall Professional Technical, August 2003.

[8] Matthew A Russell," Dojo: The Definitive Guide", O'Reilly Media, Incorporated, June 2008.

[9] Ray,Erick, "Learning XML".O"Reilly & Association.Inc.,2001.

[10] Thomas A. Powell,"The Complete Reference Ajax", McGraw-Hill/Osborne Media, February 2008.