2006

Restaurant recommendation system (RRS)

Tai-jung Lin

Follow this and additional works at: https://scholarworks.lib.csusb.edu/etd-project

Part of the Databases and Information Systems Commons

Recommended Citation
https://scholarworks.lib.csusb.edu/etd-project/3009

This Project is brought to you for free and open access by the John M. Pfau Library at CSUSB ScholarWorks. It has been accepted for inclusion in Theses Digitization Project by an authorized administrator of CSUSB ScholarWorks. For more information, please contact scholarworks@csusb.edu.
RESTAURANT RECOMMENDATION SYSTEM
(RRS)

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
Tai-jung Lin
June 2006
RESTAURANT RECOMMENDATION SYSTEM

(RRS)

A Project
Presented to the
Faculty of
California State University,
San Bernardino

by
Tai-jung Lin
June 2006

Approved by:

Dr. Keith Schubert, Chair, Computer Science  6/6/06

Dr. George Georgiou

Dr. Ernesto Gomez

Date
ABSTRACT

Restaurant Recommendation System (RRS) is an on-line system to search restaurants. The system allows visitors to browse information about the restaurants, including searching restaurants, viewing/giving recommendations, and viewing/rating restaurants. Visitors can only view recommendations and view rating results. If the user would like to give recommendations or rate restaurants, he or she has to log in or sign up. Additionally, the user can check their history of recommendations they gave.

In addition to customer functions, the system also provides functions to the administrator that allows he or she to manage the contents of the site, including edit customer, edit restaurants, edit recommendations, and edit ratings, etc.

This project is based on a Java Server Pages (JSP) language, Java Server Programming, which is a server side scripting language. In this system, I use MySQL database to maintain persistent data and Tomcat as a web system server.
ACKNOWLEDGMENTS

I sincerely thank my advisor, Dr. Keith Schubert, for his encouragement and guidance during my graduate studies, and for being right there with the thrusters when asked. He gave me the freedom to pursue this research, also give me significant help to resolve my problems and give me positive directions to do my research. I thank the members of my thesis committee, Dr. Georgiou and Dr. Gomez, for their counsel and assistance. They offered me a lot of precious information and suggestions during all the period of doing my project. I would like to thank my sister and brother in law, deeply from my heart, who provide me the chance to fulfill my every dream in life. Their support and encouragement were a large part of what kept me going.

Thanks to my parents, their love warmly supports me in every day and their encouragement gives me the power to accomplish every success in my life.

The support of the National Science Foundation under award 9810708 is gratefully acknowledged.
TABLE OF CONTENTS

ABSTRACT .................................................................................... iii
ACKNOWLEDGMENTS ................................................................. iv
LIST OF TABLES .......................................................................... x
LIST OF FIGURES ......................................................................... xi

CHAPTER ONE: INTRODUCTION
  1.1 Purpose of the Project ......................................................... 1
  1.2 Project Products ................................................................. 2

CHAPTER TWO: SYSTEM ARCHITECTURE
  2.1 System Interfaces ............................................................... 5
  2.2 Hardware Interfaces .......................................................... 5
  2.3 Software Interfaces ........................................................... 5

CHAPTER THREE: DATABASE DESIGN
  3.1 Data Analysis ..................................................................... 7
  3.2 Database Model-Relational Diagram ................................. 7
  3.3 Database Model-Relational Schema .................................... 9
  3.4 Data Type and Detail ......................................................... 10

CHAPTER FOUR: PROJECT IMPLEMENTATION
  4.1 User Interface Design ......................................................... 14
  4.2 Graphical User Interface .................................................... 15
    4.2.1 Index Page ................................................................. 15
    4.2.2 Guest Main Page ....................................................... 16
    4.2.3 Guest Signup Page .................................................... 17
    4.2.4 Guest Complete Signup Page .................................... 18
    4.2.5 Guest Search Restaurant Page ................................. 19
4.2.31 Member Distance Search Page ............ 48
4.2.32 Member Ratings Search Page ............ 49
4.2.33 Member Poster Search Page ............. 50
4.2.34 Member Recommendation Page .......... 51
4.2.35 Member Post Page .......................... 53
4.2.36 Member Reply Page ...................... 54
4.2.37 Member Rating Page ..................... 55
4.2.38 Member View Rating Page ............. 56
4.2.39 Member Give Rate Page ................. 57
4.2.40 Administrator Login Page ............... 58
4.2.41 Administrator Main Page ............... 59
4.2.42 Administrator Edit Member Page ...... 60
4.2.43 Administrator Check Member Page ...... 61
4.2.44 Administrator Delete Member Page .... 62
4.2.45 Administrator Edit Restaurant Page ... 63
4.2.46 Administrator Add Restaurant Page ..... 65
4.2.47 Administrator Restaurants Page ...... 66
4.2.48 Administrator Restaurant Page ....... 67
4.2.49 Administrator View Restaurant Page ... 68
4.2.50 Administrator Delete Page .............. 69
4.2.51 Administrator Edit Restaurant Page ... 70
4.2.52 Administrator Modified Page .......... 71
4.2.53 Administrator Edit Posting Page ..... 72
4.2.54 Administrator View Posting Page ..... 73
4.2.55 Administrator Search Title Page ...... 74

vii
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.56</td>
<td>Administrator Delete Post Page</td>
<td>75</td>
</tr>
<tr>
<td>4.2.57</td>
<td>Administrator Search Poster Page</td>
<td>76</td>
</tr>
</tbody>
</table>

CHAPTER FIVE: SYSTEM VALIDATION

5.1 Introduction .................................. 77
5.2 Unit Test Plan .................................. 77
5.3 Subsystem Test Plan ............................ 80
5.4 System Test Plan ............................... 82

CHAPTER SIX: MAINTENANCE MANUAL

6.1 Software Installation ............................ 84
6.1.1 Install Java Platform ........................ 84
6.1.2 Install Tomcat ............................... 85
6.1.3 Install Security ............................. 86
6.1.4 Install Database ............................. 87
6.2 Variable Modifications .......................... 89
6.2.1 System Variables ............................ 89
6.3 System Installation/Migration .................. 90
6.4 Backup and Restore ............................. 90
6.4.1 System Backup ............................... 91
6.4.2 Database Backup .............................. 91
6.4.3 System Restore ............................... 92
6.4.4 Database Restore ............................. 92
6.4.5 Re-installation Manual ....................... 92

CHAPTER SEVEN: CONCLUSION AND FUTURE DIRECTIONS

7.1 Conclusion ..................................... 94
7.2 Future Direction ............................... 95
LIST OF TABLES

Table 1. Structure of Table Restaurant ................. 10
Table 2. Structure of Table Signup ...................... 11
Table 3. Structure of Table Board ...................... 11
Table 4. Structure of Table Zipcode ..................... 12
Table 5. The Unit Test Results .......................... 78
Table 6. Subsystem Test Results .......................... 81
Table 7. System Test Results .............................. 83
LIST OF FIGURES

Figure 1. System Deployment Diagram ...................... 4
Figure 2. Database Model-Relational Diagram ............. 8
Figure 3. Database Model-Relational Schema ............... 9
Figure 4. Use Case Diagram ................................. 13
Figure 5. Index Page ....................................... 15
Figure 6. Guest Main Page .................................. 16
Figure 7. Guest Signup Page ................................. 17
Figure 8. Guest Complete Signup Page ..................... 18
Figure 9. Guest search Restaurant Page .................... 19
Figure 10. Guest Search Restaurant Result Page .......... 20
Figure 11. Guest rate Page .................................. 21
Figure 12. Guest View Restaurant Detail Page ............. 22
Figure 13. Guest View Map Page ............................. 23
Figure 14. Guest Advanced Search Page .................... 25
Figure 15. Guest Search Distance Page ..................... 26
Figure 16. Guest Search Rating Page ....................... 27
Figure 17. Guest Search Recommender Page ............... 28
Figure 18. Guest View Recommendations Page ............ 30
Figure 19. Guest Post Error Page .......................... 31
Figure 20. Guest Rate Page .................................. 32
Figure 21. Guest View Rated Restaurant Page ............. 33
Figure 22. Member Login Page .............................. 34
Figure 23. Member Forget Password Page ................. 35
Figure 24. Member Password Return Page .................. 36
Figure 50. Administrator Add Restaurants Page .......... 65
Figure 51. Administrator Restaurants Page .............. 66
Figure 52. Administrator Restaurant Page ................ 67
Figure 53. Administrator View Restaurant Page .......... 68
Figure 54. Administrator Delete Page ..................... 69
Figure 55. Administrator Edit Restaurant Page .......... 70
Figure 56. Administrator Modified Page ................... 71
Figure 57. Administrator Edit Posting Page ............... 72
Figure 58. Administrator view Posting Page .............. 73
Figure 59. Administrator Search Title Page ............... 74
Figure 60. Administrator Delete Post Page ............... 75
Figure 61. Administrator Search Poster Page ............ 76
CHAPTER ONE

INTRODUCTION

This project is a restaurant recommendation system (RRS). RRS is an on-line system to search restaurants. Visitors can browse all restaurants in L.A, and get information about restaurant name, type, address, phone, rating, price, and map. The functions include searching restaurants, viewing/giving recommendations, and viewing/rating restaurants. Additionally, the user can edit their account information and check their recommendation records. The biggest difference between visitors and members is that visitors can only view recommendations and rating results. If visitors would like to give recommendations or rate restaurants, they have to log in or sign up.

1.1 Purpose of the Project

The purpose of this system is to let people get ideas about which restaurant will be great for them. This system can give people some suggestions; also you can get others’ opinions from this site. Further more, you can find the best restaurants by viewing the ratings page, which gathers many members’ experience and response. This system is designed to people to search the information you send, and response all those restaurants matched the customers’ request. For
instance, you can search by a price range or by location. Further more; the restaurant result includes a map link, so you don’t need to check the map by yourself. Except viewing other’s opinions, you can give suggestions by rating restaurants to other people. This system is like a communication bulletin for people who love to eat. In this site, there are many ways to search restaurants and rate, include by zip code, by type, by keyword, by distance, by price, and by recommendation search. These will be easier for people to use.

1.2 Project Products

The RRS project led to the following products:

- Implementation of RRS: a working web-based online application system with JSP programs, JavaScript and MySQL database, which achieves the specific needs of RRS. All the forms follow the original paper application form in order to accomplish convenient and familiar processes for the user.

- System documentation: a project documentation, which is available with system design, specifications, project implementation and testing reports.
CHAPTER TWO

SYSTEM ARCHITECTURE

The RRS project aims to be user friendly and convenient for the customers. The architecture of RRS is composed with a web server, a database server, and a client browser. Customer browsers use the Internet (TCP/IP) to get restaurant information. Web server is under HTTP/HTTPS. The web application connects to the database with TCP, and accesses database functionality through JDBC.

In order to choose cross-platform compatibility implementation components, this project utilizes Tomcat as Web server, and MySQL as database server. The other components, such as the web browsers, the system will support all standard web browsers such as Netscape Navigator/Communicator (version 4.0 or higher) or Internet Explorer (version 5.0 or higher), which are dependant on which kind of browsers the customers use.

The architecture of this project is illustrated in the figure 1 below.
Figure 1. System Deployment Diagram
2.1 System Interfaces

The RRS is a architecture that displays the user interface in a Web browser using that handles requests from the client browser and provides access to the a MySQL database via JDBC. The HTTP/HTTPS server is provided by Apache Tomcat, which also implements JSP.

2.2 Hardware Interfaces

RRS will not implement hardware interface directly. However, it will trust the underlying operating system (Windows, NT, ME, XP, Linux, UNIX, MAC) to handle the hardware interfaces.

2.3 Software Interfaces

As explained above, there will be two different software interfaces depending on the type of access that the user is demanding or the function that the user wants the software to perform. The reasons that are why I choose the software interfaces used in this project are summarized as following:

- Operating system (windows XP): This project uses the windows XP because this is the most common browsers now, and it will be easier for people to use.
• Web Server/Container (Jakarta Tomcat Server 5.0.28): Tomcat implements the servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, which provides an environment for Java code to run in cooperation with a web server. Tomcat server is a Java-based Web Application container that was created to run Servlets and JavaServer Pages (JSP) in Web applications.

• JAVA 2 Platform, Standard Edition (J2SE): A Java-based, runtime platform that provides many features for developing Web-based Java applications, including database access (JDBC API) interface technology, and security for both local network and Internet use and it’s required in the Tomcat JAVA Container.

• Database Server (MySQL Server 4.0): MySQL is open source database software. MySQL also provides a JDBC driver to easily connect from a JAVA program.

• Java Database Connector (JDBC): MySQL connector.

• Languages: HTML/JavaScript/JSP.
CHAPTER THREE
DATABASE DESIGN

The database that will be required by RRS will be written in MySQL. Once the database has been created using MySQL, all interfacing with the database will be done from JSP. All interfaces as seen by the users of the system will be through pages with HTML forms generated from JSP.

3.1 Data Analysis
The data for designing and implementing the schema of the database depends on four entities: restaurant table, signup table, rate table, and board table. All the input data will be checked by using JSP.

3.2 Database Model-Relational Diagram
All the entities and relations used in RRS are described in Figure 2 E-R Diagram:
Figure 2. Database Model-Relational Diagram
3.3 Database Model-Relational Schema

The conceptual model ER diagram maps into the following relational table design, which shows in the following tables.

<table>
<thead>
<tr>
<th>Restaurant</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RestName</td>
<td>type</td>
<td>area</td>
<td>phone</td>
<td>address</td>
</tr>
<tr>
<td>zipcode</td>
<td>map</td>
<td>id</td>
<td>foodrate</td>
<td>atmosrate</td>
</tr>
<tr>
<td>ratecount</td>
<td>price</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>signup</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Password</td>
<td>Sex</td>
<td>Email</td>
<td>Age</td>
</tr>
<tr>
<td>ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>zipcode</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>zipcode</td>
<td>lon</td>
<td>lat</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>board</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Poster</td>
<td>PostTime</td>
<td>Content</td>
<td>RepID</td>
</tr>
<tr>
<td></td>
<td>LastRep</td>
<td>RepNum</td>
<td>ConNum</td>
<td>Title</td>
</tr>
</tbody>
</table>

Figure 3. Database Model-Relational Schema
3.4 Data Type and Detail

The logical model establishes the following detailed design in MySQL database. The following tables describe data type, length, primary key, null or non-null keys, and extra information, such as auto_increment.

Table 1. Structure of Table Restaurant

<table>
<thead>
<tr>
<th>field</th>
<th>type</th>
<th>null</th>
<th>key</th>
<th>default</th>
<th>extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>RestName</td>
<td>varchar(30)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(15)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>area</td>
<td>varchar(15)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>Phone</td>
<td>varchar(12)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>address</td>
<td>varchar(50)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>zipcode</td>
<td>Int(5)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>map</td>
<td>varchar(255)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>id</td>
<td></td>
<td></td>
<td>PRI</td>
<td>0</td>
<td>auto_increment</td>
</tr>
<tr>
<td>foodrate</td>
<td>Float(10,2)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>atmosrate</td>
<td>Float(10,2)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>ratecount</td>
<td>Int(11)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>price</td>
<td>Int(11)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
</tbody>
</table>
Table 2. Structure of Table Signup

<table>
<thead>
<tr>
<th>field</th>
<th>Type</th>
<th>null</th>
<th>key</th>
<th>default</th>
<th>extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>varchar(15)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>Password</td>
<td>varchar(30)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>Sex</td>
<td>char(1)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>Email</td>
<td>varchar(30)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>Age</td>
<td>Int(2)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>user_id</td>
<td>Int(11)</td>
<td>PRI</td>
<td>0</td>
<td></td>
<td>auto_increment</td>
</tr>
</tbody>
</table>

Table 3. Structure of Table Board

<table>
<thead>
<tr>
<th>field</th>
<th>Type</th>
<th>null</th>
<th>key</th>
<th>default</th>
<th>extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>int(11)</td>
<td>PRI</td>
<td>0</td>
<td></td>
<td>Auto_increment</td>
</tr>
<tr>
<td>Poster</td>
<td>varchar(20)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>PostTime</td>
<td>date</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Content</td>
<td>text</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RepID</td>
<td>int(11)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>LastRep</td>
<td>time</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RepNum</td>
<td>int(11)</td>
<td>Yes</td>
<td></td>
<td></td>
<td>null</td>
</tr>
<tr>
<td>ConNum</td>
<td>int(11)</td>
<td>Yes</td>
<td></td>
<td></td>
<td>null</td>
</tr>
<tr>
<td>Title</td>
<td>varchar(40)</td>
<td>Yes</td>
<td></td>
<td></td>
<td>null</td>
</tr>
</tbody>
</table>
Table 4. Structure of Table Zipcode

<table>
<thead>
<tr>
<th>field</th>
<th>Type</th>
<th>null</th>
<th>key</th>
<th>default</th>
<th>extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>zipcode</td>
<td>int(11)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>lon</td>
<td>double(16,8)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
<tr>
<td>lat</td>
<td>double(16,8)</td>
<td>Yes</td>
<td></td>
<td>null</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER FOUR

PROJECT IMPLEMENTATION

The design of RRS aims to perform 15 main functions for 3 different prospected users. The following figure is the Use Case Diagram of this project.

Figure 4. Use Case Diagram
4.1 User Interface Design

User interfaces will be established on the web and therefore it can use all user interfaces provided by the web browser including all plug-ins and any added functionality that the browser may possess. This system will support all kinds of standard web browsers such as Netscape Navigator/Communicator (version 4.0 or higher) or Internet Explorer (version 5.0 or higher). There are two types of user interfaces.

(1) Static user interfaces: the interface will be static for all people to browse the site, regardless of the access rights of the person. They will be only in the form of HTML forms or pages that have HTML Data and pictures on them.

(2) Dynamic user interface: the interfaces will be generated dynamically on the server side using JSP. These user interfaces will provide information tailored to the user who is logged in.
4.2 Graphical User Interface

4.2.1 Index Page

This page is the starting page for all people (Guest, Member and Administrator) who are going to be using this software product. Guest and Member should enter from Guest button, and Administrator should enter from Admin button.

Figure 5. Index Page
4.2.2 Guest Main Page

This page has a picture link to search restaurant. This page also reminds guest need to sign in then they have rights to rate. It includes the link for member to login in and the link for guest to sign up.

Figure 6. Guest Main Page
4.2.3 Guest Signup Page

This page is a sign up form. It is designed for the new user who wants to join RRS to rate or recommend some great restaurants. This form includes some information about username, password, sex/gender, email, and age selection.

Figure 7. Guest Signup Page
4.2.4 Guest Complete Signup Page

This page shows successfully signing up message, and it also will show the information that the guest fill out.

Figure 8. Guest Complete Signup Page
4.2.5 Guest Search Restaurant Page

This page gives guest to search restaurants by 3 different ways. It includes: select by a cuisine, select by an area, and search by keying in a restaurant name. Guest can only select by single way, or select by 2 or 3 ways. This page includes an advanced search link.

Figure 9. Guest search Restaurant Page
4.2.6 Guest search restaurant Page

This page gives information about the restaurants that guest search by his/her specific selections. This page will show the restaurant name, type, area, food rating, atmosphere rating, and a rating link. If guest is interested in some of the restaurant results, guest can click the restaurant name button to view the restaurant detail. If guest want to give rating, guest can click the rate button. But guest don’t have the right to rate, it will forward to a page regarding the login information.

Figure 10. Guest Search Restaurant Result Page
4.2.7 Guest Rate Page

This page shows the information about only member can rate. It has the login link for guest to login.

Figure 11. Guest rate Page
4.2.8 Guest view Restaurant Detail Page

This page gives information about the restaurant. It includes the restaurant name, type, area, address, phone, and map link. It will open a new window for the map link to show how to get there.

Figure 12. Guest View Restaurant Detail Page
4.2.9 Guest View Map Page

When guest click the map link, it will open a new window of a map. It shows streets near the restaurant to let people know how to get there.

Figure 13. Guest View Map Page
This page gives advanced search selections about the restaurant. It includes 3 different ways.

- First one: it includes 3 selections. Search by zip code, search by miles, and search by price. For example, guest can search by entering a zip code, selecting a range of miles near that zip code like 10 miles, and choosing a range of price like 20-30$ to induce the searching range.

- Second one: search by rating. It includes 2 selections, search by food rating and/or search by atmosphere rating.

- Third one: search by recommender. In recommendations page, it will show the recommendations by members. If guest is interested in some specific members' recommendations, guest can search here to view all their post.
Figure 14. Guest Advanced Search Page
4.2.11 Guest Search Distance Page

This page gives information about the restaurants that is in the range of the distance. It will include the restaurant name link, area, price, food rating, atmosphere rating, and a rating link.

Figure 15. Guest Search Distance Page
4.2.12 Guest Search Rating Page

This page gives information about the restaurants that is in the range of the rating. It will include the restaurant name link, type, area, food rating, atmosphere rating, and a rating link.

Figure 16. Guest Search Rating Page
4.2.13 Guest Search Recommender Page

This page gives information of the specific poster’s post. In recommendations page, it will show the recommendations by members. If guest is interested in some specific members’ recommendations, guest can search here to view all their post.

Figure 17. Guest Search Recommender Page
4.2.14 Guest View Recommendations Page

This page is divided into 2 frames. The top frame includes the RRS logo, the title of post and a link of post. The button frame shows the content of post. Guest clicks the title then the content details will appear on the button frame.

- **RRS logo:** it has a link to back to main page.
- **Title:** this form includes title link, post date, poster, reply number, viewed number, and last reply time.
- **Content:** it shows the detail of poster name, post date, title, and content. It also has a reply link for other member's replying.
### Figure 18. Guest View Recommendations Page

<table>
<thead>
<tr>
<th>Title</th>
<th>Date</th>
<th>Poster</th>
<th>Reply</th>
<th>View</th>
<th>Last reply</th>
</tr>
</thead>
<tbody>
<tr>
<td>outlet</td>
<td>2006-04-10</td>
<td>Jemmy</td>
<td>0</td>
<td>2</td>
<td>2006-04-10 02:20:20</td>
</tr>
<tr>
<td>great sushi in beverly hills</td>
<td>2006-03-28</td>
<td>ma</td>
<td>0</td>
<td>7</td>
<td>2006-03-28 06:23:34</td>
</tr>
<tr>
<td>www</td>
<td>2006-03-23</td>
<td>ma</td>
<td>0</td>
<td>5</td>
<td>2006-03-23 01:10:46</td>
</tr>
</tbody>
</table>

**Postermila**  
Date: 2006-03-28  
Title: great sushi in beverly hills  
**Content:**  
I wanna recommend a great sushi in beverly hills, it taste like hawaii-japan style, but not cheap...

**Reply**
4.2.15 Guest Post Error Page

This page gives information about post error messages. Only members can post. There is a signup link for guest to applying an account. After successfully signing up, guest turn into member, he or she can give recommendations to other people.

Figure 19. Guest Post Error Page
4.2.16 Guest Rate Page

This page shows a form of rating stars of the restaurants which have been rated. It includes the information of restaurant link, food rating, and atmosphere rating. Below the form, there is a restaurant search link for people to search specific restaurants to rate.

Figure 20. Guest Rate Page
4.2.17 Guest View Rated Restaurant Page

It includes the information of restaurant name, type, rate link, area, address, phone, food rating, atmosphere rating, total rate, and a map link.

Figure 21. Guest View Rated Restaurant Page
4.2.18 Member Login Page

This page let members to login. If the input data is incorrect, the browser will display an appropriate error message. If members forget password, members can click the forget password button to ask for their password. If He/ or she is not a member yet, he or she can click the signup button to fill up a form to apply a new account.

Figure 22. Member Login Page
4.2.19 Member Forget Password Page

This page will tell member to enter correct account name and email, after they are checked by database and correctly, then the password will be shown in next page.

Figure 23. Member Forget Password Page
4.2.20 Member Password Return Page

This page shows a password sending message. The system will check if the username and email is correct. If both of them are correct with the database, the system will collect a random number as a temporary password. Members can check their email and get the password to login.

Figure 24. Member Password Return Page
4.2.21 Member Main Page

After successfully login, the page will be forwarded to main page. This page has picture links. Members can click it to search restaurants, rate restaurants, and give restaurants recommendations directly.

Figure 25. Member Main Page
4.2.22 Member Check Account Page

Member check account page has two links: my account and my history. My account will show information of member’s sign up form. My history will show member’s recommendation history.

Figure 26. Member Check Account page
4.2.23 Member Edit Account Page

Member edit account page let member to modify the information of account name, password, sex/gender, email, and age selection. The back button will forward to member check account page.

Figure 27. Member Edit Account Page
4.2.24 Member Account Modified Page

This page shows member's information which is modified. It also includes those information that will never be changed. Back to my account button will forward to member check account page.

![Figure 28. Member Account Modified Page](image_url)

Figure 28. Member Account Modified Page
4.2.25 Member My History Page

This page shows member’s recommendations history. It appears in form. The form includes the fields of username, post time, title, and content. The back button will go back to member check account page.

Figure 29. Member My History Page
4.2.26 Member Restaurant Search Page

This page gives member to search restaurants by 3 different ways. It includes: select by a cuisine, select by an area, and search by keying in a restaurant name. Member not only can select by a single way, but also can select by 2 or 3 ways. This page includes an advanced search link.

Figure 30. Member Restaurant Search Page
4.2.27 Member Restaurant Result Page

This page gives information about the restaurants that member search by his/ or her specific selections. This page will show the restaurant name, type, area, food rating, atmosphere rating, and a rating link. If member is interested in some of the restaurant results, member can click the restaurant name button to view the restaurant detail. If member is interested in some of the restaurant results and want to give rating, member can click the rate button.

Figure 31. Member Restaurant Result Page
4.2.28 Member View Restaurant Page

This page gives information about the restaurant. It includes the restaurant name, type, area, address, phone, and map link. It will open a new window for the map link to show how to get there.

Figure 32. Member View Restaurant Page
4.2.29 Member View Map Page

When member click the map link, it will open a new window of a map. It shows streets near the restaurant to let people know how to get there.

Figure 33. Member View Map Page
4.2.30 Member Advanced Search Page

This page gives advanced search selections about the restaurant. It includes 3 different ways.

- First one: it includes 3 selections. Search by zip code, miles, and price. For example, guest can input a zip code, and select a range of miles near that zip code like 10 miles, and choose a range of price like 20-30$.

- Second one: search by rating. It includes 2 selections, search by food rating and/or search by atmosphere rating.

- Third one: search by recommender. If member is interested in some specific members' recommendations, member can search here to view all their posts.
Figure 34. Member Advanced Search Page
4.2.31 Member Distance Search Page

This page gives information about the restaurants that is in the range of the distance. It will include the restaurant name link, type, area, food rating, atmosphere rating, and a rating link.

Figure 35. Member Distance Search Page
4.2.32 Member Ratings Search Page

This page gives information about the restaurants that is in the range of the rating. It will include the restaurant name link, type, area, food rating, atmosphere rating, and a rating link.

Figure 36. Member Ratings Search Page
4.2.33 Member Poster Search Page

This page gives information of the specific poster's post. In recommendations page, it will show the recommendations by members. If member is interested in some specific members' recommendations, he /or she can search here to view all their post.

Figure 37. Member Poster Search Page
4.2.34 Member Recommendation Page

This page is divided by 2 frames. The top frame includes the RRS logo, the title of post and a link of post. The bottom frame shows the content of post. Member clicks the title then the content details will appear on the button frame.

- RRS logo: it has a link to back to RRS main page.
- Title: this form includes title link, post date, poster, reply number, viewed number, and last reply time.
- Content: it shows the detail of poster name, post date, title, and content. It also has a reply link for other member's replying.
Figure 38. Member Recommendation Page
4.2.35 Member Post Page

Members can post their recommendations here. This page is on the button of frame. Members need to fill up in the text box, and input the article’s title and content then click post it will be submitted, or click clear then you can input the text again.

Figure 39. Member Post Page
4.2.36 Member Reply Page

This page shows on the button of frame. The title will set as Re: title name. It will include the content that you want to reply. Members need to fill up in the text box, and input the content then click post button, and it will be submitted, or click clear then you can input the text again.

Figure 40. Member Reply Page
4.2.37 Member Rating Page

This page shows a form of rating stars of the restaurants which have been rated. It includes the information of restaurant link, food rating, and atmosphere rating. Below the form, there is a restaurant search link for people to search specific restaurants to rate.

![Figure 41. Member Rating Page](image_url)
4.2.38 Member View Rating Page

It includes the information of restaurant name, type, rate link, area, address, phone, food rating, atmosphere rating, total rate, and a map link.

Figure 42. Member View Rating Page
4.2.39 Member Give Rate Page

This page gives 2 choices: rate by food and/or rate by atmosphere. The rating is from 1 (low) to 5 (high), then click rate button, it will be submitted and the page will forward to recently rated page.

Figure 43. Member Give Rate Page
4.2.40 Administrator Login Page

This page is a login page for administrator. Administrator need to input the correct username and password then he/ or she can enter the site to manage RRS system.

Figure 44. Administrator Login Page
4.2.41 Administrator Main Page

After administrator successfully login, next page will be the main page. This page has a welcome administrator message in the center. There are 5 functions on the left bar for administrator to manage the RRS. In the center of the page, there are picture links to enter those manage pages include edit customer, edit restaurants, edit recommendations, and edit rate.

Figure 45. Administrator Main Page
4.2.42 Administrator Edit Member Page

Edit customer page has 2 ways to check member's username. One is to click the link of all members. Another one is to input the username. After database catch it, next page will goes to member's account information.

Figure 46. Administrator Edit Member Page
4.2.43 Administrator Check Member Page

This page will list members' entire user ID which is saved in the database. The user ID will be listed as a link in alphabetical order. Administrator can click the username link to edit member's account.

Figure 47. Administrator Check Member Page
4.2.44 Administrator Delete Member Page

While Administrator click the delete button, this page will pop out a new window of confirming delete message. After the member is been deleted, the user ID will be automatically deleted from the database. Once the system completes the process, then the next page will go back to edit member’s page.

Figure 48. Administrator Delete Member Page
4.2.45 Administrator Edit Restaurant Page

It includes add new restaurants and edit restaurants.

• Add restaurants: Administrator can click the add restaurants link to add a new restaurant.

• Edit Restaurants:
  1. Search all restaurants: Administrator can click all restaurants to get all restaurants and the list is in alphabetical order.
  2. Search by 4 selections: Administrator can search restaurants by 4 selections; it includes select by area, select by type, select by zip code, and select by name.
Figure 49. Administrator Edit Restaurant Page
4.2.46 Administrator Add Restaurant Page

This page is a form for administrator to fill out the restaurant's information; the column includes restaurant name, type selections, area selections, address, phone, zip code, price, and map. If the input data is incorrect, the browser will display an appropriate error message. Otherwise, the RRS will generate a new account and the database will update it.

Figure 50. Administrator Add Restaurants Page
4.2.47 Administrator Restaurants Page

This page will list all of the restaurants in database. The list will show the restaurant name and area. Administrator can click the link of restaurant name to view or edit the restaurant.

Figure 51. Administrator Restaurants Page
4.2.48 Administrator Restaurant Page

After administrator selects a function to search restaurant, it will show the search result. This page will show the column of restaurant name, type, and area. Administrator can click the link of restaurant name to view or edit it.

![Restaurant search result](image)

Figure 52. Administrator Restaurant Page
4.2.49 Administrator View Restaurant Page

This page shows the restaurant information, includes the restaurant name, type, area, phone, zip code, map, price, food rate, and atmosphere rate. It also has links of delete and edit. Or administrator can click back button to back to the edit restaurant page.

Figure 53. Administrator View Restaurant Page
4.2.50 Administrator Delete Page

While administrator click delete button, it will pop up a window to confirm the delete message. After the restaurant is been deleted, the page will goes back to edit restaurant page.

Figure 54. Administrator Delete Page
4.2.51 Administrator Edit Restaurant Page

This page will output the restaurant information, includes the restaurant name, type, area, phone, zip code, map, price, food rate, and atmosphere rate. Administrator can modify in those columns or click check others button to back to the edit restaurant page.

Figure 55. Administrator Edit Restaurant Page
4.2.52 Administrator Modified Page

This page will output the restaurant information those have been modified, includes the restaurant name, type, area, phone, zip code, map, price, food rate, and atmosphere rate. Administrator can click back to search button to back to the edit restaurant page.

Figure 56. Administrator Modified Page
4.2.53 Administrator Edit Posting Page

Administrator can edit recommendation by 2 functions: One is check all recommendations. Another one is check by selections; it includes selections of area, type, zip code, and name.

Figure 57. Administrator Edit Posting Page
4.2.54 Administrator View Posting Page

This page will show all recommendations by title and poster. Administrator can click the title or poster link to check or edit the content. Or administrator can click the back button to back to the edit recommendation page.

Figure 58. Administrator view Posting Page
4.2.55 Administrator Search Title Page

After administrator click the recommendation’s title link to view the detail. It will forward to this page, and it shows all the detail of this title. It includes title, poster, post time, content, reply number, and last reply time. Administrator can click delete button to delete it.

Figure 59. Administrator Search Title Page
4.2.56 Administrator Delete Post Page

After administrator click delete button, it will go to this page which shows a deleting confirm message. The title of recommendation will be automatically deleted from the database. Once the system completes the process, it will forward the administrator to edit recommendation page again.

Figure 60. Administrator Delete Post Page
4.2.57 Administrator Search Poster Page

After administrator click the recommendation’s poster link to view the detail. It will forward to this page, and it shows this entire poster’s recommendation. It includes title, poster, post time, content, reply number, and last reply time. Administrator can click delete button to delete it.

Figure 61. Administrator Search Poster Page
CHAPTER FIVE
SYSTEM VALIDATION

5.1 Introduction

To maintain the consistent functionality of designed system, here is an extensive test plan to be conducted throughout the design and implementation process.

The system test is conducted in three phases:
1. Unit Test Plan
2. Subsystem Test Plan
3. System Test Plan

5.2 Unit Test Plan

The Unit test presents the basic level of testing on all the individual components. Following are the Step wise procedure for unit testing:
Table 5. The Unit Test Results

<table>
<thead>
<tr>
<th>Unit Test</th>
<th>Tests Performed</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guest Main page</td>
<td>• Check the correctness of the displayed data in JSP included pages components.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Verify the login and new user link working properly.</td>
<td></td>
</tr>
<tr>
<td>Guest Login page</td>
<td>• Verify the login function working and get the error message properly.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Check entire links working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Test the correctness of the validation for all input data.</td>
<td></td>
</tr>
<tr>
<td>Forget Password Page</td>
<td>• Check the correctness of the displayed data.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Check all the buttons and links working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Check JavaScript function working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page get the correct member information.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page can sent email by the message.</td>
<td></td>
</tr>
<tr>
<td>Member’s Main Page</td>
<td>• Check the correctness of the displayed data in JSP included pages components.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Check the buttons and links working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the user save in session after login.</td>
<td></td>
</tr>
<tr>
<td>Member Manage</td>
<td>• Verify the page get the correct user information.</td>
<td>OK</td>
</tr>
<tr>
<td>Account Page</td>
<td>• Verify all links and buttons working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Test the correctness of the validation for all input data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page can get the error message and working properly by the message.</td>
<td></td>
</tr>
<tr>
<td>Recommendation</td>
<td>• Check the correctness of the displayed data.</td>
<td>OK</td>
</tr>
<tr>
<td>Page</td>
<td>• Check the entire buttons and links working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Test the correctness of the validation for all input data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page can get the error message and working properly by the message.</td>
<td></td>
</tr>
<tr>
<td>Rate Page</td>
<td>• Check the correctness of the displayed data.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Test the link working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Check JavaScript function working</td>
<td></td>
</tr>
<tr>
<td>Unit Test</td>
<td>Tests Performed</td>
<td>Results</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>---------------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Administrator Main Page</td>
<td>• Check the correctness of the displayed data in JSP included pages components.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Check all the links working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the user save in session after login.</td>
<td></td>
</tr>
<tr>
<td>Add Restaurant Page</td>
<td>• Check the entire buttons, link and selection list working properly.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Test the correctness of the validation for all input data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Check JavaScript function working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page can get the error message and working properly by the message.</td>
<td></td>
</tr>
<tr>
<td>Delete/Edit Restaurant Page</td>
<td>• Check the correctness of the displayed data.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Check button, links and check boxes working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Check JavaScript function working properly.</td>
<td></td>
</tr>
<tr>
<td>Edit Restaurant Information page</td>
<td>• Verify the page get the correct restaurant information.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Verify all links and buttons working as expected.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Test the correctness of the validation for all input data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page can get the error message and working properly by the message.</td>
<td></td>
</tr>
<tr>
<td>Delete/Edit Member Account Page</td>
<td>• Check the correctness of the displayed data.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Check button, links and check boxes working properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Check JavaScript function working properly.</td>
<td></td>
</tr>
<tr>
<td>Edit Member Account Information page</td>
<td>• Verify the page get the correct member information.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Verify all links and buttons working as expected.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Test the correctness of the validation for all input data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page can get the error message and working properly by the message.</td>
<td></td>
</tr>
<tr>
<td>Edit Recommendation page</td>
<td>• Verify the page get the correct information.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Verify all links and buttons working as</td>
<td></td>
</tr>
</tbody>
</table>
## 5.3 Subsystem Test Plan

Subsystem testing is the next step up in the testing process where all related units from a subsystem does a certain task. Thus, the subsystem test process is useful for detecting interface errors and specific functions. Table 24 show subsystem test results in detail.

Once the entire unit test plan successfully conducted there is need for Integration test to check the validity of the system by merging the two modules separately.

<table>
<thead>
<tr>
<th>Unit Test</th>
<th>Tests Performed</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>expected.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Test the correctness of the validation for all input data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page can get the error message and working properly by the message.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Check JavaScript function working properly.</td>
<td></td>
</tr>
<tr>
<td>Edit Rate page</td>
<td>• Verify the page get the correct information.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Verify all links and buttons working as expected.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Test the correctness of the validation for all input data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the page can get the error message and working properly by the message.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Check JavaScript function working properly.</td>
<td></td>
</tr>
</tbody>
</table>
Table 6. Subsystem Test Results

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Tests Performed</th>
<th>Results</th>
</tr>
</thead>
</table>
| Sign up Subsystem              | • Make sure the subsystem checks the guest information before create a new account for him/her.  
• Check if the subsystem can detect the error of creating of the user that exists in the subsystem                                                                                       | OK      |
| Authorize Subsystem            | • Test if it can get the error message.  
• Make sure the result of authorizing user is correct.  
• Verify the login user information is store in session properly.  
• Verify the login page redirect to the correct browsing or editing page after the user logins in.                                                                 | OK      |
| Accounts Management Subsystem  | • Make sure all the existing users are list in the user list.  
• Check if the subsystem can detect the error of creating of the user that exists in the subsystem.  
• Check if the user can update personal account properly.  
• Verify the created user information is the same as the information provided.  
• Verify the subsystem can delete a user account properly.  
• Make sure the password query function can working properly.                                                                                                                            | OK      |
| Editing Account Subsystem      | • Make sure the subsystem checks the user privilege before forwarding to edit page.  
• Verify the subsystem check the user privilege before update the page information.  
• Verify if the subsystem shows the page properties is the users are the owner or the administrator.                                                                                   | OK      |
| Restaurant Management Subsystem| • Make sure all the existing restaurants are list in the restaurants list.  
• Check if the subsystem can detect the error of creating of the restaurants that exists in the subsystem.  
• Check if the administrator can update restaurants information properly.  
• Verify the created restaurants information is the same as the information provided.  
• Verify the subsystem can delete a restaurant properly.                                                                                                                              | OK      |
<p>| Edit                           | • Make sure the subsystem checks the                                                                                                                                                                         | OK      |</p>
<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Tests Performed</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restaurant Subsystem</td>
<td>restaurants information before forwarding to edit page.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify the subsystem check the restaurant information before update the page information.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Verify if the subsystem shows the restaurant information page properties.</td>
<td></td>
</tr>
<tr>
<td>Edit recently rated</td>
<td>• Verify the subsystem check the rated restaurants information before update the page information.</td>
<td>OK</td>
</tr>
<tr>
<td>Restaurant Subsystem</td>
<td>• Make sure all the rated restaurants titles are list in the rated restaurants list.</td>
<td></td>
</tr>
<tr>
<td>Browsing Subsystem</td>
<td>• Check if the subsystem checks for user privilege before showing pages.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>• Verify the page is showing properly after the user click on the page link.</td>
<td></td>
</tr>
</tbody>
</table>

5.4 System Test Plan

This is the test plan to be conducted after designing the every module involved in the system design. System test plan is a test process that uses real data, which the system is intended to manipulate, to test the system. First of all the subsystem will be integrated into one system. Then test the system by using a variety of data to see the overall results.

The steps for the system test plan are showed in the following table:
Table 7. System Test Results

<table>
<thead>
<tr>
<th>System Test</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Install the RRS into server.</td>
<td>OK</td>
</tr>
<tr>
<td>2. Start up all services such as JSP engine, MySQL database engine.</td>
<td>OK</td>
</tr>
<tr>
<td>3. Running testing by using real data on all forms and reports.</td>
<td>OK</td>
</tr>
</tbody>
</table>
CHAPTER SIX
MAINTENANCE MANUAL

6.1 Software Installation

The system must be portable. The application must not be tied down to a single platform or operating system vendor. The system must have the ability to be installed and run with no performance loss under standard servers including: MS Windows, all versions of UNIX and Linux operating systems.

In RRS, it requires MySQL, J2SDK, TOMCAT, and JDBC to run the programs. Following will detail the installation of those four software systems.

6.1.1 Install Java Platform

(1) Run the installation program for J2SE SDK, accept the default location for installation, and deselect the registration checkbox to save time.

(2) Add an environmental variable called JAVA_HOME that is set to the path to your jdk installation. For instance, JAVA_HOME = C:\j2sdk1.4.2_07.

(3) To do this in Windows XP, go to Control Panel, run system, select the advanced tab, and click the environment variables button. Click on New under user variables, and set a variable with name JAVA_HOME to value C:\j2sdk1.4.2_07.
6.1.2 Install Tomcat

(1) Download the appropriate version of Tomcat

Download the appropriate version of Tomcat 5.0 from the Apache website. http://tomcat.apache.org/

Either grabs the ZIP or tar.gz version. Most Windows users should just grab the ZIP version.

(2) Uncompress the file.

This should create a new subdirectory named jakarta-tomcat.

(3) Set up

Set the environment variable JAVA_HOME to point to the root directory of your JDK hierarchy. Be sure the Java interpreter is in your PATH environment variable.

Something like the following will work for Windows, depending upon the JDK installation directory:

SET JAVA_HOME=C:\jdk1.2.2

(4) Change to the bin directory and start Tomcat using the command-line command startup.

(5) Running Tomcat using the command-line command startup should produce output similar to the following:

myhost> startup

Tomcat Web Server Version 5.0.28
Loaded configuration from: file:E:/tomcat/server.xml
Configuring web service using "default"
Configuring web service using
"file:E:/tomcat/examples/WEB-INF/web.xml"

default: init
jsp: init

Configuring web service using "default"

Configuring web service using
"file:E:/tomcat/webpages/WEB-INF/web.xml"

default: init
jsp: init

Starting tcp endpoint on 8080 with
org.apache.tomcat.service.http.HttpConnectionHandler

Starting tcp endpoint on 8007 with
org.apache.tomcat.service.connector.Ajp12ConnectionHandle

6.1.3 Install Security

The description below uses the variable name
$CATALINA_HOME to refer to the directory into which you have
installed Tomcat 5, and is the base directory against which
most relative paths are resolved. To install and configure
SSL support on Tomcat 5, you need to follow these simple
steps. For more information, read the rest of this HOW-TO.

(1) Create a certificate keystore by executing the
following command: %JAVA_HOME%\bin\keytool -genkey -keyalg
RSA -keysize 2048 -dname CN=localhost -alias tomcat -keystore C:\keystore -keypass changeit -storepass changeit

(2) You can find a file named keystore in c drive, and copy it to C:\Program Files\Apache Software Foundation\Tomcat 5.0\conf

(3) Copy the following into tomcat\conf\server.xml

```xml
<Connector port="8443"
  maxThreads="150"
  minSpareThreads="25"
  maxSpareThreads="75"
  enableLookups="false"
  disableUploadTimeout="true"
  acceptCount="100"
  debug="0"
  scheme="https"
  secure="true"
  clientAuth="false"
  sslProtocol="TLS"
  keystore="conf/keystore" />
```

(4) Uncomment the "SSL HTTP/1.1 Connector" entry in $CATALINA_HOME/conf/server.xml and tweak as necessary.

6.1.4 Install Database

(1) Download the appropriate version of MySQL
Download the appropriate version of MySQL 4.0 from the MySQL website. http://mysql.com/

(2) Uncompress the file, and run the file mysql-essential-4.1.12-win32.msi.

Accept default settings except for the following:
Select Skip Sign-Up (to save time).
Select Best Support For Multiligialism
Select Include Bin Directory in Windows PATH
Set the root account password.
6.2 Variable Modifications

In RRS, we have to change some environment variables in the server.xml in Tomcat server configuration.

6.2.1 System Variables

1. Open the file "server.xml" in the directory 
   ${TOMCAT_HOME}/conf" via text editor.

2. Scroll down until you see the context area we added in at chapter 5.4.1.

3. The variable "path" in Context indicates the context path of the web application. The default value would be "/RRS."

4. The variable "docBase" in Context is the files directory for the web application. The default value would be "/home/RRS/web."

5. The parameter "contextPath" indicate the context path for RRS which would be the same as the value of path.
6.3 System Installation/Migration

1. All the JSP programs and HTML programs are stored in Tomcat5.0\webapps\RRS\web
2. All the classes are stored in Tomcat5.0\webapps/RRS/web/WEB-INF/classes
3. All the pictures are stored in Tomcat5.0\webapps\RRS\web\images
4. Place the web.xml for RRS in Tomcat5.0\webapps\RRS\web\WEB-INF

6.4 Backup and Restore

Data backups should be logically scheduled and completed as so onto reliable sources. To protect system information is one of the administrator’s most important tasks. Backup allows the administrator to restore a file system in the event of failure or loss of the original. This manual is useful when system need to be ported in different machines and server. The following steps should be followed carefully and on a strict schedule. There are two steps to back up RRS. First step is backup the system files. Second step is backup the database which is used by RRS. Third step is re-installation manual.
6.4.1 System Backup

All the RRS system files are stored in the directory "Tomcat5.0\webapps\RRS\web" and the subdirectory of its. Thus, in order to backup the system files, we can compress this directory by using the compress program rar to backup the system files:
right click on the directory and choose add to the archive, then select rar as archive format.

6.4.2 Database Backup

1. mysql\dump --opt --user=username --password database > dumpfile.sql

2. Edit the dump file and put these lines at the beginning:
   SET AUTOCOMMIT = 0;
   SET FOREIGN_KEY_CHECKS=0;

3. Put these lines at the end:
   SET FOREIGN_KEY_CHECKS = 1;
   COMMIT;
   SET AUTOCOMMIT = 1;

4. mysql --user=username --password database < dumpfile.sql

Once above command is entered in shell script carefully, the database will be stored in the file which is
specified in the command line. This database can be evoked and stored in the CD.

6.4.3 System Restore

To restore the system file, simply extract the backup file by using the WinRAR program. Right click on the backup file and choose "unzip to" then you select the directory that the system file should be.

6.4.4 Database Restore

To restore the database needed for the RRS, restore from FILENAME.mysql is a three step process:

- Drop the database
  
  mysqladmin -u USERNAME -p drop RRS

- Recreate the database
  
  mysqladmin -u USERNAME -p create RRS

- Import the backup data
  
  mysql -u USERNAME -p RRS < FILENAME.mysql

6.4.5 Re-installation Manual

1. Install Tomcat and setup it (5.0.28)
2. Install j2sdk (as a develop tool) (1_4_2_07_window-i586)
3. Install MySQL (4.0)
4. Install the JDBC driver.
5. Install the MySQL table to the MySQL database.
   
   (mysql -u root -p < RRS.sql)
6. Copy all the files and subdirectories under ROOT and database Info to the Tomcat. (We use default, therefore, they are
/usr/Jakarta-tomcat-5.0.28/webapps/ROOT
/usr/Jakarta-tomcat-5.0.28/webapps/databaseInfo)

7. Setup the dbProps.properties file under
/....../webapps/databaseInfo stated above according to the computer name and the root user password of MySQL.
(We use default, therefore, the directory is
/usr/Jakarta-tomcat-4.1.31/webapps/databaseInfo
and the content is:

```
url = jdbc:mysql://localhost/RRS
user = root
password = root
driver = com.mysql.jdbc.Driver
```
CHAPTER SEVEN

CONCLUSION AND FUTURE DIRECTIONS

7.1 Conclusion

The RRS main functions include:

(1) Search: You can search restaurants by restaurant name, type, area, price, rating, recommendation, and distance. Searching by many ways will be easier for you to find the exactly restaurant you interest.

(2) Rate: This function helps you to see the restaurant’s grade clearly from people. You can check it by food ratings and atmosphere ratings. Also you can give ratings for your opinions to people.

(3) Recommendation: This function is like a bulletin. You can leave a message about the restaurant you would like to recommend or a message of the question about the restaurant. Also you can reply the article you interested.

The objective of the RRS project is to let people check great restaurants through the website. They can get the restaurant information and price here, so that they don’t need to call restaurants or check the yellow page for
detail. They can find great restaurants by people's rating or recommendation to get suggestions. Those are from people's experience and compare. In the other way, people try new restaurants from others' suggestions, the restaurant get more customers. It is an encouragement for the great restaurants.

7.2 Future Direction

To build more friendly graphical interfaces, the next goal for a further project is to improve the performance of system and improve the member's benefit such as providing more functions for member, like online reservation function, online order menu, and website, etc.
REFERENCES


