Travel Guide on Palm
Hotel Room Reservation System
(TgoP – HRRS)

Submitted by
WONG PAK LIANG
WEK990022

Supervisor
Dr. Mazliza Othman

Moderator
Cik Rafidah Mohd. Noor

Submission date
24 January 2002

Faculty of Computer Science and Information Technology
University of Malaya
2001/2002

Abstract

Since the released of the Pilot 1000 by Palm Computing in 1996, the Pilot's market grew faster than any other computing product in history. As an extension of desktop system, this handheld computing device allows users to view and enter small amounts of data while away from the desktop.

In Malaysia, they are a lot of website, which provide information in travel. However, from the study we have done, there is no significant number shown that any of the website provide service on Palm. This study has bring us the idea to develop a website or more accurately a web clipping solution for the travel guide around Klang Valley on Palm, Travel Guide on Palm (TGoP).

The TGoP is divided into two subsystems, which are Information Sub System and Hotel Room Reservation System. This report focuses on the Hotel Room Reservation System (HRRS).

The HRRS provides an online room reservation service which mainly focus on the Palm users. The users may retrieve hotel information from the HRRS and makes a reservation via the wireless network.

The HRRS development is divided into four parts, which are client section, portal administrator section, hotel administrator section and officer section. The client is the person who be serviced. The portal administrator maintains the HRRS to make sure the services are available at any time. The hotel administrator control the officer to make sure the room reservation service is offered all the time.

The successful of the TGoP – HRRS will widen the usage of the Palm OS device and at the same time encourage the tourism industry in Malaysia.

Acknowledgment

First and foremost, I would like to express my deepest gratitude to my supervisor, Dr. Mazliza Othman for her guidance, advices and encouragements throughout the entire project. Thanks to Cik Rafidah Mohd. Noor for her useful and informative feedbacks to the project.

I please to thank my project partner, Mr. Chung Yun Liang for his cooperation and sharing his crucial thought throughout the project. Besides that, special thanks also the other fellow course mates for sharing their knowledge and ideas.

I would like to thank Mr. Adrian Wong, the vice president of marimari.com who willing to allocate his time for an interview with us. A warmest thank to Mr. Lee Kee Beng who has shared his knowledge about Palm.

Last but not the least, thanks to my lovely family for their supports and encourages.

Table of Contents

ABSTRACT I

ACKNOWLEDGMENT II

TABLE OF CONTENTS III

LIST OF FIGURES XIII

LIST OF TABLES XVII

Chapter 1: Introduction 1

1.1 PROJECT INTRODUCTION 2

1.2 PROJECT OBJECTIVES 2

1.3 PROJECT SCOPE 3

1.4 PROJECT LIMITATIONS 3

1.5 PROJECT EXPECTED OUTCOME 4

1.6 PROJECT SCHEDULE 6

Chapter 2: Literature review 7

2.1 TOURISM INDUSTRY IN MALAYSIA 7

2.2 THE PALM COMPUTING PLATFORM 8

2.2.1 **Palm OS Philosophy** 8

2.2.2 **Connecting to the Desktop** 9

2.2.3 **Communicating with other devices** 9

2.3 PALM.NET SYSTEM 10

2.3.1 **Palm.Net network** 11

2.3.1.1 Palm Web Clipping Proxy Server 12
2.3.1.2 UDP 12
2.3.1.3 Compressed HTML 13
2.3.1.4 Security 14

2.3.2 **Palm Query Application** 15

2.4 EXISTING HOTEL ROOM RESERVATION SYSTEM REVIEW 16

2.4.1 **Hotel room reservation system on Palm** 16

2.4.2 **Hotel room reservation system on desktop** 17

2.4.2.1 Marimari.com 17
2.4.2.2 Hotelplatform.com 19

2.4.3 **Comparison of marimari.com and hotelplatform.com** 19

Chapter 3: System Analysis and Requirements Specification 21

3.1 SYSTEM DEVELOPMENT MODEL 21

3.2 PROJECT DEFINITION 23

3.3 SYSTEM REQUIREMENTS 24

3.3.1 **Functional Requirements** 24

3.3.1.1 Portal Administrator Module 24

3.3.1.2 Hotel Administrator Module 25
3.3.1.3 Officer Module 26
3.3.1.4 Client Module 27

3.3.2 Non-functional Requirements 28

3.4 DEVELOPMENT TOOLS SPECIFICATION 30

3.4.1 Server Hardware and Software specifications 30

3.4.1.1 Web Server technology 30
3.4.1.2 Database management system technology 30
3.4.1.3 Data access technology 30

3.4.2 Client Hardware and Software specifications 31

3.4.3 Administrators and Officer Hardware and Software specifications 31

3.4.4 Programming language 31

3.4.5 Other development tools 33

Chapter 4: System Design 34

4.1 SYSTEM ARCHITECTURE 34

4.2 SYSTEM STRUCTURE 35

4.3 SYSTEM FLOW CHART 38

4.3.1 Client section 38

4.3.2 Portal Administrator section 39

4.3.3 Hotel Administrator section 40

4.3.4 Officer section 41

4.4 DATA FLOW DIAGRAM 42

4.4.1 Client section DFD 43

4.4.2 Client section level 1 DFD – Reservation process 44

4.4.3 Client section level 1 DFD – Reservation cancellation process 45

4.4.4 Portal Administrator section DFD 45

4.4.5 Hotel Administrator section DFD 46

4.4.6 Officer section DFD 47

4.5 DATABASE DESIGN 48

4.5.1 Client table 48

4.5.2 Month table 49

4.5.3 CreditCard table 49

4.5.4 Prof table 49

4.5.5 Reservation table 50

4.5.6 City table 51

4.5.7 State table 51

4.5.8 Country table 51

4.5.9 Grade table 52

4.5.10 Location table 52

4.5.11 Hotel table 53

4.5.12 Room table 53

4.5.13 Backenduser table 55

4.6 USER INTERFACE DESIGN 56

4.7 DISCUSSION 58

Chapter 5: System Implementation 59

5.1 IMPLEMENTANTATION ON MVC DESIGN PATTERN 59

5.1.1 Model 59

5.1.2 View 59

5.1.3 Controller 59

5.2 WEB SERVER CONFIGURATION 60

5.3 DATABASE IMPLEMENTATION 61

5.4 SSL IMPLEMENTATION 62

5.5 JAVAMAIL IMPLEMENTATION 63

5.6 IMPLEMENTATION ON PALM 65

5.7 MODULES IMPLEMENTATION 66

5.7.1 Client Section 66

5.7.1.1 Sign Up Module 66
5.7.1.2 Login Module 67
5.7.1.3 Reservation Module 67
5.7.1.4 Password Tracking Module 68

5.7.1.5 User Profile Update Module 68
5.7.1.6 Logout Module 68

5.7.2 Backend User Section 69

5.7.2.1 Backend User Login Module 69
5.7.2.2 Backend User Profile Update Module 69
5.7.2.3 Backend User Logout Module 69

5.7.3 Portal Administrator Section 70

5.7.3.1 Portal Administrator Administration Module 70
5.7.3.2 Hotel Administration Module 70
5.7.3.3 Hotel Administrator Administration Module 70
5.7.3.4 Client Administration Module 71

5.7.4 Hotel Administration Section 71

5.7.4.1 Officer Administration Module 71

5.7.5 Officer Section 71

5.7.5.1 Hotel Information Updating Module 71
5.7.5.2 Room Administration Module 71
5.7.5.3 Reservation Administration Module 72

5.8 SYSTEM CODING 73

5.8.1 Coding Style 73

5.8.2 Documentation 74

5.9 SYSTEM STRUCTURE 75

Chapter 6: System Testing 76

6.1 INTRODUCTION 76

6.2 UNIT TESTING 76

6.3 INTEGRATION TESTING 77

6.4 FUNCTIONING TESTING 78

6.5 PERFORMANCE TESTING 79

6.5.1 Reliability Testing 79

6.5.2 Integrity Testing 79

6.5.3 Authentically Testing 79

6.5.4 Confidentially Testing 80

6.5.5 Efficiency Testing 80

Chapter 7: System Evaluation 81

7.1 PROBLEMS ENCOUNTERED AND SOLUTIONS 81

7.1.1 Inexperience in using development tools 81

7.1.2 Inexperience in setting up SSL on Tomcat 81

7.1.3 Difficulties in designing appearance for Palm 82

7.2 SYSTEM STRENGTHS 82

7.2.1 Provide Secure Internet Communication 82

7.2.2 Wireless Access 83

7.2.3 Mouse/Stylus Driven 83

7.2.4 Separated Databases 83

7.3 SYSTEM LIMITATIONS 84

7.3.1 No Real Environment Testing 84

7.3.2 Limited support on Web Browser 84

7.3.3 Partial Integration of Back End System 85

7.3.4 No Report Generating 85

7.3.5 No Encryption on password 85

7.4 FUTURE ENHANCEMENTS 85

7.4.1 Fully Integration on Back End 85

7.4.2 Increase Compatibility on other Web Browsers 86

7.4.3 Report Generating and System Logging 86

7.4.4 Password Encryption 86

7.4.5 Add more relevant Features 87

7.5 CONCLUSION 87

Appendix A – User Manual 89

1) SERVER MANUAL 89

a) Server Requirements 89

b) Server Software Installation Guide 89

i) Java Development Kit 1.3.1 Installation Guide 89
ii) Tomcat 3.2.3 Installation Guide 90
iii) MySQL Installation Guide 90
iv) MySQL JDBC Driver Installation Guide 91
v) JSSE Installation Guide 91
vi) JavaMail Installation Guide 92

2) FRONT END (CLIENT) MANUAL 92

a) **Client System Requirements** 92

b) **PQA Installation Guide** 92

c) **Getting Started** 92

d) **Password Forgotten** 94

e) **Making a Reservation** 95

f) **On Palm** 96

g) **Retrieve Reservation** 98

h) **Cancel a Reservation** 98

i) **Update Profile** 100

j) **Change Password** 100

3) BACK END USER MANUAL 100

a) **Back end User System Requirements** 100

b) **Update Profile** 100

c) **Change Password** 100

d) **Super User Manual** 101

i) Login 101
ii) Add Portal Admin 101
iii) View Portal Admin 102
iv) Delete Portal Admin 102

e) **Portal Admin Manual** 104

i) View Hotel 104
ii) Delete Hotel 104
iii) View Hotel Admin 106

iv) Add Hotel Admin 106
v) Delete Hotel Admin 108
vi) View Member 108

f) Hotel Admin Manual 111

i) View Officer 111
ii) Add Officer 111
iii) Delete Officer 112

g) Officer Manual 114

i) Update Hotel Profile 114
ii) Add New Room Type 114
iii) Edit Existing Room Information 116
iv) Delete Existing Room Type 116
v) Reservation Records Searching 117
vi) Managing Reservation 117
vii) Delete Overdue Records 119

References 121

List of Figures

Figure 2.1	Palm.Net network	11
Figure 2.2	Hotel reservation on Palm.	16
Figure 2.3	Marimari.com as an arbiter between the user and hotel.	17
Figure 2.4	Marimari.com hotel reservation process flow chart.	18
Figure 2.5	Hotelplatform.com hotel room reservation system.	19
Figure 3.1	System development model.	22
Figure 3.2	Travel Guide on Palm (TGoP) System and subsystems.	23
Figure 4.1	The TGoP – HRSS architecture.	34
Figure 4.2	Structure of HRSS.	35
Figure 4.3	Structure of client section.	36
Figure 4.4	Structure of portal administrator section.	36
Figure 4.5	Structure of hotel administrator section.	37
Figure 4.6	Structure of officer section.	37
Figure 4.7	Process flow chart of client section.	38
Figure 4.8	Process flow chart of portal administrator section.	39
Figure 4.9	Process flow chart of hotel administrator section.	40
Figure 4.10	Process flow chart of officer section.	41

Figure 4.11	Context Data Flow Diagram (DFD) for HRSS.	42
Figure 4.12	Level 0 DFD of client section.	43
Figure 4.13	Level 1 DFD of client section –Reservation process.	44
Figure 4.14	Level 1 DFD of client section – Reservation cancellation process.	45
Figure 4.15	Level 0 DFD of portal administrator section.	45
Figure 4.16	Level 0 DFD of hotel administrator section.	46
Figure 4.17	Level 0 DFD of officer section.	47
Figure 4.18	ER diagram of Client and Hotel with the other tables	54
Figure 5.1	MVC Design Pattern	60
Figure 6.1	Integration testing in Reservation Module	78
Figure A-1	TGoP main page	93
Figure A-2	Sign Up page	94
Figure A-3	Password Forgotten	95
Figure A-4	Find available rooms	96
Figure A-5	Result set for available rooms	97
Figure A-6	Reservation Form 1	97
Figure A-7	Reservation Form 2	98
Figure A-8	Reservation List	99
Figure A-9	Cancel a reservation	99

Figure A-10	Super User main page	101
Figure A-11	Add Portal Admin	102
Figure A-12	View Portal Admin	103
Figure A-13	Delete Portal Admin	103
Figure A-14	Searching for hotels	104
Figure A-15	Hotel List	105
Figure A-16	Hotel details	105
Figure A-17	Hotel Admin list	106
Figure A-18	Add Hotel Admin	107
Figure A-19	Delete Hotel Admin	107
Figure A-20	Find members	108
Figure A-21	Delete member	109
Figure A-22	Hotel Admin main page	110
Figure A-23	Officer list	111
Figure A-24	Add Officer	112
Figure A-25	Delete Officer	113
Figure A-26	Officer main page	114
Figure A-27	Update hotel information	115
Figure A-28	Room managing	115

Figure A-29 Edit room information 116

Figure A-30 Reservation records searching and results 117

Figure A-31 Reservation details – Approve reservation request 118

Figure A-32 Reservation details – Approve cancellation request 119

Figure A-33 Overdue record 120

Figure A-34 Delete reservation record 120

List of Tables

Table 1.1	Project timeline 1.	6
Table 1.2	Project timeline 2.	6
Table 2.1	Malaysia tourist arrivals 1999, 2000 and 2001.	7
Table 2.2	Comparison between marimari.com and hotelplatform.com.	20
Table 4.1	Client table.	48
Table 4.2	Month table.	49
Table 4.3	CreditCard table.	49
Table 4.4	Prof table.	49
Table 4.5	Reservation table.	50
Table 4.6	City table.	51
Table 4.7	State table.	51
Table 4.8	Country table.	51
Table 4.9	Grade table.	52
Table 4.10	Location table.	52
Table 4.11	Hotel table.	53
Table 4.12	Room table	54
Table 4.13	Administrator table.	55

Table 4.14 User interface for different functions and activities. 56

Table 4.15 User interface for different functions and activities (continue). 57

Chapter 1: Introduction

A travel guide is a very important reference for people who wish to plan their trip to a holiday destination. From the travel guides, we can get information like accommodation, transportation, description about some attractive places even the culture of the community of the place.

Conventionally, the travel guides are present in many forms such as handbooks, magazines, or articles in newspaper. After the Internet has connected the world, we can find these references appear in the Internet. To fully utilize the Internet, some of countries, like Malaysia, publish the travel guides on the Internet to promote this smokeless industry for the country. Therefore, the Internet users all around the world are able to get this information from Internet.

However, this information is not easily accessible except printed out or accessed using a laptop that will be a burden during the trip. This problem can be solved if the information can be gained from a handheld device such as a Personal Digital Assistant (PDA). Thus, a handheld device with wireless network connected will allow users to plan their tour easily with a few taps on the handheld device.

1.1 Project Introduction

The Travel Guide on Palm (TGoP) is a system which is specially dedicated to the Palm OS handheld device (Palm) users as a guide and assistant in their tour in Kuala Lumpur. In the system, the basic tourism information about places around the Klang Valley area will be provided, such as map, accommodation, transportation, currency exchange, and weather forecasting information.

This system is complemented with the online Hotel Room Reservation System (HRRS). The room rates and simple descriptions of the hotels will be displayed on the Palm. Users may choose a hotel based on a set of criteria they like and do a reservation via theirs Palm. The hotel will receive reservations from the users and then sends a confirmation to them. The hotels may also update their information such as the latest room rates for the hotel.

1.2 Project Objectives

The objectives of this project are:

- To provide a useful TGoP application to the Palm users.
- To provide a convenient online hotel room reservation service for the Palm users.
- To encourage the tourism industry in Malaysia.

1.3 Project Scope

Considering that the scope of this project is indeed wide, the implementation of a complete system would require a significant amount of time and manpower. Therefore, in the context of this project, the system is restricted to:

- Provide an extensive information search for information in the database;
- Provide a categorized and easy to find information system;
- Develop an efficient and reliable database system;
- Provide a user friendly room booking process;
- Provide an intelligent reservation tracking system;
- Provide an efficient and reliable database updating and maintaining interface for both the administrator and user.

1.4 Project Limitations

In the TGoP-HRRS, some unavoidable constraints will limit the system development. Those are:

- Device limitation

  The Palm as a handheld device has a smaller screen size and resolution if compare to desktop. Therefore, displaying a right information is more important than fitting as much information on the screen as possible. The system must balance between showing enough information and keeping the interface uncluttered and

simple to use. In addition, limited memory space of the Palm also creates a challenge for us during the application writing.

- Expensive wireless bandwidth

  Due to the scarce and expensive wireless bandwidth (the wireless network even not yet been introduced in Malaysia, but we hope it will be done soon), we have to minimize the amount of data that must be transferred.

1.5 Project Expected Outcome

Due to the device limitation as mention before, the system's user interface will be a combination of Palm OS application and web clipping application. Most of the information will be a database on Palm, which in form of Palm Os and only some information and the reservation will use the web clipping application. From the point of view of the users, the expected outcomes are:

- Information viewing

  This feature allows users to compare the hotel room rates and simple description about the hotel such as location and ranking.

- Resource updating

  User may update their database in the Palm during the synchronization between the Palm and desktop.

- Reservation

  The users may make a room reservation base on their criteria. They just need to simply fill up a provided form on the Palm, and then send the request to server.

The backend is controlled by a portal administrator, hotel administrator and the officer. The outcome may allow the portal administrator to:

- Add/delete hotel administrator;
- Cancel a client membership.

The hotel administrator will be able to:

- Add/delete officer;
- Modify/update the authority of the officer;
- View records for the information of the officer and the room reservation records.

For the officer, the expected outcomes are allow them to:

- Add/cancel the hotel room reservation;
- Modify/update the hotel room reservations;
- Add/delete the hotel rooms information;
- Modify/update the hotel rooms information;
- View the records of the room reservations.

1.6 Project Schedule

This project will be carried out in two stages. The activities involve in the first stage are concept definition, research and literature review, requirements and specifications analysis and the system design. This stage is to plan and schedule the project development to ensure that it is carried out according to the required standard. The following activities in the second stage are incremental prototyping and integration and testing. The schedule for the activities of the project is as follows:

ID	Task Name	Start	Finish	Duration
1	First Stage	11-Jun-01	02-Sep-01	84d
2	Concept definition	11-Jun-01	17-Jun-01	7d
3	Research and literature review	18-Jun-01	08-Jul-01	21d
4	Requirements and specifications analysis	09-Jul-01	29-Jul-01	21d
5	System desgin	30-Jul-01	02-Sep-01	35d

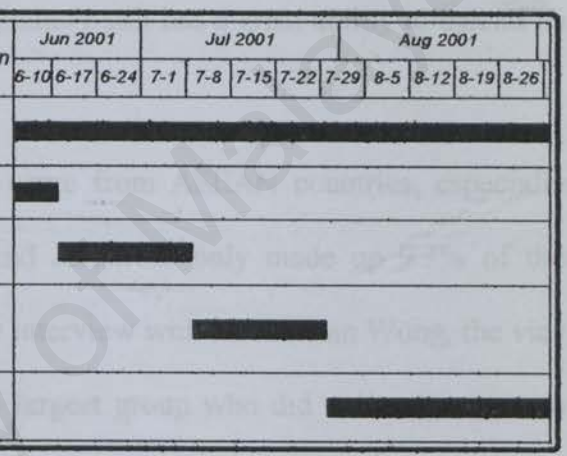

Table 1.1 Project timeline 1.

ID	Task Name	Start	Finish	Duration
1	Second Stage	03-Sep-01	27-Jan-02	147d
2	Incremental and prototyping	03-Sep-01	30-Dec-01	119d
3	Integration and testing	31-Dec-01	27-Jan-02	28d

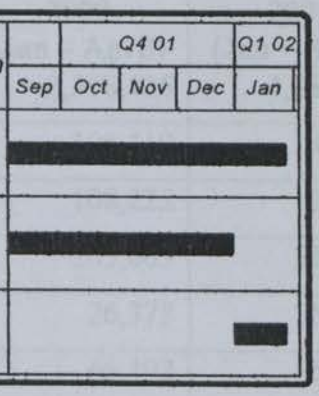

Table 1.2 Project timeline 2.

Chapter 2: Literature review

2.1 Tourism industry in Malaysia

Tourism industry is one of the main incomes of our country. According to the statistic, in the year 1999, there are 7,931,149 tourists who visited our country. In 2000, the number has increased to 10,221,582, which is an increase of 28.9% compare to 1999. In the first season (January to April) of year 2001, total visitors are 4,894,084. If compared to the same term in 2000, we find that there is a great increase of 53.4%. This study strongly convinces us that the tourism industry in our country still has a great ability to extend its strength.

From the statistic, the majority tourists are came from ASEAN countries, especially Singapore. Although visitors from Europe and Americas only made up 9.3% of the tourists in the first season of year 2001, in the interview with Mr. Adrian Wong, the vice president of marimari.com said, they are the largest group who did online hotel room reservation using marimari.com. The hotel reservation has profit this dot com company RM100,000 per month.

Region	1999 (Jan – Dec)	2000 (Jan – Dec)	2000 (Jan – April)	2001 (Jan – April)
Asia	6,763,480	8,682,189	2,598,435	4,085,485
Americas	122,079	307,692	106,419	143,018
Oceania	154,078	267,626	108,222	125,475
Europe	306,060	581,599	265,663	313,365
Africa	29,863	74,314	26,378	68,777
Others	555,589	308,162	66,492	157,964
Grand total	7,931,149	10,221,582	3,171,609	4,894,084

Table 2.1 Malaysia tourist arrivals 1999, 2000 and 2001.
Source: Immigration Department of Malaysia (KL)

2.2 The Palm Computing Platform

2.2.1 Palm OS Philosophy

The Palm OS handheld is designed as a satellite device, which is an extension of the desktop system. The handheld provides a window to desktop data, allowing that data to be viewed anywhere. Though it is indeed possible to perform many complex tasks with Palm OS handheld, their form and functions are optimized for viewing data and entering small amounts of data.

In order to meet the above goals, the handheld device must adhere to certain criteria:

- Small size

  It needs to be small enough to be carried anywhere. Most of the devices currently available for the Palm Computing platform easily fit in a shirt pocket.

- Ergonomic interface

  Using the device must be simple and quick enough to not interrupt whatever the user is doing such as during meeting, in airports and in others situation that there is no convenient place to set the device down. Thus, the useful information should be available instantly and with a minimum of user interaction.

- Desktop integration

  The handheld must synchronize easily and reliably with the desktop computer. Synchronizing with the desktop not only backs up important data, but it also allows the user to input large amounts of the data on a desktop machine with a mouse and keyboard. Palm Computing's HotSync technology provides a quick

data transfer between the handheld and the desktop.

2.2.2 Connecting to the Desktop

The database synchronization between Palm OS handheld and desktop is an important feature. The cradle makes a connection between desktop and the handheld allows each device to borrow the strengths of the other. The software component forms the vital link between the Palm OS device and the desktop computer is called a conduit. The conduit resides on the desktop and is called by the HotSync during the data synchronization.

2.2.3 Communicating with other devices

The ability of Palm Computing platform to communicate with other devices is a key part of its success. There are a number of different communications protocols offered by current versions of the Palm OS.

- Serial

  Palm OS devices use the serial protocol to synchronize through a cradle with a desktop computer. With the right cable or third-party software, the Palm OS can also talk to just about anything, from the modems to temperature probes to GPS receivers.

- TCP/IP

  The Internet standard protocol, TCP/IP allows a Palm OS device to connect to any

machine on the Internet and exchange data with it.

- Wireless

  Palm OS version 3.2 introduced wireless communication on a Palm OS device via the Palm.net wireless network. The Palm OS wireless system uses a strategy called web clipping to minimize the amount of data that must be transferred due to the expensive wireless bandwidth.

- IrDA

  Starting with Palm OS version 3.0, Palm devices can communicate via the industry-standard Infrared Data Association (IrDA) protocol via infrared (IR) with a similarly equipped devices such as cell phones, pagers and even desktop or laptop computers.

- Beaming

  The Palm OS exchange manager provides facilities for beaming individual records and applications between two devices via infrared (IR).

2.3 Palm.Net System

The Palm.Net system is designed for Palm devices to access to real-time data but not casual browsing. It is possible for browsing, but the increased cost and volume of data involved with visiting most standard web sites makes it impractical over a wireless network.

Typical scenarios involve users accessing the following kinds of information on the Internet: news, sports scores, weather, traffic reports, driving directions, airline schedules and flight information, stock quotes, hotel and restaurant information and email.

2.3.1 Palm.Net network

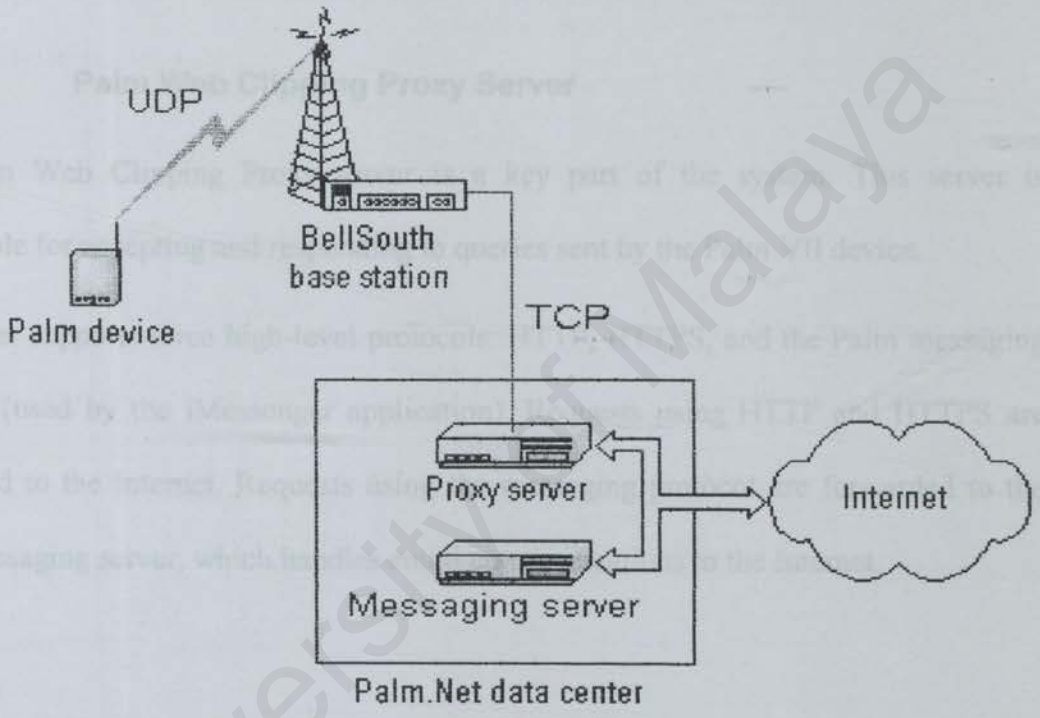

Figure 2.1 Palm.Net network

Figure 2.1 illustrates the physical Palm.Net network. The Palm VII device communicates via radio modem to a nearby BellSouth Wireless Data network base station. From there, data is sent over a private link to the Palm Web Clipping Proxy server in the Palm data center. The proxy server interprets users' requests and passes them to other computers on the Internet, using standard HTTP protocols, to handle as appropriate.

Responses are returned to the proxy server, which communicates them to the Bell South wireless network and back to the Palm VII device via radio modem.

The wireless radio link operates at approximately 8 kbps, so is best suited for exchanging small amounts of information. After accounting for headers, error correction, and other overhead, the effective data throughput is roughly 2 kbps, so compactness is critical.

2.3.1.1 Palm Web Clipping Proxy Server

The Palm Web Clipping Proxy server is a key part of the system. This server is responsible for accepting and responding to queries sent by the Palm VII device.

The server supports three high-level protocols: HTTP, HTTPS, and the Palm messaging protocol (used by the iMessenger application). Requests using HTTP and HTTPS are forwarded to the Internet. Requests using the messaging protocol are forwarded to the Palm messaging server, which handles email communications to the Internet.

2.3.1.2 UDP

One way that Palm optimizes the limited network bandwidth is to use UDP (User Datagram Protocol). All communications between the Palm VII device and the wireless network use UDP. This transmission protocol is extremely efficient and lightweight, resulting in the exchange of the fewest packets possible over the wireless network. Often requests and responses require just a single data packet. This is much more efficient than the relatively verbose TCP. Using UDP decreases users' airtime costs because fewer

packets are required for each request and response.

UDP does not normally function as a reliable protocol, however, the wireless connection between the Palm device and the BellSouth Wireless Data network has guaranteed delivery and reliability built into it via other mechanisms, so there is no need for the extra overhead of a full connection-oriented protocol such as TCP.

WWW requests that are passed to the Internet by the proxy server use TCP to guarantee reliability over the Internet.

2.3.1.3 Compressed HTML

Another way that Palm efficiently uses the limited bandwidth of the Palm.Net system is to compress HTML.

Web clippings are rendered on the Palm VII device by the Clipper application. Clipper renders compressed HTML data. Both the query applications and WWW data returned from the Internet are compressed.

- When creating Palm query applications, the Query Application Builder program compresses HTML content and combines multiple HTML pages and images into a single query application.
- All HTML information returned to the Palm device from the Internet is dynamically compressed by the Palm Web Clipping Proxy server before transmission through the wireless network to the Palm device.

It is important to note that the Palm device accesses standard HTML data that resides on

standard HTML web servers on the Internet. The compression by the proxy server is transparent to the users and the web server on the Internet.

If a web page that is not Palm-friendly is browsed, the proxy server removes images, scripting code, Java code, frames, and other non-supported elements before sending the content to the Palm device. Additionally, the content is truncated to prevent large amounts of unexpected data from being transmitted. The user can request more data as desired.

2.3.1.4 Security

All wired parts of the network support security via the SSL (Secure Sockets Layer) protocol widely used by servers and browsers on the Internet. However, SSL is impractical to run over a low bandwidth wireless network because it is quite verbose.

Palm implements a level of security for the wireless portion of the network that is equivalent to the 128-bit SSL encryption algorithms, but is optimized for use on a wireless networks. The wireless part of the network is protected by a security system that includes encryption, message integrity checking, and server authentication.

Message encryption is done via an elliptic curve cryptography engine supplied by Certicom Corporation. Message integrity checking protects against transmission errors or message manipulation. Server authentication prevents the wireless session between the Palm device and the proxy server from being hijacked or spoofed.

2.3.2 Palm Query Application

The primary mechanism that Palm has provided for users to interact with the WWW (World Wide Web) is the Palm query application (PQA). Palm query applications encapsulate locally stored HTML content, possibly including one or more query forms, through which the user can submit requests for information from the WWW. Returned data, called web clippings, are displayed by the web clipping viewer application (called Clipper here) that runs on the Palm device.

Palm query applications are created by the Query Application Builder program that runs on a desktop computer. This program translates one or more pages of HTML content into a single compact database (.pqa file) that the user installs on the Palm device.

When creating the .pqa file, the Query Application Builder translates HTML into a compressed format. The Clipper application works with this compressed format, rather than HTML directly. The reason for this is that HTML is an inefficient format for the transmission of data over the network and storage of information. Compression minimizes the amount of information sent over the radio and reduces the size of query applications stored on the Palm device.

GIF and JPEG images incorporated into source HTML files are converted to the Palm bitmap format (2-bit graphics) before being stored in the query application file.

2.4 Existing hotel room reservation system review

There are many web sites that provide online hotel room reservation. Most of them are similar. In general, we may categorize these web sites in to two major groups. The first group is the web site itself acting as an arbitrator between the user and hotel. Another approach is the web sites only provide the hyper link for the user to do the reservation directly with the hotel.

2.4.1 Hotel room reservation system on Palm

The hotel room reservation system on Palm is not much difference from the system on web. The major different between these two is the system on Palm provides static information for the users using a PQA that they need to download from the Internet. Due to the high cost of the wireless bandwidth, the users only connect to the Internet when they submit their form to the Internet. The remaining process will depends on the approach that the web site uses which similar to the system on desktop. This is discussed further in the following sections.

One of the examples is Hotel Reservation Network (www.180096hotel.com or www.hoteldiscount.com).

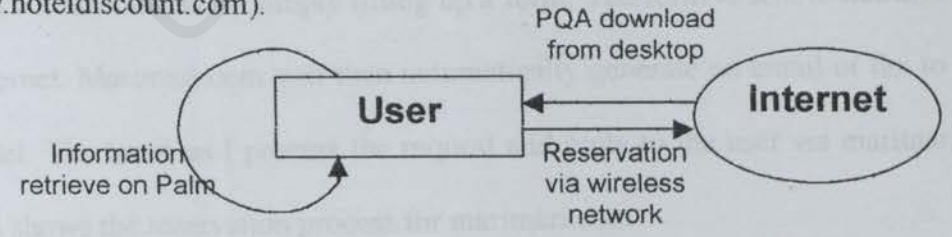

Figure 2.2 Hotel reservation on Palm.

2.4.2 Hotel room reservation system on desktop

2.4.2.1 Marimari.com

Marimari.com is one of the web sites that provides tourism information in Malaysia. Besides, marimari.com also provide hotel information and room reservation for hotels in Malaysia. Marimari.com acts as an arbiter between the user and hotel.

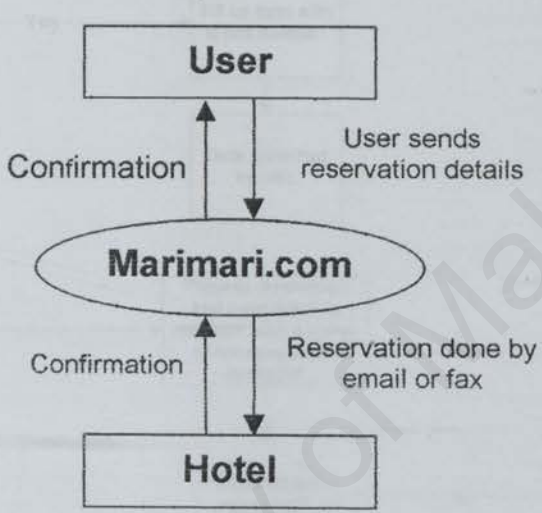

Figure 2.3 Marimari.com as an arbiter between the user and hotel.

At marimari.com, we can obtain hotel information such as ranking, room rates, facilities, location and a simple description about the hotel. Based on this information, users can make a reservation by simply filling up a form. The form is sent to marimari.com via the Internet. Marimari.com will then automatically generate an email or fax to the specified hotel. The hotel will process the request and reply to the user via marimari.com. Figure 2.4 shows the reservation process for marimari.com.

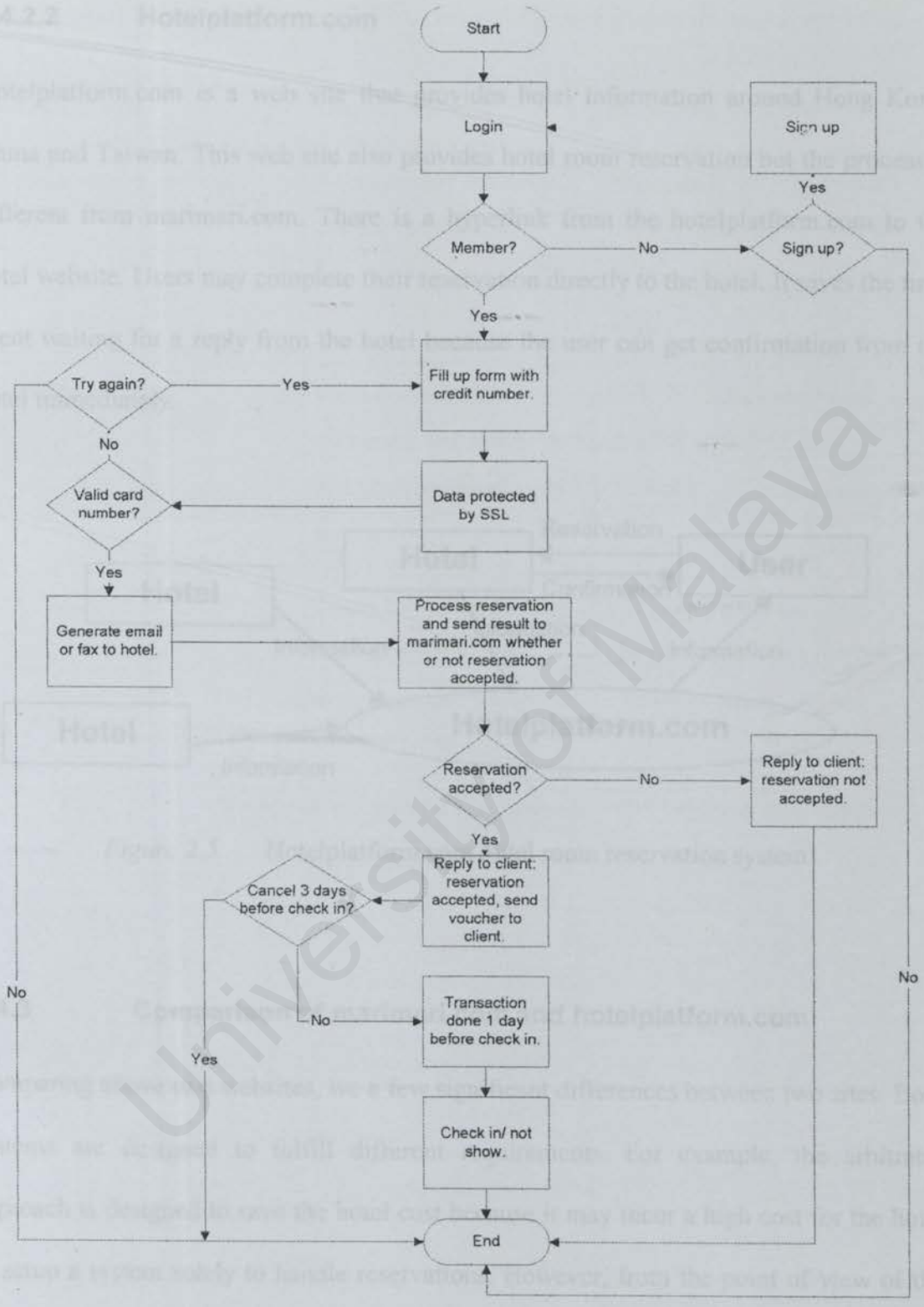

Figure 2.4 Marimari.com hotel reservation process flow chart.

2.4.2.2 Hotelplatform.com

Hotelplatform.com is a web site that provides hotel information around Hong Kong, China and Taiwan. This web site also provides hotel room reservation but the process is different from marimari.com. There is a hyperlink from the hotelplatform.com to the hotel website. Users may complete their reservation directly to the hotel. It saves the time spent waiting for a reply from the hotel because the user can get confirmation from the hotel immediately.

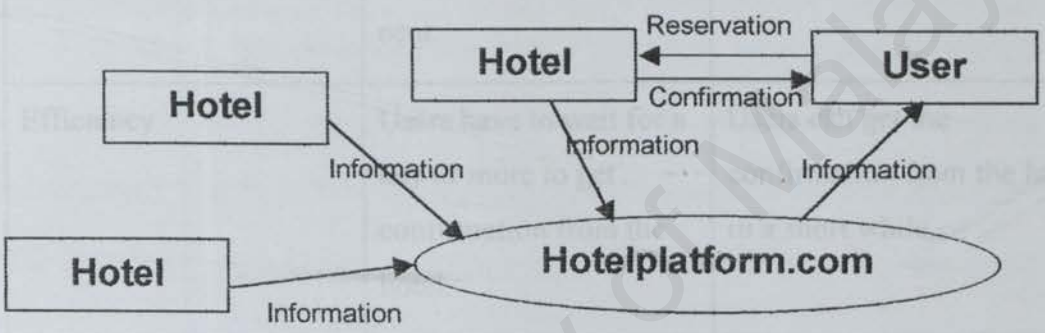

Figure 2.5 Hotelplatform.com hotel room reservation system.

2.4.3 Comparison of marimari.com and hotelplatform.com

Comparing above two websites, we a few significant differences between two sites. Both systems are designed to fulfill different requirements. For example, the arbitrator approach is designed to save the hotel cost because it may incur a high cost for the hotel to setup a system solely to handle reservations. However, from the point of view of the dot com company, it is much easier for them if the reservation is done by users directly

with the hotel. The following table shows the difference between the two approaches.

Differences	Marimari.com	Hotelplatform.com
Reservation process	As an arbiter between the user and hotel.	Provides hotel information but the reservation is done by user directly to the hotel.
System cost	High costs for the company, but hotels do not need to spend even a cent.	Both company and hotel need to invest on the system.
Efficiency	Users have to wait for a day or more to get confirmation from the hotel.	Users can get the confirmation from the hotel in a short while.

Table 2.2 Comparison between marimari.com and hotelplatform.com.

Chapter 3: System Analysis and Requirements Specification

3.1 System Development Model

A synthesis of the Waterfall, Prototyping and Incremental development model has been identified as a methodology to be used for the Travel Guide on Palm - Hotel Room Reservation System (TGoP-HRRS) development.

The overall development is broken into distinct phases much like the waterfall model, with the exception that prototypes are used during the development of the HRRS modules. The incremental aspect of the model comes in when additional functions are added iteratively to each of the module prototypes. Functions are added incrementally so that the framework is quickly developed and later on added with more features.

The initial stages of the development model are conducted on a general basis covering most of the HRRS as a whole. Upon reaching the prototyping stage, individual modules are developed separately and prototyping is also conducted separately from each other. During this time, the future integration needs of the separate modules also have to be catered for and are handled during this time for future needs.

Once each of the individual modules has been sufficiently developed to a reasonable stage of completeness, they are brought together for the final integration into the complete HRRS, unifying the separate parts and forming a coherent system. Final testing is then conducted on the HRRS as a whole. Following the conclusion of this project, the HRRS may be further added with more modules to increase its functionality.

The advantages in using this development model are:

- Additional future functionalities can be added effortlessly into the system;
- It makes explicit which intermediate products are necessary in order to begin the next stage;
- Allow all or part of the system to be constructed quickly to verify the requirements so that problems can be solved earlier.

Besides that, there are many analysis methods that are carried out for the development of this project:

- Brainstorming
- Internet
- Try out the real Palm OS applications
- Reference books
- Group discussion

The system development model as described is depicted in Figure 3.1:

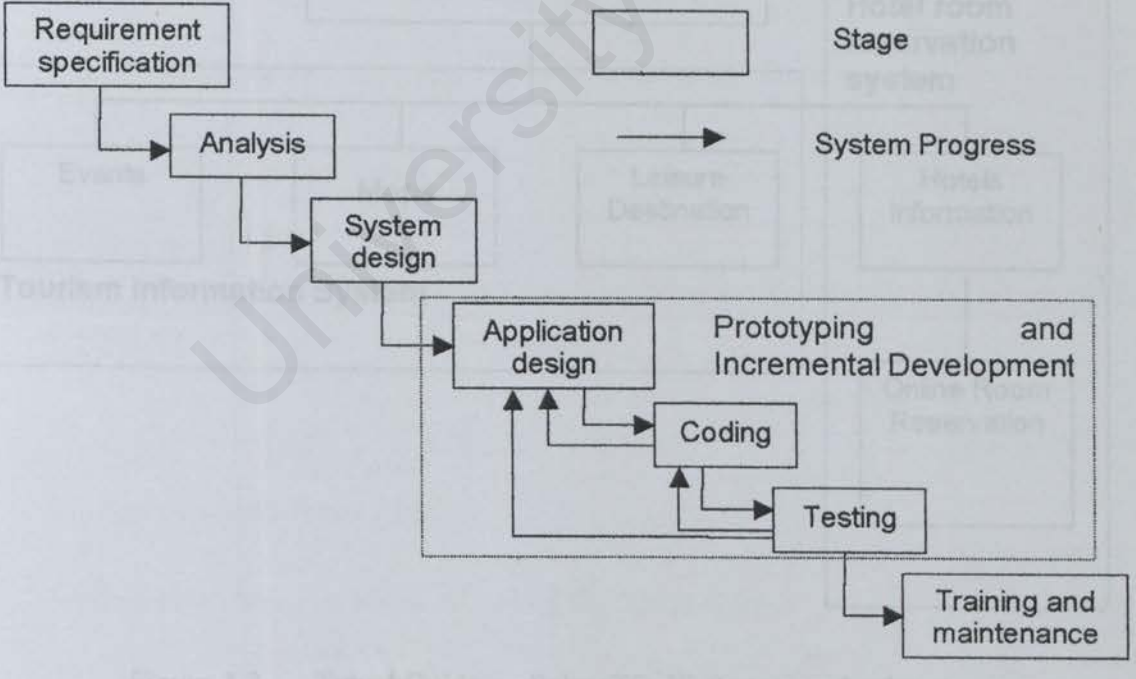

Figure 3.1 System development model.

3.2 Project Definition

As mentioned before, the TGoP is a system that allows users to view tourism information and make a hotel room reservation via the wireless or wired network. Thus, the TGoP is split in to two subsystems in the development phase. One of the subsystems is tourism information system which provides information such as events, maps, and leisure destination review. This subsystem is documented by Chung Yun Liang. The other subsystem is hotel room reservation system which provides online reservation for user and complemented with hotels information such as hotel room rates, ranking and location.

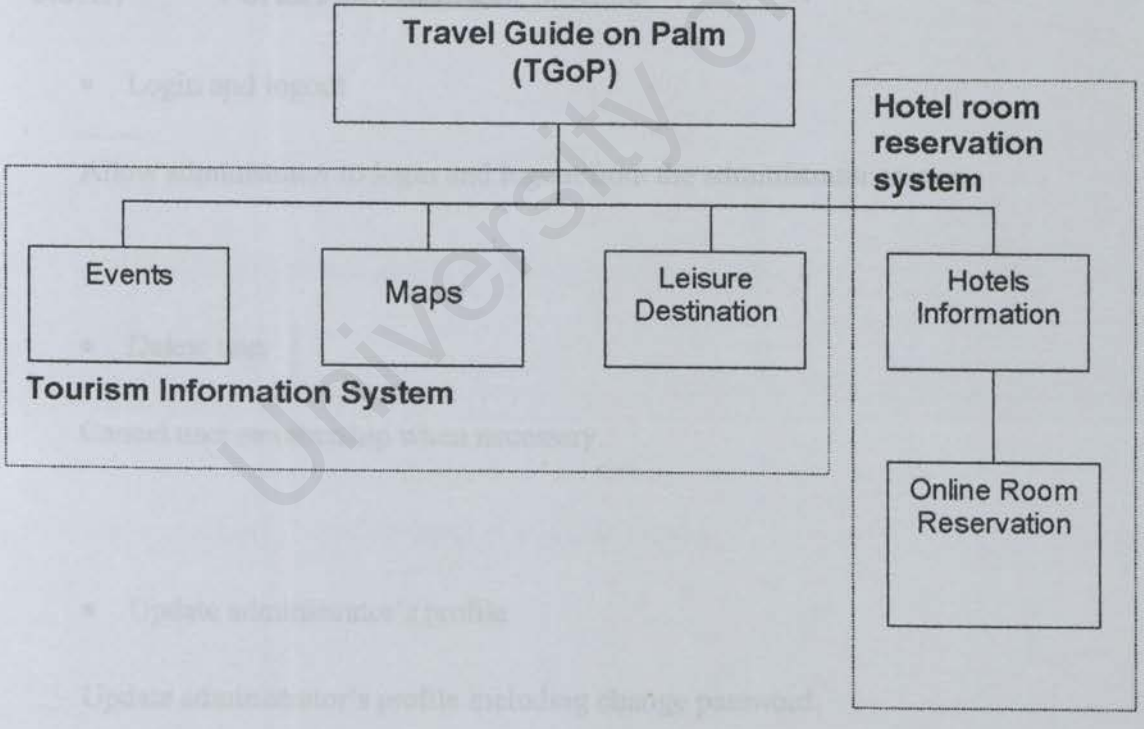

Figure 3.2 Travel Guide on Palm (TGoP) System and subsystems.

3.3 System Requirements

The requirements of the TGoP – HRRS can be divided into functional requirements and non-functional requirements. Functional requirements describe the activities and services of the system that must be provided. Non-functional requirements are limitations on the system that constraint the boundary of the system.

3.3.1 Functional Requirements

Functional requirements of the TGoP – HRRS basically are analyzed based on the portal administrators, hotel administrators, officers and Palm users (clients).

3.3.1.1 Portal Administrator Module

- Login and logout

Allow administrator to login and logout from the administrator section.

- Delete user

Cancel user membership when necessary.

- Update administrator's profile

Update administrator's profile including change password.

3.3.1.2 Hotel Administrator Module

- Login and logout

Allow the administrator to login and logout from the administrator section.

- Add officer

The administrator may add officers with certain level of authority given to the officers.

- Delete officer

The specific officer may be deleted if necessary.

- Update administrator's profile

Update administrator's profile including change password.

- Change authority

Change officer's authority level to meet the needs.

3.3.1.3 Officer Module

- Login and logout

Allow officer to login and logout from the officer section.

- Approve reservation

Approve reservation when a room available for booking and customer credit card number is valid.

- View and update reservation status

The officer may keep track on all users' reservation status and cancel a reservation when necessary.

- Update room rates information

The room rates can be adjusted to suite the market needs.

- Update officer's profile

Update officer's profile including change password.

3.3.1.4 Client Module

- Login and logout

Allow client to login and logout from the client section.

- Reservation

Make reservation via the wireless network or wired network. The reservation process supports transaction using credit card number.

- Cancel reservation

Cancel reservation before check in.

- View reservation status

Keep track of user's reservation status.

- Update user's profile

Update user's profile including change password.

- Password tracking

Allow users to trace their password if they forgot their password.

3.3.2 Non-functional Requirements

The following are the non-functional requirements for TGoP - HRRS:

- Reliability

The system must provide a consistent and accurate output for every process. System failures must to be minimized. The system has to be stable and consistent in any environment.

- Integrity

The system only allows authorized users to be accessed. The users need to login before access the system.

- Authentically

Only authorized users may modify the system and information. Users are only authorized to modify their own profile.

- Confidentially

The important data transfer over the network, such as credit card number must be encrypted.

- Efficiency

The system must ensure efficiencies in system execution and data storage. The simplicity of the interface allows users to become familiar the system in a short time. The system should provide concise information due to the limited storage of Palm and expensive wireless bandwidth.

- Scalability

Due to the rapidly increasing number of Palm users, system users might be increased rapidly as well. Therefore, the system should be allowed to expand without affect the normal operation of the system.

- Flexibility

The system changes should be easily implemented. Changes may often take the form of upgrades or enhancements.

- Documentation

The system must be fully documented to provide guidance to all existing or potential users.

3.4 Development Tools Specification

3.4.1 Server Hardware and Software specifications

3.4.1.1 Web Server technology

- Jakarta-Tomcat

Jakarta-Tomcat is a web server which supports Java Servlet and Java Server Pages (JSP). It can be a standalone web server or integrated into the other servers such as Apache Web Server.

3.4.1.2 Database management system technology

- MySQL

MySQL is a relational database management system. The data are stored in separate tables and linked by defined relations to combine data from several tables on request. This makes MySQL fast, reliable and easy to use even to handle large amount of data.

3.4.1.3 Data access technology

- Java Database Connectivity (JDBC)

The JDBC package is a set of Java classes that can be used by applications to make database calls. It is designed for use by programmers to perform low-level access to a database.

3.4.2 Client Hardware and Software specifications

The client must be a Personal Digital Assistant (PDA) user with:

- Palm OS 3.2 or above as computing platform;
- Clipper, a browser application resident on the Palm OS device;
- Palm VII or modem adapted for wireless connection;
- PC with Microsoft Windows 95 or above

3.4.3 Administrators and Officer Hardware and Software specifications

The administrators and officer's computers requires:

- Any compatible PC with at least 64 MB RAM recommended;
- Web browsers, Internet Explorer 5.0 or above or Netscape Navigator 4.75 or above.

3.4.4 Programming language

- Java Server Pages (JSP)

JSP provides the functionality for building dynamic and interactive web pages. JSP can run on any web server and it provides a flexibility to choose a suitable platform.

- Java Servlet

A Java Servlet enhances the functionality of a web server. In a client-server relationship, the client requests that some action be performed and the server performs the action and responds to the client. A Servlet in the server will dynamically generates custom HTML documents (responses) to be displayed by browsers and that maintain unique session information for each client.

- Hypertext Markup Language (HTML)

HTML is used to display the most of the web page. Some of the HTML files is compressed to generate Web Clipping application.

- Extensible Markup Language (XML)

XML is a tool for defining languages, where a language is a set of tags and attributes with various constraints on them. For instance, XML has been to define a successor to HTML called XHTML.

- C and C++

C and C++ are the most common languages used to develop applications for Palm Computing platform.

- Java 2 Micro Edition (J2ME)

J2ME is designed specially to run on small computing devices like cellular telephones or personal digital assistants.

3.4.5 Other development tools

- Palm OS Emulator (POSE)

POSE emulates a Palm OS handheld at the hardware level, right down to the processor. POSE can do almost anything that an actual handheld is capable of doing.

- CodeWarrior® Lite for Palm OS®

CodeWarrior Lite is a smaller version of the commercial CodeWarrior product, yet it contains the same award-winning IDE and high-quality tools that the commercial version has. CodeWarrior Lite allow us to create, compile, and debug applications that will run on a Palm connected organizer or the Palm OS Emulator.

- Web Clipping Application (WCA) Builder

Web Clipping Application Builder tool is a Query Application Builder program that compresses HTML content and combines multiple HTML pages and images into a single query application called palm quey application (pqa). The pqas will install in the Palm for users to interact with the Internet.

Chapter 4: System Design

4.1 System Architecture

The TGoP – HRRS is using distributed three-tier Web application architecture. Each separated layer is allowed to develop separately from the others to any degree of sophistication. The system architecture is shown in Figure 4.1.

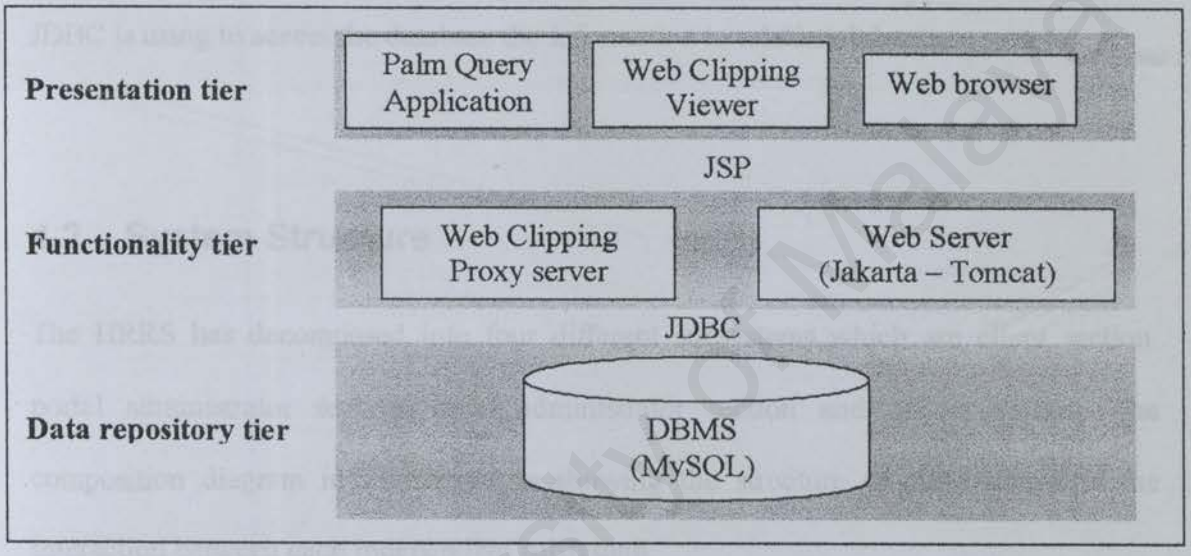

Figure 4.1 The TGoP – HRRS architecture.

The presentation tier responsible is to provide View, Controller and Model to the system user. It consists of palm query application (PQA), web clipping viewer and web browser. The palm query application is combination of compressed HTML content with multiple HTML pages and images. The palm user will use the PQA to send query to the web clipping proxy server. All HTML information will display on the Palm device with the web clipping viewer. However, the users need to download the PQA from the Internet using web browser on desktop. For the administrators and officer, they may update the

database using the web browser via Internet.

The middle tier, functionality tier process requests from users and send back responses to the users. The web server is responsible to process requests from web browser and web clipping proxy server as well in the form of HTML or generate from the JSP. The web clipping proxy server will response to the Palm user query.

For the data repository tier, the MySQL server is responsible to manage the database. JDBC is using to access the database the information in relational database system.

4.2 System Structure

The HRRS has decomposed into four different subsystems which are client section, portal administrator section, hotel administrator section and officer section. The composition diagram in Figure 4.2 is showing the structure of the system and the interaction between each independent subsystem.

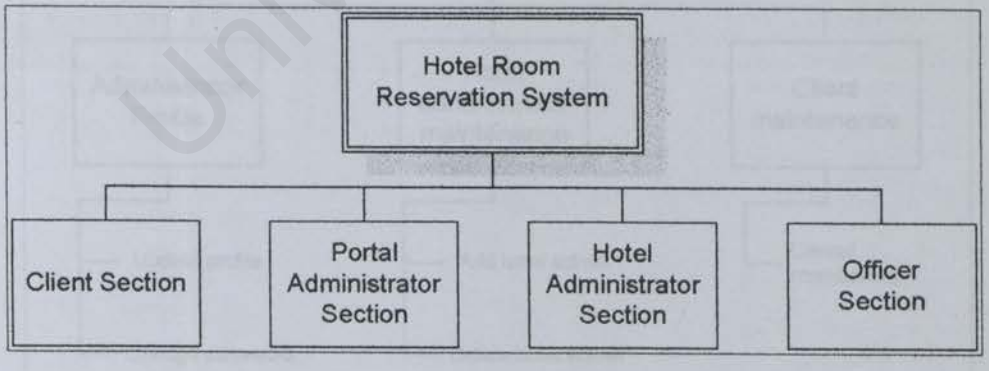

Figure 4.2 Structure of HRRS.

Figure 4.3 to 4.6 show the details for the activities and tasks of each section.

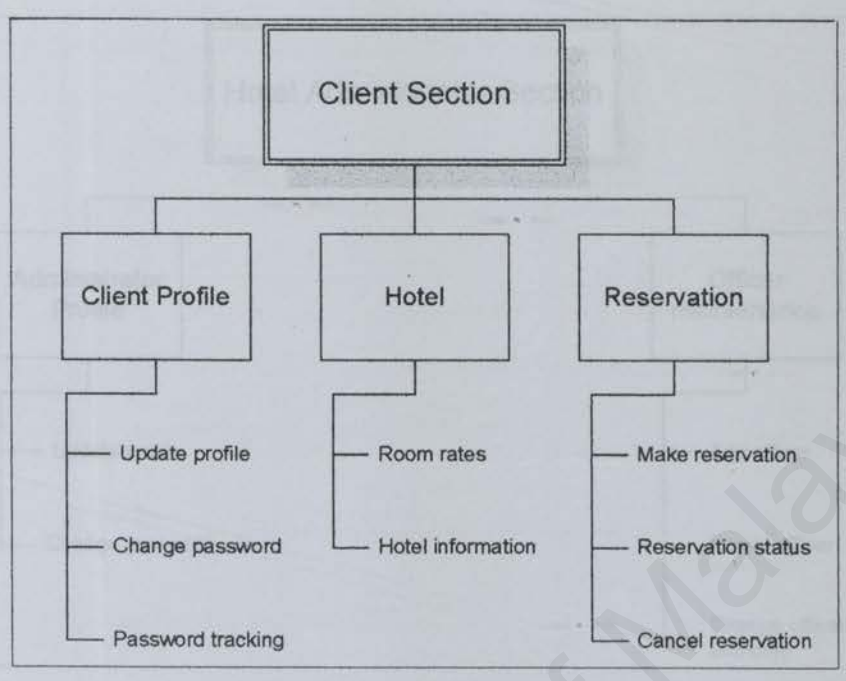

Figure 4.3 Structure of client section.

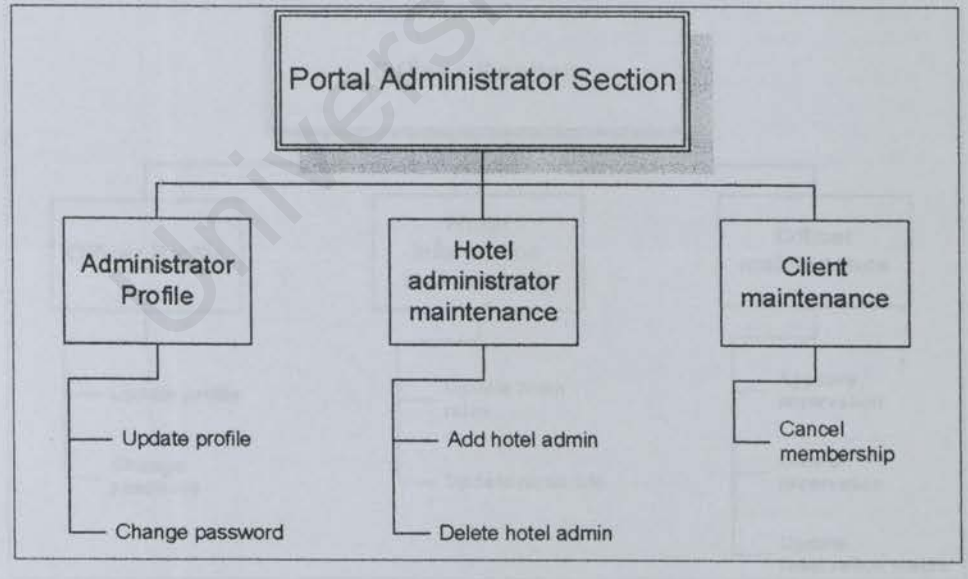

Figure 4.4 Structure of portal administrator section.

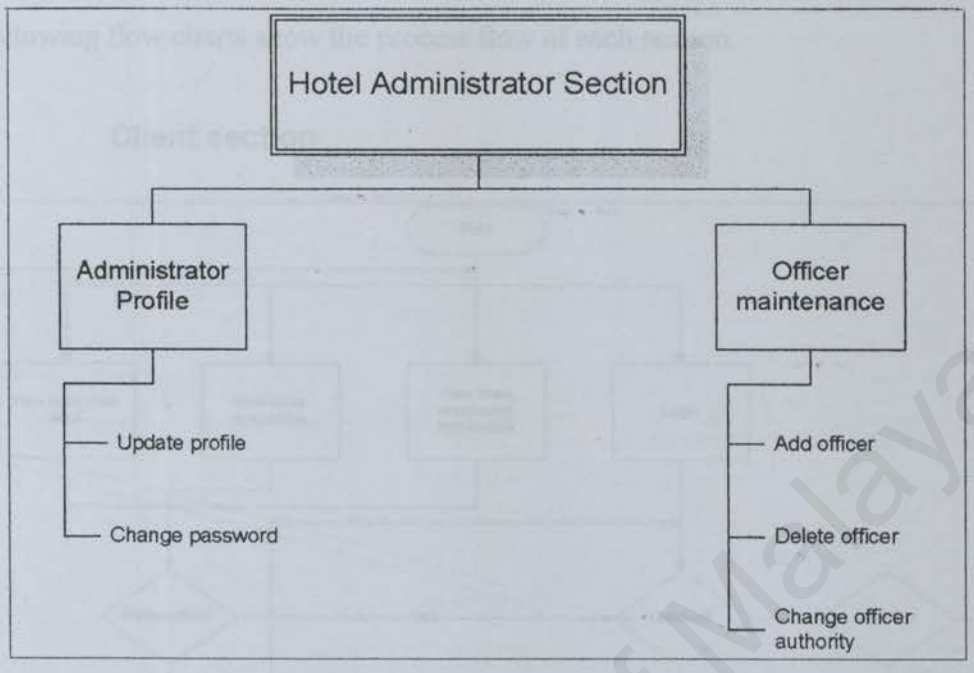

Figure 4.5 Structure of hotel administrator section.

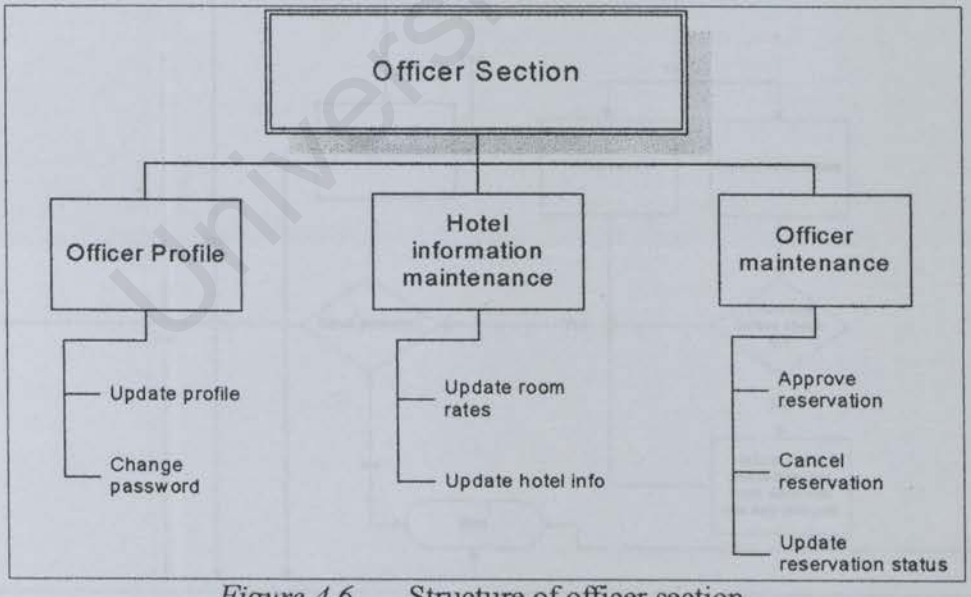

Figure 4.6 Structure of officer section.

4.3 System Flow Chart

The following flow charts show the process flow of each section.

4.3.1 Client section

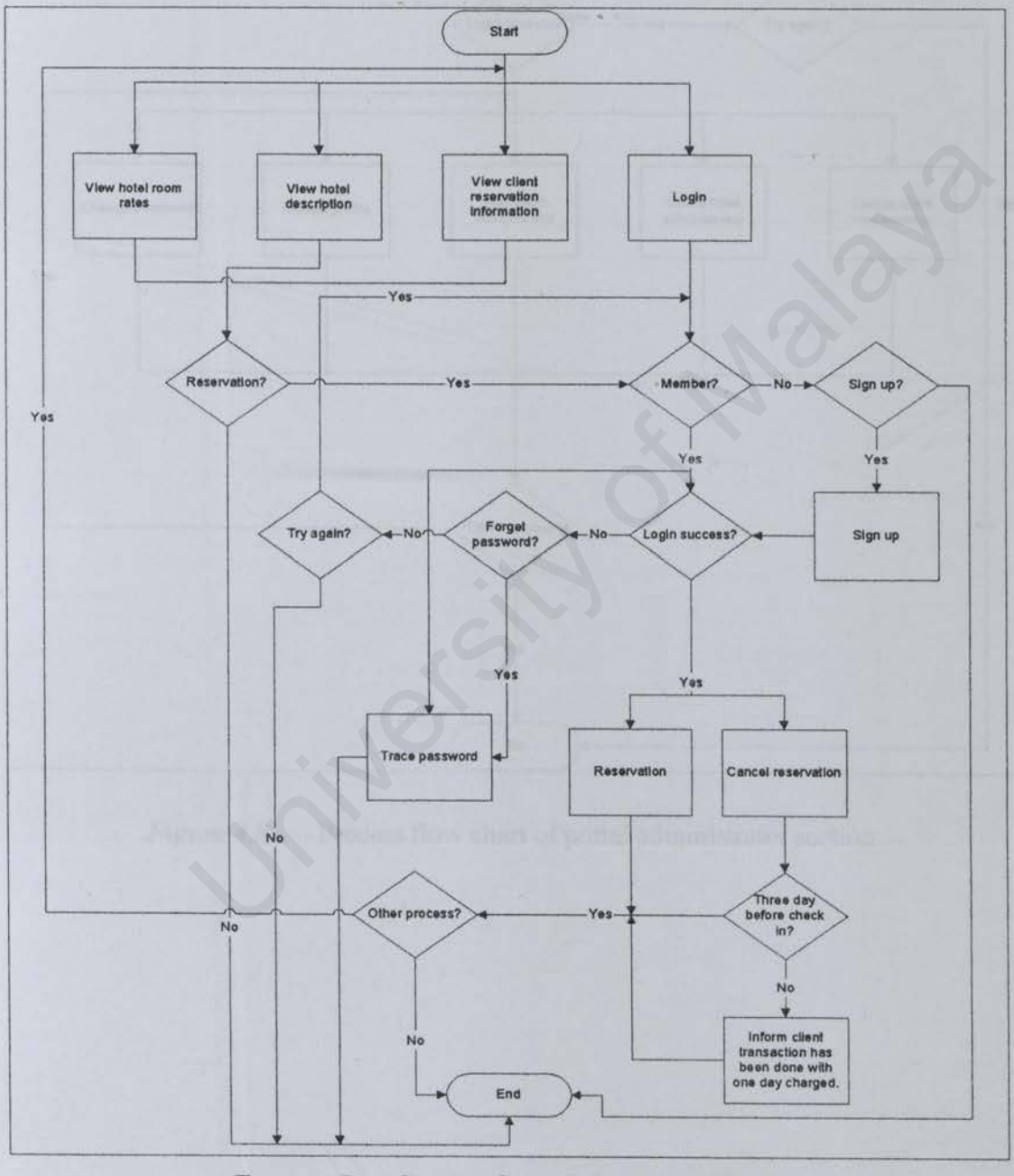

Figure 4.7 Process flow chart of client section.

4.3.2 Portal Administrator section

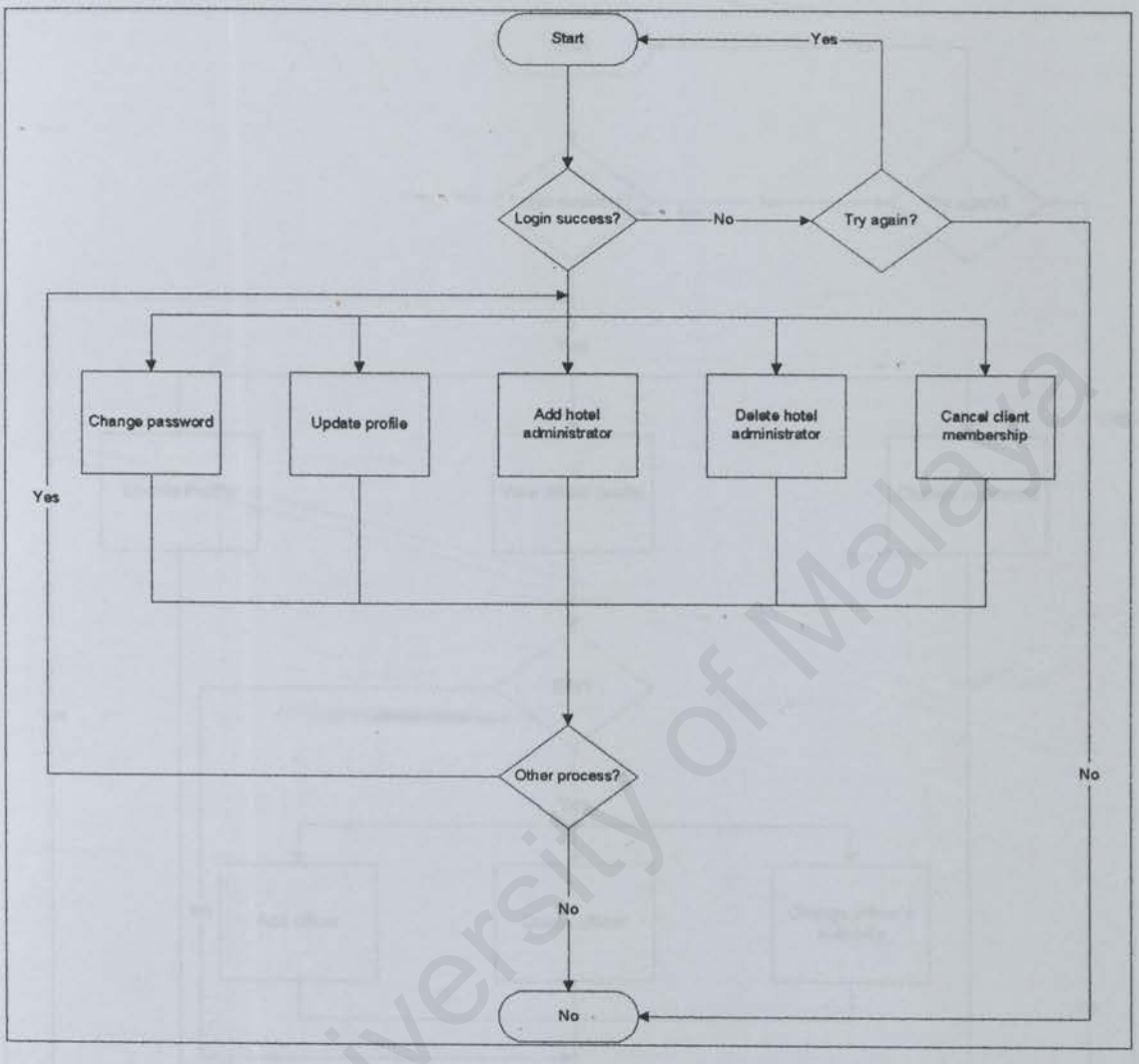

Figure 4.8 Process flow chart of portal administrator section.

4.3.3 Hotel Administrator section

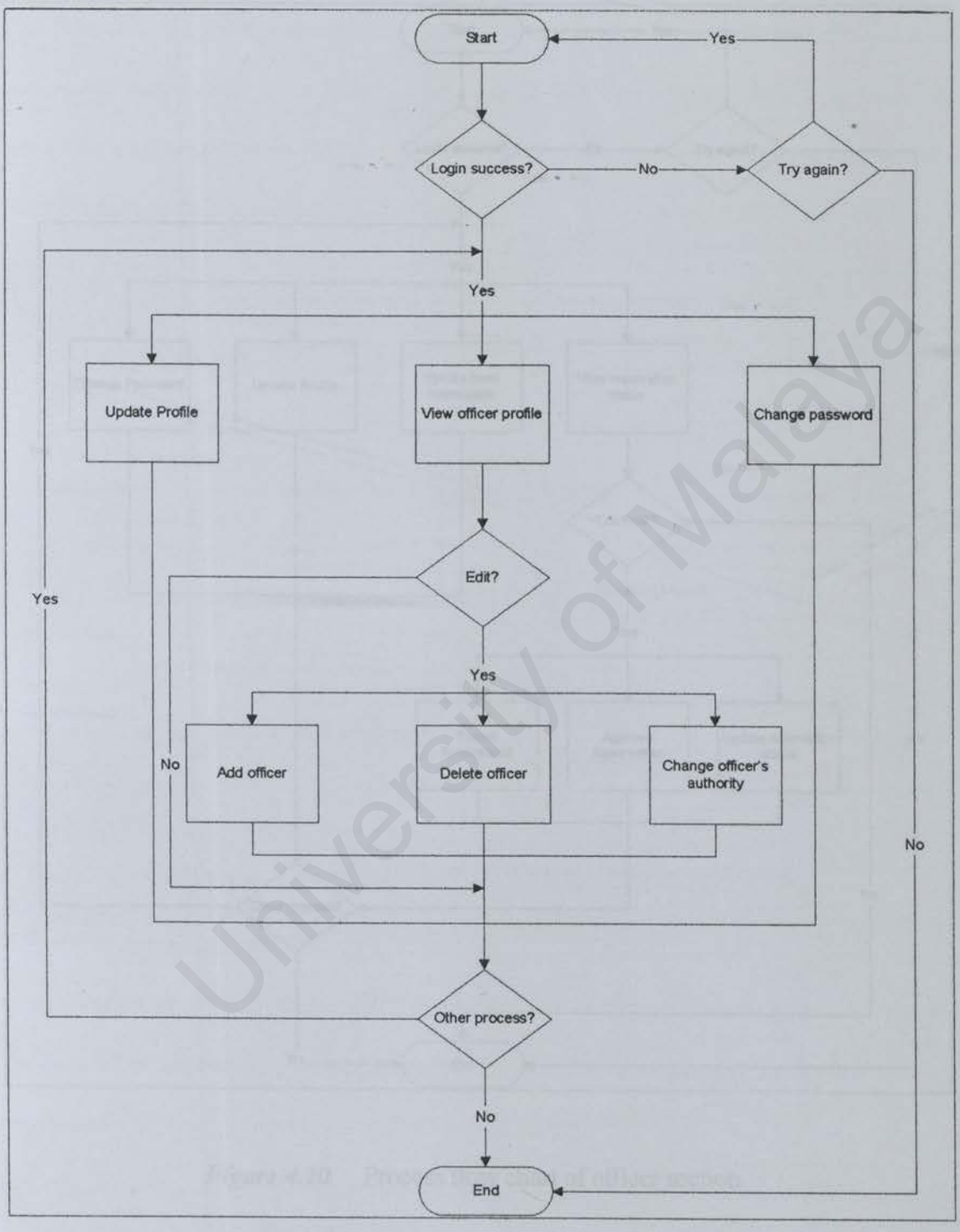

Figure 4.9 Process flow chart of hotel administrator section.

4.3.4 Officer section

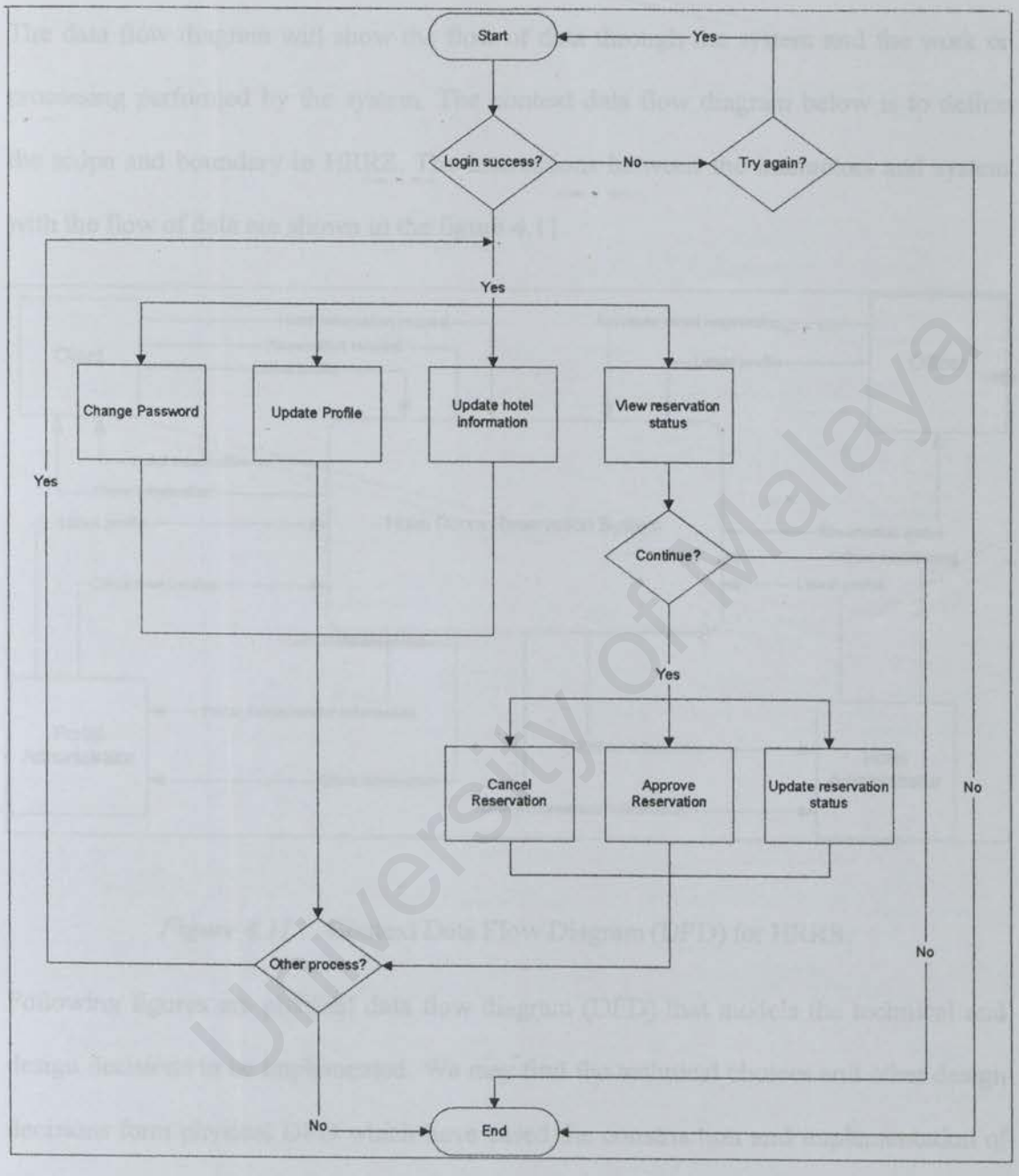

Figure 4.10 Process flow chart of officer section.

4.4 Data Flow Diagram

The data flow diagram will show the flow of data through the system and the work or processing performed by the system. The context data flow diagram below is to define the scope and boundary in HRRS. The interactions between the interactors and system with the flow of data are shown in the figure 4.11.

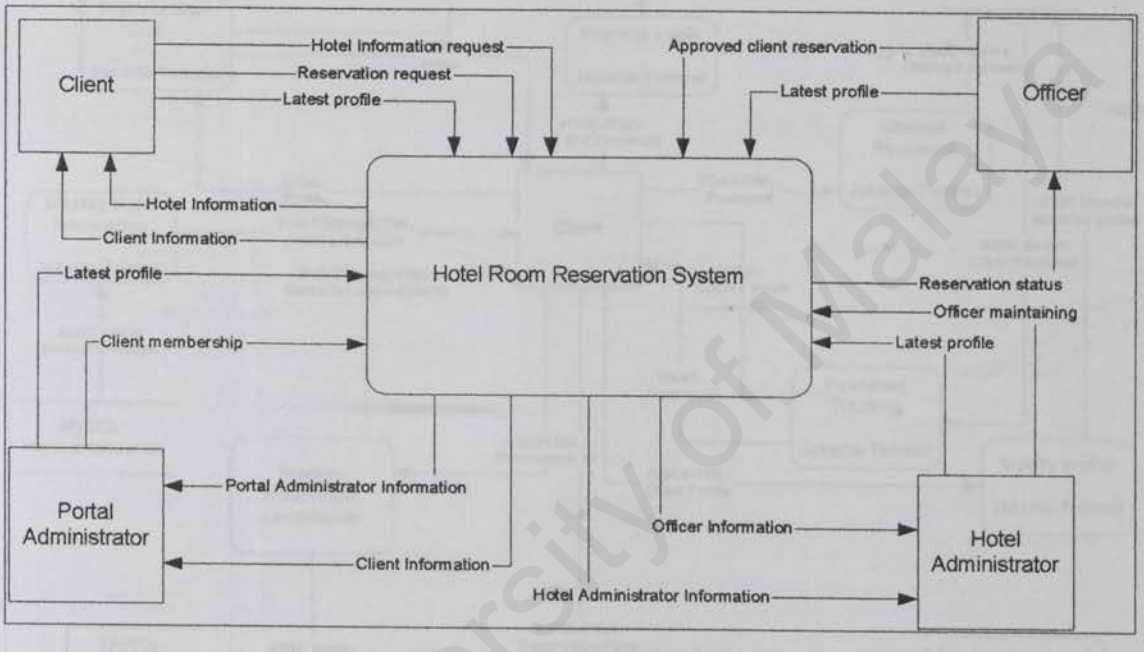

Figure 4.11 Context Data Flow Diagram (DFD) for HRRS.

Following figures are physical data flow diagram (DFD) that models the technical and design decisions to be implemented. We may find the technical choices and other design decisions form physical DFD which have eased the construction and implementation of the system.

4.4.1 Client section DFD

There are two levels of DFD for the client section. The level 0 shows the outline of the client section DFD. For some process such as reservation will discuss further in the level 1 DFD.

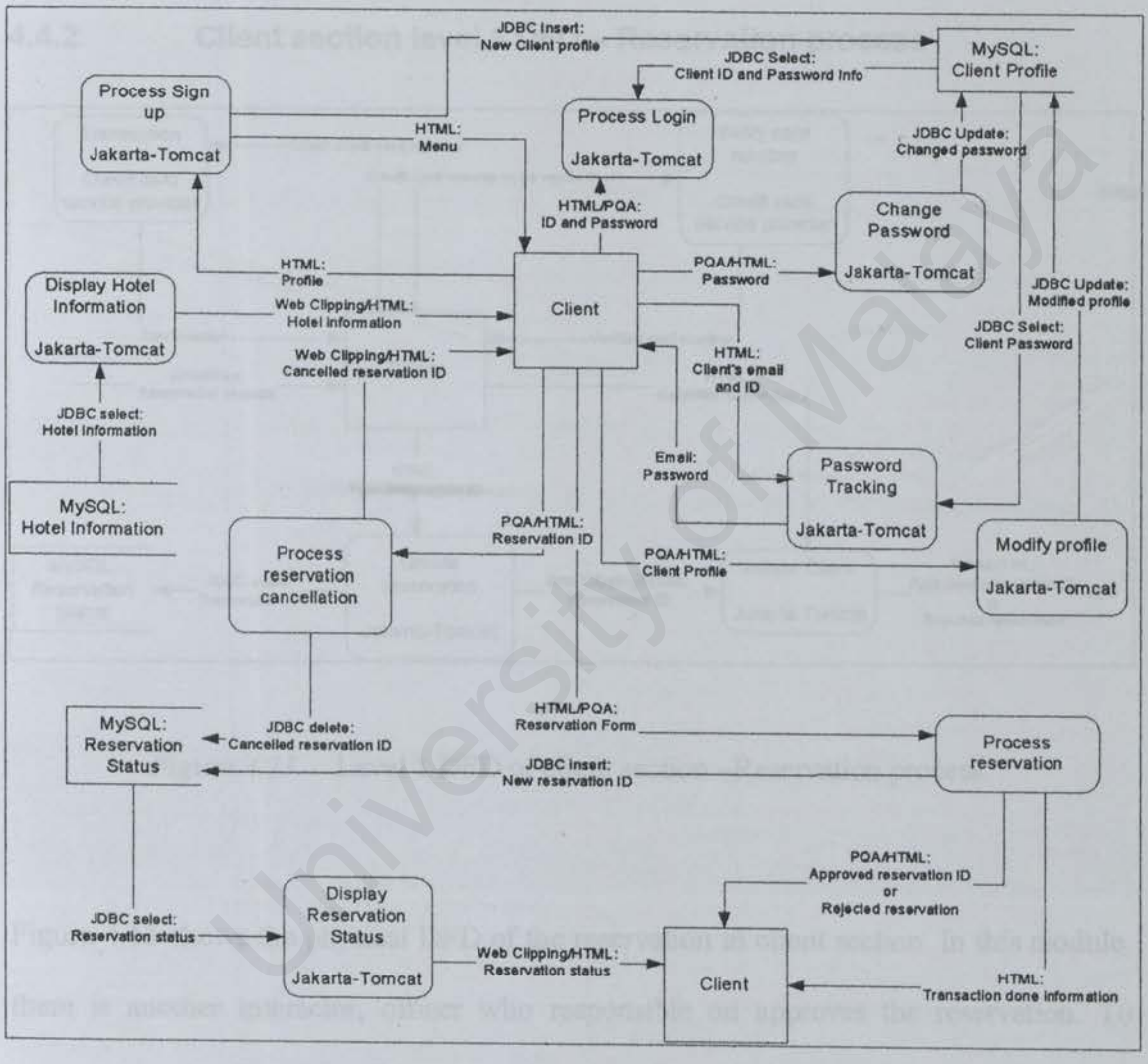

Figure 4.12 Level 0 DFD of client section.

Mainly, the Jakarta-Tomcat web server is responsible to process all the processes. For the presentation tier, the data will be presented either in HTML pages if the client using

desktop or PQA if client using Palm. Generally, the static HTML pages are stored in the web server but the dynamic HTML pages will generated by JSP. The JDBC is used to interact with the data store.

4.4.2 Client section level 1 DFD – Reservation process

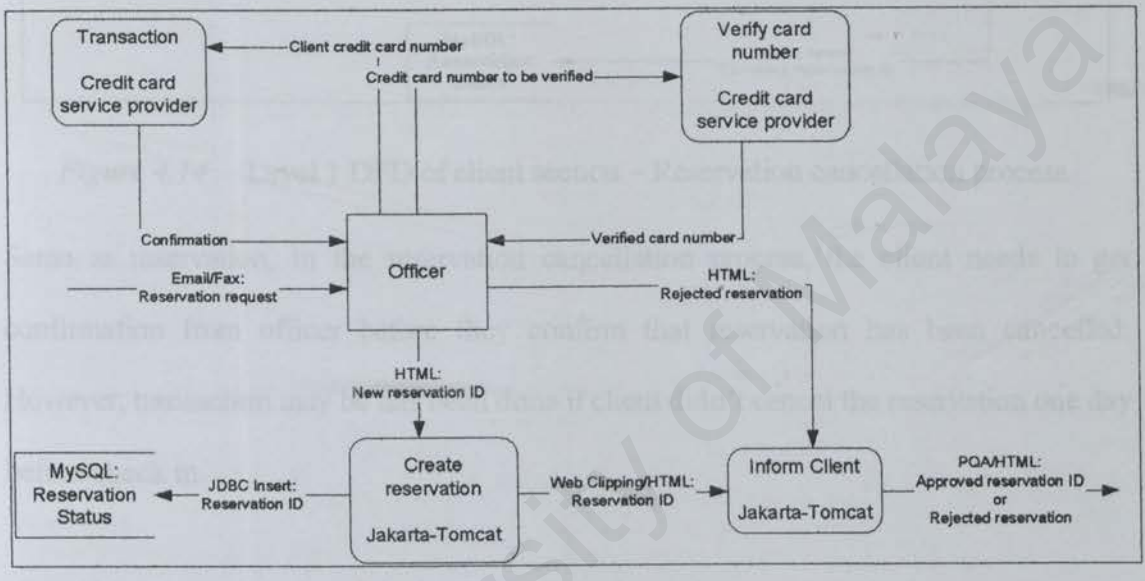

Figure 4.13 Level 1 DFD of client section –Reservation process.

Figure 4.13 shows the physical DFD of the reservation in client section. In this module, there is another interactor, officer who responsible on approves the reservation. To approve a reservation, the officer need to login to the system to create a reservation. The server will then inform the client that his reservation has been done with the reservation ID attached. The reservation status data store will be updated as well at the same time.

4.4.3 Client section level 1 DFD – Reservation cancellation process

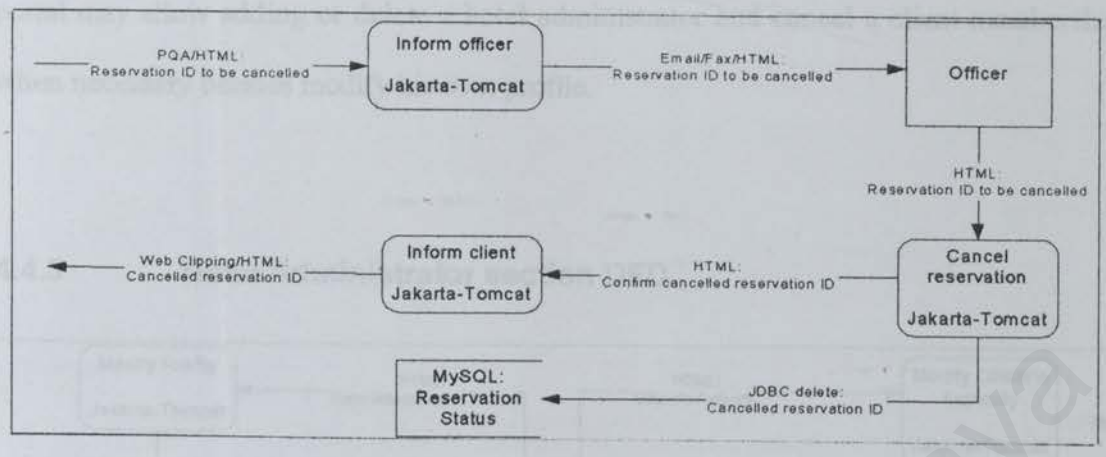

Figure 4.14 Level 1 DFD of client section – Reservation cancellation process.

Same as reservation, in the reservation cancellation process, the client needs to get confirmation from officer before they confirm that reservation has been cancelled. However, transaction may be has been done if client didn't cancel the reservation one day before check in.

4.4.4 Portal Administrator section DFD

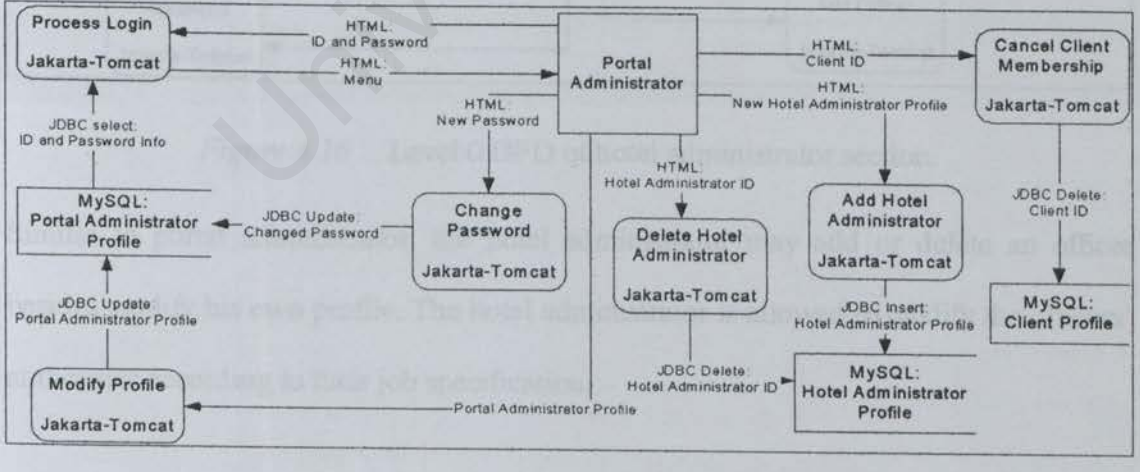

Figure 4.15 Level 0 DFD of portal administrator section.

The DFD of portal administrator section only presents in level 0. From Figure 4.15, the portal may allow adding or delete a hotel administrator and cancel a client membership when necessary besides modify his own profile.

4.4.5 Hotel Administrator section DFD

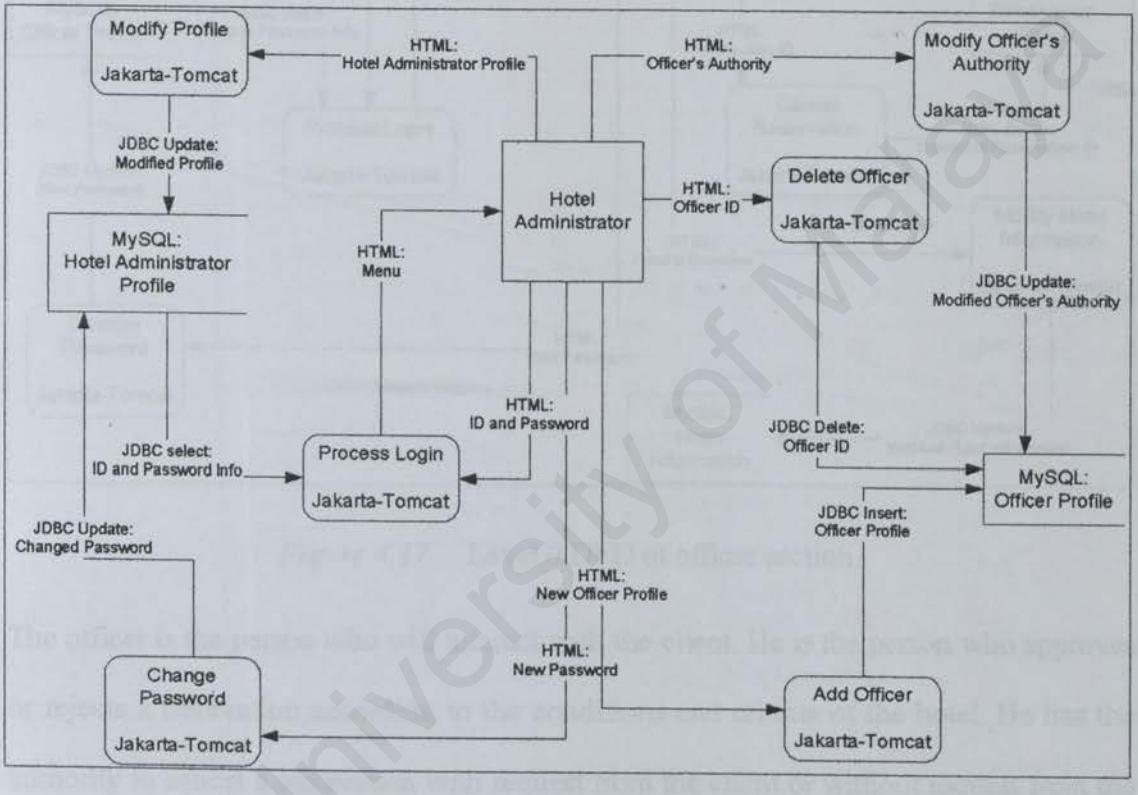

Figure 4.16 Level 0 DFD of hotel administrator section.

Similar to portal administrator, the hotel administrator may add or delete an officer besides modify his own profile. The hotel administrator is allowed to modify the officers' authorities according to their job specification.

4.4.6 Officer section DFD

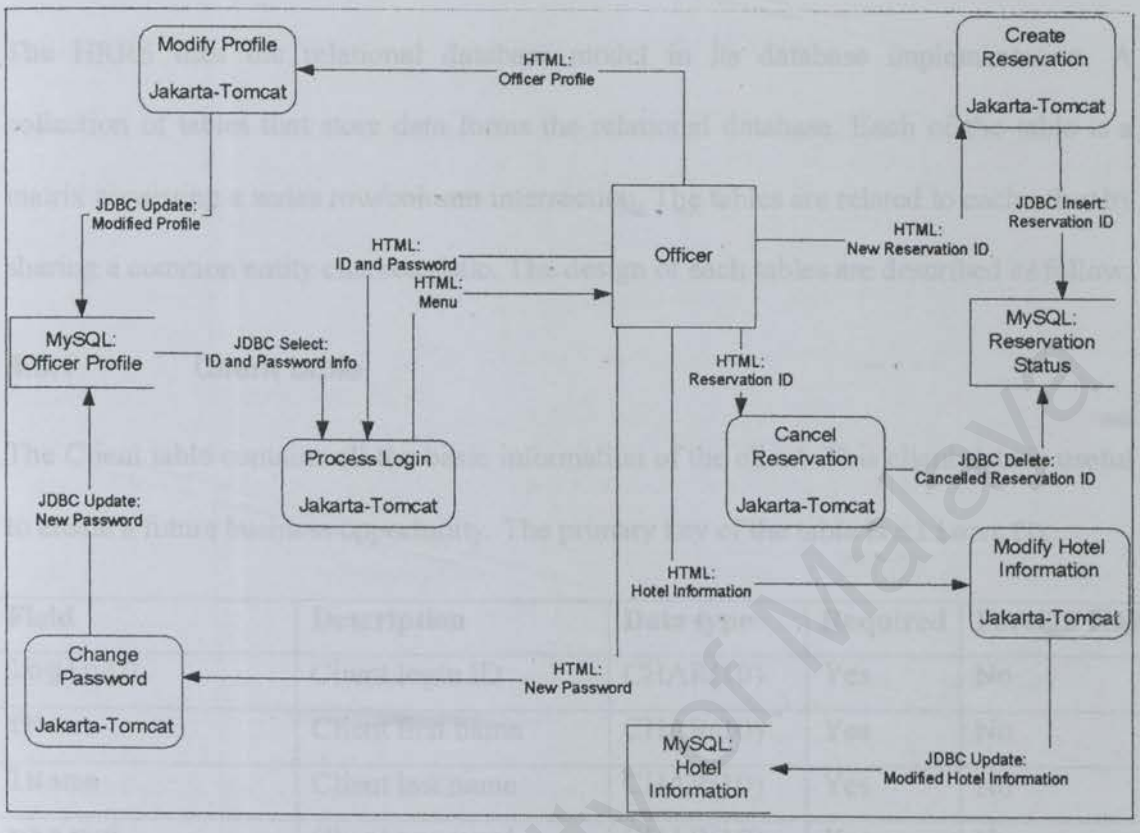

Figure 4.17 Level 0 DFD of officer section.

The officer is the person who will interact with the client. He is the person who approves or rejects a reservation according to the conditions and criteria of the hotel. He has the authority to cancel a reservation with request from the client or without request from the client when necessary.

4.5 Database Design

The HRRS uses the relational database model in its database implementation. A collection of tables that store data forms the relational database. Each of the table is a matrix consisting a series row/column intersection. The tables are related to each other by sharing a common entity characteristic. The design of each tables are described as follow:

4.5.1 Client table

The Client table contains all the basic information of the client. This client data is useful to create a future business opportunity. The primary key of the table is `clientID`.

Field	Description	Data type	Required	Foreign Key
`loginID`	Client login ID	CHAR(10)	Yes	No
`fName`	Client first name	CHAR(10)	Yes	No
`lName`	Client last name	CHAR(10)	Yes	No
`passwd`	Client password	CHAR(10)	Yes	No
`date`	Client date of birth	INTEGER(2)	Yes	No
`monthID`	Client month of birth	INTEGER(2)	Yes	No
`year`	Client year of birth	INTEGER(4)	Yes	No
`postcode`	Postcode	LONG	Yes	No
`cityID`	City ID	INTEGER	Yes	Yes
`stateID`	State ID	INTEGER	Yes	Yes
`countryID`	Country ID	INTEGER	Yes	Yes
`email`	Client's email	CHAR(30)	Yes	No
`profID`	Profession ID	INTEGER	No	Yes
`gender`	Gender	Char (2)	Yes	No

Table 4.1 Client table.

4.5.2 Month table

The Month table contains a set of month from January to December.

Field	Description	Data type	Required	Foreign Key
monthID	Month ID	INTEGER(2)	Yes	No
month	Month	CHAR(15)	Yes	No

Table 4.2 Month table.

4.5.3 CreditCard table

The creditCard table contains the credit card company with their acronym.

Field	Description	Data type	Required	Foreign Key
creditCardCoID	Credit Card Company ID	INTEGER(2)	Yes	No
creditCardCoName	Credit Card Company Name	CHAR(30)	Yes	No
shortName	Acronym for credit card company	CHAR(20)	No	No

Table 4.3 CreditCard table.

4.5.4 Prof table

The Prof table contains records about the profession of the user. The primary key of the table is profID.

Field	Description	Data type	Required	Foreign Key
profID	Profession ID	INTEGER	Yes	No
profName	Profession Name	CHAR(20)	Yes	No

Table 4.4 Prof table.

4.5.5 Reservation table

The Reservation table is used to store reservation information for each different reservation ID. The primary key of the table is `reservationID`.

Field	Description	Data type	Required	Foreign Key
`clientID`	Client loginID	CHAR(20)	Yes	No
`reserveID`	Reservation ID	INTEGER	Yes	No
`checkInDate`	Date of check in	DATETIME	Yes	No
`checkOutDate`	Date to check out	DATETIME	Yes	No
`apStatus`	Reservation approve status	ENUM(WA, AP, NA)	Yes	No
`caStatus`	Reservation cancel status	ENUM(CWA, CAP, CNA)	Yes	No
`overStatus`	Reservation overdue status	ENUM(OV, NO)	Yes	No
`roomID`	Room ID	INTEGER	Yes	No
`payment`	Payment to pay	DECIMAL	Yes	No
`noAdult`	Number of adult	INTEGER(1)	No	No
`noChild`	Number of children	INTEGER(1)	No	No
`creditCardNo`	Credit card number	INTEGER(20)	Yes	No
`creditCardCoID`	Credit card company ID	INTEGER(2)	Yes	Yes
`expiration`	Credit card expiration date	TIMESTAMP(4)	Yes	No
`name`	Client name.	CHAR(50)	Yes	No
`requirement`	Client's additional requirements.	LONG TEXT	No	No

Table 4.5 Reservation table(continue).

4.5.6 City table

The City table is used to keep the cities' name. Primary key is `cityID`.

Field	Description	Data type	Required	Foreign Key
`cityID`	City ID	INTEGER	Yes	No
`cityName`	City name	CHAR(20)	Yes	No
`countryID`	Country ID	CHAR(2)	Yes	Yes

Table 4.6 City table.

4.5.7 State table

This table holds record about the states' name. The primary key of the table is `stateID`.

Field	Description	Data type	Required	Foreign Key
`stateID`	State ID	INTEGER	Yes	No
`stateName`	State name	CHAR(20)	Yes	No
`countryID`	Country ID	CHAR(2)	Yes	Yes

Table 4.7 State table.

4.5.8 Country table

The Country table keeps all the country name around the world. The primary key of the table is `contryID`.

Field	Description	Data type	Required	Foreign Key
`countryID`	Country ID	INTEGER	Yes	No
`countryName`	Country name	CHAR(20)	Yes	No

Table 4.8 Country table.

4.5.9 Grade table

This table is collection of grading of hotels. The primary key of the table is `gradeID`.

Field	Description	Data type	Required	Foreign Key
`gradeID`	Grade ID	INTEGER	Yes	No
`gradeName`	Grade name	CHAR(20)	Yes	No

Table 4.9 Grade table.

4.5.10 Location table

The location table contains geographical location of hotels. The primary key of the table is `locationID`.

Field	Description	Data type	Required	Foreign Key
`locationID`	Location ID	INTEGER	Yes	No
`locationName`	Location name	CHAR(20)	Yes	No

Table 4.10 Location table.

4.5.11 Hotel table

Hotel table stores records of hotels with basic information of the hotel. The `gradeID`, `locationID` and `roomID` describe the ranking, location and room information of the hotel. The Hotel table primary key is `hotelID`.

Field	Description	Data type	Required	Foreign Key
`hotelID`	Hotel ID	INTEGER	Yes	No
`hotelName`	Hotel name	CHAR(30)	Yes	No
`address`	Hotel address	CHAR(200)	Yes	No
`postcode`	Postcode	LONG	Yes	No
`cityID`	City ID	INTEGER	Yes	Yes
`stateID`	State ID	INTEGER	Yes	Yes
`countryID`	Country ID	INTEGER	Yes	Yes
`tel`	Hotel telephone number	CHAR(20)	Yes	No
`fax`	Hotel fax number	CHAR(20)	No	No
`email`	Hotel email	CHAR(30)	No	No
`description`	Description of hotel	LONG TEXT	Yes	No
`gradeID`	Grade ID	INTEGER	Yes	Yes
`locationID`	Location ID	INTEGER	Yes	Yes

Table 4.11 Hotel table.

4.5.12 Room table

The Room table contains the information about the room rate and room type. Primary key is `roomID`.

Field	Description	Data type	Required	Foreign Key
roomID	Room ID	INTEGER	Yes	No
roomType	Room type	CHAR(10)	Yes	No
roomRate	Room rate	CURRENCY	Yes	No
roomTotal	Total of room	INTEGER(2)	Yes	No
roomDescription	Description of room	LONG TEXT	No	No
HotelID	Hotel ID	INTEGER(5)	Yes	No

Table 4.12 Room table

The entity relation (ER) diagram in Figure 4.18 shows the relationship among the Client and Hotel with the other tables.

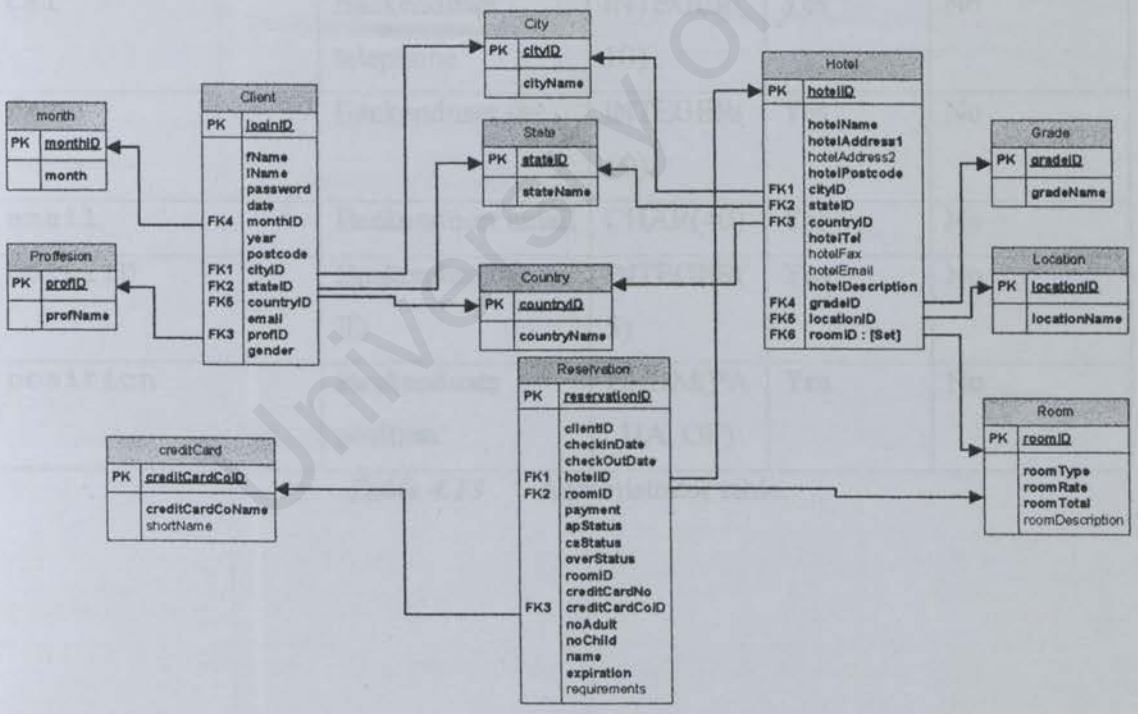

Figure 4.18 ER diagram of Client and Hotel with the other tables.

4.5.13 Backenduser table

The backenduser table keeps records for both the hotel and portal administrators. The `position` in the table determines the position and the authority of the backenduser : PA – Portal Admin, HA – Hotel Admin and OF – Officer. The `hotelID` is the hotel that backenduser belonging to or 0 if the backenduser is portal admin. The primary key of the table is `loginID`.

Field	Description	Data type	Required	Foreign Key
`loginID`	Login ID	CHAR(10)	Yes	No
`name`	Backenduser name	CHAR(50)	Yes	No
`passwd`	Backenduser password	CHAR(10)	Yes	No
`tel`	Backenduser telephone	INTEGER(10)	Yes	No
`fax`	Backenduser fax	INTEGER(10)	Yes	No
`email`	Backenduser email	CHAR(40)	Yes	No
`hotelID`	Backenduser hotel ID	INTEGER(5)	Yes	No
`position`	Backenduser position	ENUM(PA, HA, OF)	Yes	No

Table 4.13 Administrator table.

4.6 User Interface Design

The user interface for the HRRS has also been given top priority beside the functionality design. For the hotel administrator, portal administrator, officer and part of the function for client, the interface will base on web pages. Therefore, HTML, JSP, Java Servlet and JavaScript will be the presentation tools. Besides the web pages, part of the client function will be ran on Palm Query Application (PQA). The Web Clipping technology will be implemented in this scenario.

The following tables show the interface styles for each function after considering their functionality.

Functions/Activities	Data	Perform Style	Components
Menu	Selected menu	Headline	Hyperlink
Login	Login name	Form	Text field
	Login password	Form	Password field
	Submit	Form	Button
Sign Up	User information	Form	Text field, password field, check box, combo box, radio button, submit button
Password tracking	Email and Login name	Form	Text field, submit button
	Password	Email	Text
Profile viewing	User profile	Table	Text

Table 4.14 User interface for different functions and activities.

Functions/ Activities	Data	Perform Style	Components
Profile editing	User profile	Form	Text field, password field, check box, combo box, radio button, confirmation button, message dialog box
Maintaining a user profile	New user profile	Form	Text field, check box, combo box, radio button
	User ID to delete	Table	Check box, confirmation button, message dialog box
	Authority to modify	Form	Check box
Reservation	Client ID and basic information	Form	Text field
	Credit card number	Form	Password field
	Selection for room and hotel information	Form	Combo box, radio buttons, check box, submit button
	Reservation status	Table	Text
	Reservation ID to delete	Form	Check box, submit button, message dialog box
	Reservation ID to update	Form and table	Check box, submit button, message dialog box
	Reservation ID to approve	Form and table	Check box, submit button, message dialog box

Table 4.15 User interface for different functions and activities (continue).

Table 4.15 and 4.16 show that the user interface to get input from user or display output to user. Each different kind of data has different respecting style to be performed either in

form view or table view or headline view. Components will be displayed are included in the table as well.

4.7 Discussion

The system design provides a guideline to system developers in implementing a system. Thus, a good system design is an important key to a successful system.

The system design we have just discussed is a result of analysis of the system requirements. However, some requirements may be left from consideration unexpectedly. Therefore, the prototyping and incremental development will take place after this phase.

After this point in the project, a prototype for the system will be developed. The feedbacks from users will be encountered and the definite system will emerge after some degree of iteratively modifications according to the feedbacks analysis.

Chapter 5: System Implementation

5.1 Implementation on MVC Design Pattern

The Model, View and Controller (MVC) design pattern was used in the system implementation. In this approach, all objects in the system (model) will be presented using appropriate codes (view) and controlled by the controller.

5.1.1 Model

Models in this system are all the objects that will interact with the systems including the system user and data that will be presented. There are seven classes to classify these objects: `PalmUser`, `Client`, `Padmin`, `Hadmin`, `Officer`, `Hotel` and `Room`.

5.1.2 View

This is the peripheral for the user to interact with the system. Coding is in JSP or HTML. The JSP will be transformed to servlet and then to HTML to display in the web browser or web clipper.

5.1.3 Controller

The servlets are acting as controllers to controller the receive input or requests from user and then perform the appropriate process including instantiate objects. The servlets

change the objects' states and response to user by parsing the objects to JSP to display the output via the HttpSession.

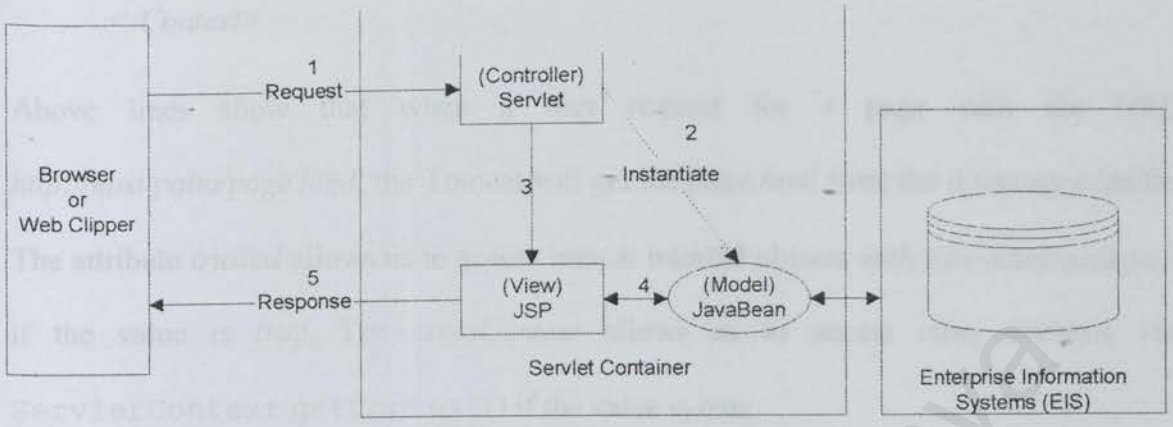

Figure 5.1 MVC Design Pattern

5.2 Web Server Configuration

Before running the Tomcat server, we need to setup the system environment in order to run the server. A new environment variable TOMCAT_HOME has to be added to point to the root directory of the Tomcat hierarchy. In addition, the JAVA_HOME has to be added to point to the root directory of the JDK hierarchy and add the Java interpreter in the PATH of the environment variable.

The Tomcat is configured using the server.xml in TOMCAT_HOME/conf folder. The server.xml providing initial configuration for the Tomcat components and specifying structure for Tomcat, meaning, letting Tomcat boot and build itself by instantiating components as specified in this file.

To add a new context path in the Tomcat we may do below configuration in server.xml.

```
<Context path="/palm" docBase="e:/palm" crossContext="false" debug="0"
reloadable="true">
</Context>
```

Above lines show that when a user request for a page with the URL *http://host/palm/page.html*, the Tomcat will get the *page.html* from the directory e:/palm. The attribute *trusted* allows us to access tomcat internal objects with `FacadeManager` if the value is *true*. The *crossContext* allows us to access other contexts via `ServletContext.getContext()` if the value is *true*.

5.3 Database Implementation

All tables in of TGOP - HRRS are divided into three databases: `Client`, `Hotel` and `Admin`. Generally, there are four users using these databases who are Client, Padmin (Portal Admin), Hadmin (Hotel Admin), Officer and Backenduser. Thus, in MySQL database, each group of user will be grant for certain privileges to access certain databases or tables. For example, the Client may allow performing action such as INSERT, SELECT and UPDATING in the `Client` database and the Padmin only may allow to SELECT on the `Client.client` table. This is to conform the authentically and integrity as mentioned in the system requirements.

The Client information will be stored in the `Client.client` table. All information that related to the Client such as reservation table will be stored in the `Client` database as well. The `Hotel` database will stores the hotel information such as `hotel` table,

`location` table and `room` tables. And the `Admin` database only consist of the `backenduser` table which storing the backend user data.

When user attempt to access the database, the respecting classes of the user for example `Client` for Client will connect to the database using the `mm.mysql` driver and connected as user Client. If the user is attempt to access sensitive data, `password` and `loginID` will be required to identify themselves.

The database is named using the format of `Database.table.column`.

5.4 SSL Implementation

TGOP - HRRS is using Java Secure Socket Extension (JSSE 1.0.2) to implement SSL and TLS. The JSSE 1.0.2 comes standard with a provider named `SunJSSE` which implement SSL 3.0 and TLS 1.0 security protocols.

To implement the SSL, we need to obtain a certificate from a Certificate Authority (CA). In this project, a self-signed certificate has been generated using the keytool command that provided Java Development Kit (JDK 1.3.1).

```
keytool -genkey -alias tomcat -keyalg RSA
```

Above command will generate a X.509 certificates (keystore) with RSA key and password "changeit". This certificate may signed by a CA by exporting it to the CA or self-signed.

At the same time, put the JSSE jars (jcert.jar, jnet.jar and jsse.jar) in the classpath so that

Tomcat server can implement them. Besides that, the following line

security.provider.2=com.sun.net.ssl.internal.ssl.Provider

need to be added in the java.security in the JAVA_HOME/jre/lib/security folder.

The Tomcat server will use the port 8443 (standard port) to provide the secure communications by using https URL protocol. When a Client attempt to do a reservation, the server will redirect the Client to https to provide secure communication on the business information such as credit card information.

In the server.xml, we need to uncomment the part in *<connector>* where

<Parameter name "port" value "8443"/>

and add below lines to point the Tomcat to the *.keystore* and to provide the *keypass* "*changeit*" to the server.

<Parameter name="keystore" value="/tomcat/conf/.keystore"/>
<Parameter name="keypass" value="changeit"/>
<Parameter name="clientAuth" value="false"/>

The parameter *clientAuth* indicates that whether the user need to authenticate the certificate or not.

5.5 JavaMail Implementation

TGOP - HRRS is using email to send notification to the Client after the Client has done an important action for instance after signed up, made a reservation and cancelled a reservation. The TGOP - HRRS using the *siswazah.fsktm.um.edu.my* SMTP server to send these notification.

To implement the mail sending in the TGOP - HRRS, the JavaMail 1.2 jar (mail.jar) and JavaBean Activation Framework 1.0.1 jar (activation.jar) need to add in the classpath. The mail.jar provides the core classes to send and receive mail. The activation.jar is to support basic MIME-type in most browsers and mail tools.

```
Properties props = System.getProperties();
props.put("mail.smtp.host",
     "siswazah.fsktm.um.edu.my");
Session mailSession =
     Session.getDefaultInstance(props, null);
```

Above codes setting the system properties with the *siswazah.fsktm.um.edu.my* SMTP server. After setting the system properties, a `Session` created to allow information sharing in the program.

```
//Define message
MimeMessage message =new MimeMessage(mailSession);

message.setFrom(
     new InternetAddress(fromAddress, fromPersonal));
message.addRecipients(Message.RecipientType.TO,
     InternetAddress.parse(to));
message.setSubject(subject);
message.setText(text);

//send message
Transport.send(message);
```

Following with the codes above, a MimeMessage created. All required information may provide in the email by invoking methods `setFrom(Address)`, `setRecipients(Message.RecipientType,Address[])`, `setSubject(String)` and `setText(String)`. After that, `Transport.send(MimeMessage)` will send the message according to the information provided.

5.6 Implementation on Palm

The TGoP so call Travel Guide on Palm, therefore, the TGoP–HRRS has to be able to run on the Palm OS device. We use POSE to emulate the real Palm OS device and outline the system on Palm.

The POSE may download from *http://www.palm.com/devzone/pose/*. To testing the system over Internet on POSE, we use the Palm Emulator proxy server with IP address 207.240.80.136.

The web clipping application has prepare some clipping interface elements that special designed to enable the web clipping application page and clipping to perform a Palm OS friendly interface.

Below showing some of the elements that used in the TGoP-HRRS:

- `PalmComputingPlatform`

  ```
  <meta name="palmcomputingplatform" content="true">
  ```

  Clipping without above tag (`PalmComputingPlatform`) are truncated by the proxy server to 1KB and images will be removed.

- `LocalIcon`

  ```
  <meta name="localicon" content="image.gif">
  ```

  The `LocalIcon` tag enable the web clipping application to store image files in the application. It is also allows other clipping and web clipping application to call this file (image.gif) using

```
<img src="file:myapp.pqa/image.gif">
```

This is to reduce the transmission of data over the air which the transmission may reduce the system performance.

- `DatePicker`

```
<input type="datepicker" name="date" value="value">
```

The input type `DatePicker` will call the Palm OS application to allow user to assign date. The date is displayed to the user in MM/DD/YY form but returned to the server in YYYY-MM-DD form.

5.7 Modules Implementation

All modules in TGOP - HRRS can be divided into four sections: Client, Portal Administrator, Hotel Administrator and Officer.

5.7.1 Client Section

5.7.1.1 Sign Up Module

The `SignUp` servlet will handle the signing up process. In this module, all required information will be checked by respecting procedure to make sure the information provide is valid to conform the integrity of data. For example, the provided email address will be checked so that the email address is in the form of *username@domain.domain*. A notification email will send to client for the successful sign up. This module only provides service to the desktop.

5.7.1.2 Login Module

The login module is controlled by the `Login` and `PalmLogin` servlets. The client is required to give the password and login name in order to login to the system. By parsing the `servletPath` of the requested page, the client can be forwarded to the page where they are login. For example, when client has found a suitable room to book before login, the client may login at that time and able to trace back the same information after login. The `PalmLogin` is used to handled request from palm.

5.7.1.3 Reservation Module

The reservation module is handled by `Reserve` and `PalmReserve` servlets. These servlet not only handled the whole reservation process, it's also handled room searching, reservation retrieving and reservation cancellation.

The room searching procedure allows client to search for the suitable and available room based on the client's criteria such as room type, room rate, location of the hotel and check in and check out date. If the required rooms are not available, the client still may get the room information but they can't making any reservation on it.

After the room reservation and cancellation requests sent to the server, the Reserve servlet will invokes the `sendMail()` method to send a email to notify client that request has been received and will be processed in three working days.

5.7.1.4 Password tracking module

`Forgot` and `PalmForgot` servlets are used to handle the password tracking if the client has forgotten his password. The client required to give their login name and email that has registered. If the login name and email are matched to the database, a email with client password and login name will be sent to the client.

5.7.1.5 User Profile Update Module

This module is handled by `ChangePass` and `Update` servlets and only provided on desktop. The `ChangePass` servlet receives old and a pair of new passwords from client. If the old password is match to the database password and the pair of new passwords are matching and follow the integrity of password, the password will be changed. The `Update` servlet also will check for the data integrity on the user data input. Error messages will be displayed and data will not be updated until the client has provided the valid data.

5.7.1.6 Logout Module

The logout module is using `Logout` servlet to invalidate all the objects that bounded to the session. Method `invalidate()` will be invoked and all bounded objects to the `HttpSession` will be released from the session. The Client will not able to do action that need to login such as reservation after logging out provided Client login again.

5.7.2 Backend User Section

5.7.2.1 Backend User Login Module

All three backend users, Portal Administrator, Hotel Administrator and Officer are using the same servlet to login. The `BUlogin` servlet will firstly connects to the database using backenduser as the user name. The user backenduser in the database is allowed to SELECT on the `Admin.backenduser` table. If the login name and password that provided by the backend user are matched to the data in table `Admin.backenduser`, the servlet will reconnect to the database according to the position of the backend user in which stored in the `Admin.backenduser.position` column.

5.7.2.2 Backend User Profile Update Module

The backend users are using `BUupdate` and `BuchangePass` servlets to update their profile and password. Basically the procedures and processes are same as Client's profile updating besides these servlets will identify the position of the user at the beginning before continue the processes of password changing and profile updating.

5.7.2.3 Backend User Logout Module

This module is handled by `BUlogout` and same as Client's logout module which unbound all bounded objects in the `HttpSession` by invoking `invalidate()` method.

5.7.3 Portal Administrator Section

5.7.3.1 Portal Administrator Administration Module

This module only used by the root user who is the super user of the system. The root basically has same functionality as other Portal Administrators besides he can monitor, add and delete portal administrator using `AdminPadmin` servlet. JavaScript is used to prompt alert message to user before deletion to avoid accidental deletion.

5.7.3.2 Hotel Administration Module

`AdminHotel` servlet is used to view, add and delete hotels that compromise on provide TGOP - HRRS services. The user is required to add at least one Hotel Administrator right after adding a new hotel. On deletion of a hotel, all the hotel administrators and officers data of regarding hotel will be deleted also.

5.7.3.3 Hotel Administrator Administration Module

The `AdminHadmin` servlet provides view, add and delete on the hotel administrator. However, the deletion of hotel administrator will not delete the officer under the hotel administrator.

5.7.3.4 Client Administration Module

This module allows portal administrator to find out who are the members that have not login for a long time such as more than six months. The portal administrator may cancel the membership of the Client if the portal administrator found that the Client was not active for a long time. `AdminClient` is used to handle these actions.

5.7.4 Hotel Administrator Section

5.7.4.1 Officer Administration Module

The hotel administrator use AdminOfficer servlet to view, add and delete Officers.

5.7.5 Officer Section

5.7.5.1 Hotel Information Updating Module

Officer is using the `UpdateHotel` servlet to update the hotel information. Required inputs that not conform to the data integrity will be required to enter again. The hotel information will not be updated provided all the inputs are valid.

5.7.5.2 Room Administration Module

The Officer may use the AdminRoom servlet to view, add, delete and edit room information of the hotel. Data integrity has to be conform too in the add and edit actions.

5.7.5.3 Reservation Administration Module

This module allows Officer to keep track the reservation that have made by the Client. The `AdminReserve` servlet allows Officer to search for appropriate reservation according to the criteria provided such as check in date, pay status and approve status. The page that display the result will be refreshed after every 2 minutes for Officer to keep track the latest reservation using the follow meta tag in HTML.

```
<META http-equiv="refresh" content="120">
```

The reservation that overdue will be highlighted in red color. The Officer may also approve/not approve a reservation or approve/not approve a cancellation based on the Client request and the availability of room required. Besides, the Officer has to verify the provide credit card information from the appropriate credit card company. A notification will be sent to the Client automatically via email after the approval has been done.

5.8 System Coding

5.8.1 Coding Style

All the codes written in Java is following Java programming standard which all the class name are in title case, for example `Hotel`, `Room` and `Client`. Methods and attributes in classes are named in lowercase except for name that is combined with a few words. The second word and words onwards of the name will use title case to differentiate each word such as `isValidPasswd(String passwd)` is combination of "is valid password". The indentation is used to differentiate each clause such as in the control structure (`for` and `while` loop), exception handling clauses (`try` and `catch` clause) and conditional structure (`if` and `else` clause). Codes below shows the coding style in `Client` class.

```
package tgop.bo;
import java.beans.*;
…
public class Client extends PalmUser{
        private String fName, postcode;
        …
        public void buildCon(){
                …
                try{
                        …
                }
                catch ( ClassNotFoundException cnfex ) {
                        …
                }
        }
        …
}
```

5.8.2 Documentation

Internal documentation in the source code was used to provide information that identifies the program, describe its data structures, algorithms and control flow. The header comment block approach is used to provide above-mentioned information. The documentation style is following standard that has specified to implement *javadoc* tool. Codes below showing the example of the header comment block of a program:

```
/**
 * Reserve.java
 * @author  Wong Pak Liang
 * Purpose To provide reservation services for Client including
 * room searching, reservation, cancel reservation and
 * retrieve previous reservation records.
 * Created on December 27, 2001, 4:52 PM
 */
```

At the beginning of a method, the comment block will be as follow:

```
/** Handles the HTTP <code>POST</code> method.
 * @param request servlet request
 * @param response servlet response
 */
protected void doPost(HttpServletRequest request,
          HttpServletResponse response)
```

If documentation follows the standard mentioned above, we may use the *javadoc* to generate a HTML documentation for future reference.

5.9 System Structure

The TGoP source codes are separated into two different folders: /palm and /palmadmin. The palm folder consists of all the HTMLs, JSPs, servlets, classes and images that will be accessed by the Client. The JSP files and HTML files are stored in root folder and the images files are in the */images* folder. The servlet classes are kept in the */WEB-INF/classes* folder and all objects' classes are packaged in the `tgop.bo` package and placed in the */WEB-INF/classes* folder too.

The /palmadmin consists of two sub-folders: /hrrs and /iss folders. The /hrrs folder contains the necessary files for the TGOP - HRRS and the /iss is container for Information Sub System(ISS). Figure 5.2 illustrates the system structure.

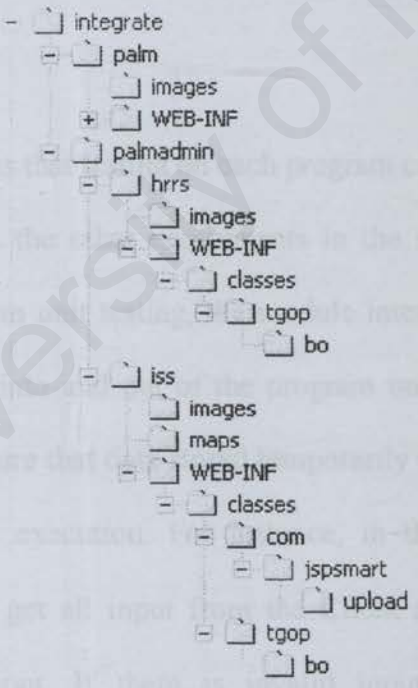

Figure 5.2 System structure.

Chapter 6: System Testing

6.1 Introduction

The purpose of system testing is to find errors and faults of the system then provides a method to correct the errors and for testing system reliability and stability. It is meant to ensure that the programs are executed correctly and conforms to the requirements specified.

The system testing starting from the unit testing, following by integration testing and functioning testing then lastly performance testing.

6.2 Unit Testing

The unit testing is the process that testing on each program component on its own and the component is isolated from the other components in the system. We may consider a module as a unit. To perform unit testing, the module interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. For instance, in the Signing Up module, the `SignUp` servlet require to get all input from the Client and stores into the database according to the given input. If there is invalid input given by Client such as *name@domain* (invalid email address) in the email address field, the `SignUp` is required to perform checking on the email address and inform Client if the address is invalid. If

the servlet failed to perform checking on the address, an algorithm fault might be occurred. Therefore, the logic of the unit, which performs checking, has to be reviewed.

6.3 Integration Testing

When collections of components have been unit-tested, the next step is ensuring that the interfaces among the components are defined and handled properly. The integration testing is the process that verifying the system components work together as described in the system and program design specifications.

The bottom-up integration is adopted in this system. The testing begins with construction and testing with atomic modules which have tested in the unit testing. For instance in Reservation module, contains atomic modules such as search room, book room, retrieve reservation and cancel reservation. After the unit testing, integration testing perform on cluster bases. The search room and book room are grouped in a cluster and testing performs on the integration of these two components. The similar testing is done to retrieve reservation records and cancel reservation modules.

Figure 6.1 illustrates the integration of the unit components in Reservation module. The clusters will be integrated into a big module after integration testing has performed on the cluster. The testing continues performs among the clusters to ensure each component integrate properly in the module. The integration testing performs on remaining modules beside Reservation module. Finally, each integrated modules will be merged and tested as a whole integrated system.

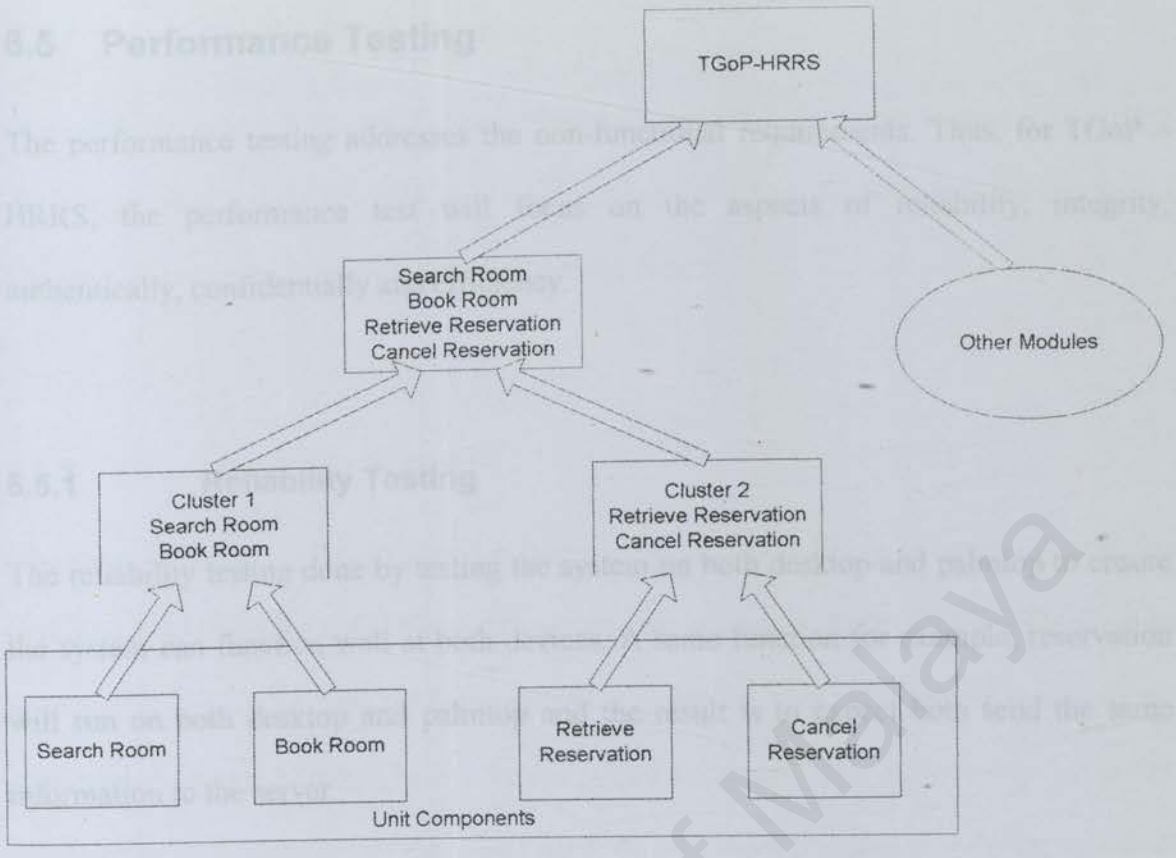

Figure 6.1 Integration testing in Reservation Module

6.4 Functioning Testing

Functioning testing is based on the system functional requirements, which are stated in the system analysis and requirements specification. For instance, in the Portal Administrator module, it may allow portal administrator to login and logout with login ID and password provided.

6.5 Performance Testing

The performance testing addresses the non-functional requirements. Thus, for TGoP – HRRS, the performance test will focus on the aspects of reliability, integrity, authentically, confidentially and efficiency.

6.5.1 Reliability Testing

The reliability testing done by testing the system on both desktop and palmtop to ensure the system can function well at both devices. A same function for example, reservation will run on both desktop and palmtop and the result is to expect both send the same information to the server.

6.5.2 Integrity Testing

Integrity testing is to ensure that unauthorized users may not access the system. Testing is done with login to the system using unregistered login ID or incorrect password.

6.5.3 Authentically Testing

Authentically testing is to ensure that users cannot perform any functions that not in his authority. For instance, a Portal Administrator may not add another Portal Administrator provided he is login as the super user. The testing done by trying to perform function that not in a user authority such as Officer tries to add a new Officer.

6.5.4 Confidentially Testing

The confidentially testing done to be conform the confidentially of user data. The testing done by trying to access a user profile after a user has logout. However, the testing on transmission over Internet with SSL was unable to test.

6.5.5 Efficiency Testing

The result from testing the efficiency of the system might not be significantly reliable due to it was tested in a LAN for the response time of the server. Although the web clipping application was tested over the Internet, but it is not the real environment it should be. The testing should be done in a wireless network because data transmission over the wireless network compare to the wired network will has a significant different response time.

Chapter 7: System Evaluation

7.1 Problems Encountered and Solutions

Through out the system development process, there are several problems encountered. Following are the major problems encountered and its solutions.

7.1.1 Inexperience in using development tools

This is the major problem that delays our development process. For example, we have chosen Forte for Java as Java programming IDE. However, we are not familiar with the features that provided by this IDE. Some of the features provided in the IDE are required extra configuration but we failed to configure it. To fully utilize this IDE, I have tried to follow the online tutorial and join in the forum in the Internet. Some of the problems have been solved but there are still some problems that haven't been solved. Fortunately, there are some alternatives to solve the problems besides using the Forte for Java. Yet, I do believe that this IDE is a powerful tool for a Java programmer if he managed to employ this tool well.

7.1.2 Inexperience in setting up SSL for Tomcat

My knowledge about Secure Socket Layer (SSL) is just theory of implementation of the protocol. I found it is difficult for me when I come to setting up the Tomcat to support SSL although I have followed the guideline that provided in the Tomcat documentation. I

have tried to use OpenSSL to generate the key and certificate and even using a free trial certificate from Verisign but failed. Finally, after reading from the Java Developer forum, I have managed to use JSSE to provide the encryption service and *keytool* of JDK to generate key and certificate then the service for HTTP/SSL has been upped.

7.1.3 Difficulties in designing appearance for Palm

The designing for web clipping application is difference from the normal web page. Due to the small screen size of PDA and expensive wireless bandwidth, the page design and the process flow have to take in the consideration of the constraints mentioned above. Thus, some of the functions have to be taken out from presenting on PDA to increase the response time to the user. Besides, the clipping pages were redesigned and rearranged their appearance so that the clipping is presentable on PDA.

7.2 System Strengths

7.2.1 Provide Secure Internet Communication

This is the minimum requirement in the e-Commerce environment. The TGoP is supported by SSL which enable secure Internet communication. The TGoP is using JSSE as the security provider that implement SSL 3.0 and Transport Layer Security (TLS) 4.0. It is to conform that security in transmission of sensitive data such as credit card information in the reservation process.

7.2.2 Wireless Access

As named, Travel Guide on Palm (TGoP) allows the user to access TGoP information anywhere with a Palm OS device provided there is a wireless access point. The user just need to download the palm query application from TGoP site previously before access TGoP services on the move.

7.2.3 Mouse/Stylus Driven

For the user's sake, the TGoP is designed in such a way to minimize the keyboard input. Most of the operations are carried out through mouse clicks or tap. The user just has to move the screen pointer to the targeted feature and click the mouse or use the stylus to tap on the targeted feature on PDA. The system would response accordingly. This will ease the user's task and save a lot of time, especially for the Palm user. Nevertheless, some of the input still required a manually input either from keyboard or using graffiti writing.

7.2.4 Separated Databases

The databases for Client, Admin and other databases are separated and managed in the different database. It is to increase the security of the databases that even the part of the database vulnerable to unauthorized user, they still are not manage to access other databases. For instance, unfortunately if an unauthorized user has succeed to access the Client database, he will only able to manipulate the data in Client database and unable to

access other database such as Hotel and Admin databases. The isolation also reduces the dependency of each database and increases the scalability and extensibility of the databases.

7.3 System Limitations

7.3.1 No Real Environment Testing

The testing of the system performance was just tested on the Internet and never been tested over the air due to there is no wireless network for testing. It might be causing the actual response time is slower than the tested time when come to the wireless environment. The intention of data losing might be increase if the amount of data too large when transfer on the wireless network which rely on the connectionless protocol, UDP yet the testing is on a more reliable connection protocol, TCP. It will reduce the system reliability and availability.

7.3.2 Limited support on Web Browser

The system testing is based on Internet Explorer 5.0. The performances of the system in other browsers such as Netscape Navigator have not been tested. Therefore, the user is recommended to use Internet Explorer 5.0.

7.3.3 Partial Integration of Back End System

Due to the uncompromised of the implementation and time constraints, the back end system is not fully integrated. The administrator, especially the portal administrator has to login to two different system in order to perform his job on both HRRS and ISS.

7.3.4 No Report Generating

There is no report generating such as monthly sales report for hotel to keep track on the reservation. Besides, the server also does not provide logging for maintenance purpose.

7.3.5 No Encryption on password

The passwords in the database are not encrypted because the database management system, MySQL does not provide password revision. It will create security issue if there is untruth employee in the back end.

7.4 Future Enhancements

7.4.1 Fully Integration on Back End

There is a possible to fully integrate the back end system if the system designs have compromised. The successful of the integration will reduce the waste of system resource and increase the performance of the back end system.

7.4.2 Increase Compatibility on other Web Browsers

To increase the number of user, the web page can be design to suit in other browsers such as Netscape Navigators.

7.4.3 Report Generating and System Logging

It is important for a company to keep track their sales time to time for analyzing to enhance the services. Therefore, it is a need to have report generating for the hotel on reservation records. The system logging also important to an administrator to monitor the performance of a system and troubleshoot when problems encountered. Thus, the servlets can use the Java's I/O API to create a log file for logging.

7.4.4 Password Encryption

The password encryption has to be implemented to increase the security and privacy of the system. Since the MySQL does not provide revision on the password, the programming language API is recommended to overcome this lack for instances, Java Cryptography Extension (JCE), Java Authentication and Authorization Service (JAAS) and Java Security Tools.

7.4.5 Add more relevant Features

There are more relevant features can be added in the TGoP-HRRS to increase the value of system. For example, the reservation form may collect more data such Client arrive time and airplane number so that the hotel may provide transport to fetch their customer.

7.5 Conclusion

The Travel Guide on Palm (TGoP) can be claimed as a pioneer to promote Malaysia tourism industry via the wireless operation on Palm OS device. The Hotel Room Reservation System (HRRS) as the sub-component of the TGoP is to top up the value of the system by providing transaction over the air on Palm OS device. Although there are a lot of similar system in North America (where Palm.net services coverage), but it is still can be considered as a contemporary effort to achieve the goals in the context of Malaysia. Overall, the TGoP-HRRS has achieved and fulfilled the objectives and requirements as determined during analysis phase.

The TGoP-HRRS is a contemporary e-Commerce solution. This system not only offers the front-end system that provides room reservation service but also supplies the back end system that provides managing, monitoring and administrating features. The full strength of the system in reliability, security, flexibility and extensibility might lead this system become a competitor to the similar business.

In the process of developing the system, I have gained invaluable knowledge and experience of the wireless network, network security and programming language. I also

realized that the development schedule is very important in order to deliver the system on time. The knowledge and experience gained through out the system development has equipped me in the future system development.

Finally, I believe that the successful development of TGoP-HRRS is the first step towards more comprehensive and inventive system development in future.

Appendix A - User Manual

1) Server Manual

a) Server Requirements

i. Pentium II or later with at least 64MB RAM;
ii. Web Server – Jakarta-Tomcat 3.2.3;
iii. Operating System – MS Windows 2000 professional;
iv. Database management system – MySQL 3.23;
v. MySQL JDBC driver;
vi. Java runtime environment (JRE 1.1 or later);
vii. Java Development Kit 1.3.1 Standard Edition;
viii. Secure Socket Layer (SSL) – Java Secure Socket Extension (JSSE 1.0.2).
ix. JavaMail 1.2

b) Server Software Installation Guide

There are a few software have to install on the server:
i. Java Development Kit 1.3.1 (JDK 1.3.1);
ii. Tomcat 3.2.3 web server;
iii. MySQL 3.23 database;
iv. MySQL JDBC driver;
v. Java Secure Socket Extension (JSSE 1.0.2);
vi. JavaMail 1.2.

i) Java Development Kit 1.3.1 Installation Guide

1. Download *Java Development Kit 1.3.1* (JDK 1.3.1) from the Java Official web site, http://java.sun.com.
2. Install JDK 1.3.1 by following its installation instructions. JDK 1.3.1 will be installed into c:/jdk1.3.1 by default.

3. Set a new environment variable (JAVA_HOME) to point to the root directory of the JDK. For instance, JAVA_HOME = c:/jdk1.3.1.

ii) Tomcat 3.2.3 Installation Guide

1. Download the Tomcat 3.2.3 from The Jakarta Project web site at http://jakarta.apache.org.
2. Install Tomcat 3.2.3 by following its installation instructions. Tomcat 3.2.3 will be installed into c:/tomcat3.2.3 by default.
3. Set a new environment variable (TOMCAT_HOME) to point to the root directory of the Tomcat. For instance, TOMCAT_HOME = c:/tomcat3.2.3
4. Open the server.xml under the c:/tomcat3.2.3/conf folder.
5. Add the following entries into server.xml within the ContetManager tag:

```
<Context path="/hrrs"
        docBase="c:/integrate/palmadmin/hrrs/"
        debug="0" reloadable="true" />
<Context path="/palm"
        docBase="c:/integrate/palm"
        debug="0" reloadable="true" />
<Context path="/images"
        docBase="c:/integrate/palmadmin/hrrs/images"
        debug="0" reloadable="true" />
```

iii) MySQL Installation Guide

1. Download MySQL 3.23 from http://www.mysql.org/.
2. Install MySQL 3.23 by following its installation instructions. MySQL 3.23 will be installed into c:/mysql by default.
3. Delete the c:/mysql/data folder.
4. Insert the TGoP Installation Disk into Drive A:
5. Copy the data folder from Drive A: to the c:/mysql to replace the folder deleted in step 3.

iv) MySQL JDBC Driver Installation Guide

1. Download the mm.mysql.jdbc-[version].jar from http://www.mysql.org/.
2. Unpack the file to an appropriate directory.
3. Set the CLASSPATH as below:

   CLASSPATH=\path\to\mm.mysql.jdbc-[version];%CLASSPATH%

v) JSSE Installation Guide

1. Download JSSE 1.0.2 API from http://java.sun.com/products/jsse/
2. Unzip the jsse-1_0_2.zip files to an appropriate directory.
3. Copy the jsse.jar, jnet.jar and jcert.jar to the JAVA_HOME/jre/lib/ext or any CLASSPATH.
4. Edit JAVA_HOME/jre/lib/security/java.security :

   Add:

   Security.provider.2=com.sun.net.ssl.internal.ssl.Provider
5. Do:

   `keytool -genkey -alias tomcat -keyalg RSA`

   RSA is essential to work with Netscape and IIS. Use "changeit" as password (or add keypass attribute). You can set parameter keystore and keypass if you want to change the default. The .keystore is placed at the c:/document and settings/user/.
6. To use the HTTP with SSL connector in tomcat, verify that below codes are activated in server.xml

```
<Connector
        className="org.apache.tomcat.service.PoolTcpConnector">
<Parameter name="handler"
        value="org.apache.tomcat.service.http.HttpConnectionHandler"/>
<Parameter name="port" value="8443"/>
<Parameter name="socketFactory"
        value="org.apache.tomcat.net.SSLSocketFactory"/>
<Parameter name="keystore" value="/tomcat/conf/keystore" />
<Parameter name="keypass" value="changeit"/>
<Parameter name="clientAuth" value="true"/>
</Connector>
```

In this example we indicate the keystore is file /tomcat/conf/keystore. The keystore password is "changeit" and we want clients to authenticate.

vi) JavaMail Installation Guide

1. Download JavaMail 1.2 API from http://java.sun.com/products/javamail/.
2. Unzip the javamail-1_2.zip file to a directory and copy the mail.jar to the JAVA_HOME/jre/lib/ext or any CLASSPATH.
3. Download JavaBeans Activation Framework from http://java.sun.com/products/javabeans/glasgow/jaf.html.
4. Unzip the jaf-1_0_1.zip file and add the activation.jar to the JAVA_HOME/jre/lib/ext or any CLASSPATH.

2) Front End (Client) Manual

a) Client System Requirements

i. Palm OS device with at least Palm OS 3.2 or later;
ii. Web Clipper which resident on the Palm OS device;
iii. Palm VII or modem adapted Palm OS device for wireless connection;
iv. PC with MS Windows 95 or later;
v. Web browser – Internet Explorer 5.0 or later;
vi. TGoP Palm Query Application (PQA) which may download from the TGoP portal.

b) PQA Installation Guide

i. Download tgop.pqa from TGoP portal;
ii. Install it using your HotSync via the cradle.

c) Getting Started

i. If you wish to use the HRRS services, you are required to sign up as TGoP member.

ii. To Sign Up as member, you may go to the TGoP portal using the web browser and click on the Sign Up link (please refer to Figure A-1).

iii. Fill up all the fields in the registration form (Figure A-2) and click "Sign Up" button to submit the form or click "Cancel" to cancel the registration.

iv. After signing up complete, you may sign in and enjoy the TGoP services.

v. Download the tgop.pqa and install it onto your Palm to enjoy our services on the move.

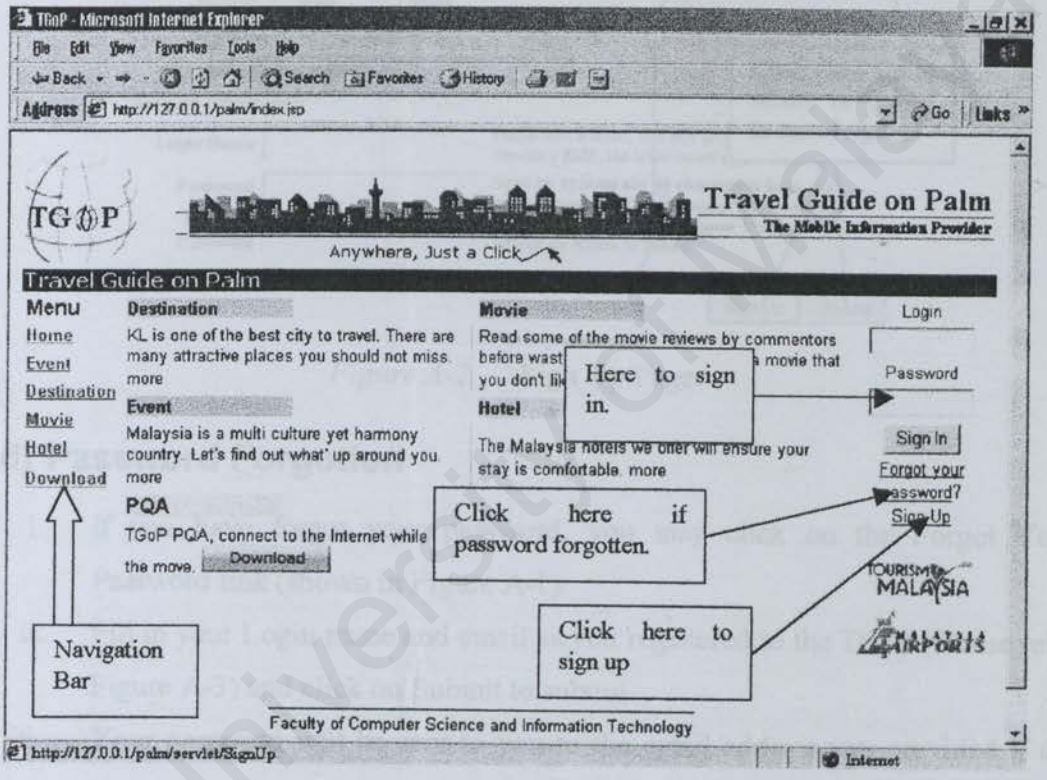

Figure A-1 TGoP main page.

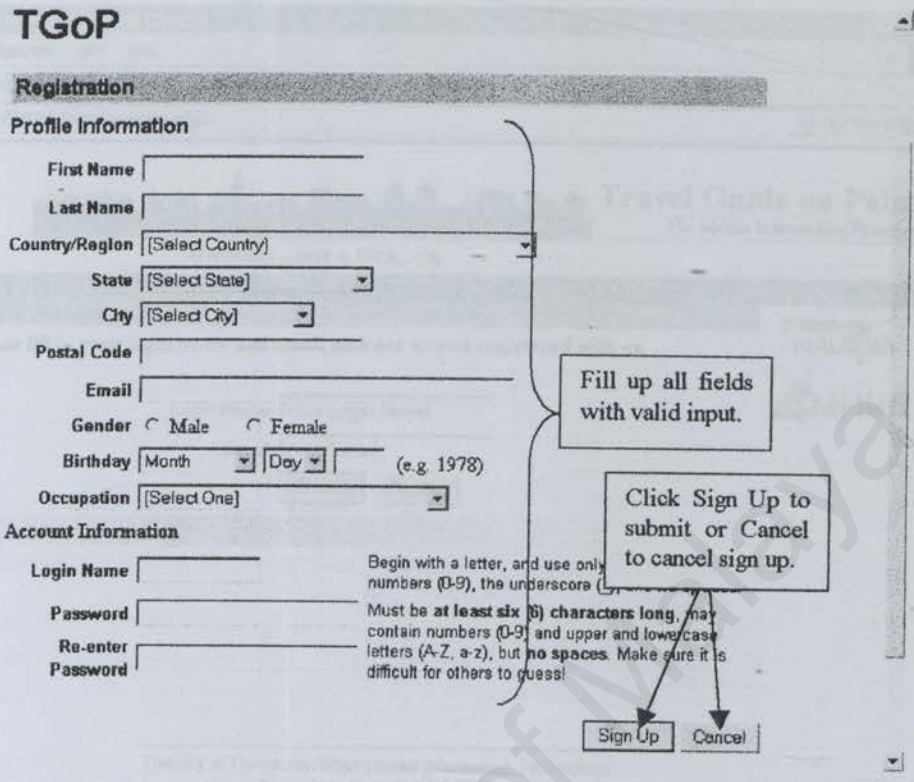

Figure A-2 Sign Up Page.

d) Password Forgotten

i. If you have forgot your password, you may click on the Forgot Your Password link (shown in Figure A-1).

ii. Fill in your Login name and email as you registered to the TGoP (Please refer Figure A-3) and click on Submit to submit.

iii. Your password will be sent to you to the email address you provided if the given address is match to the address in the database.

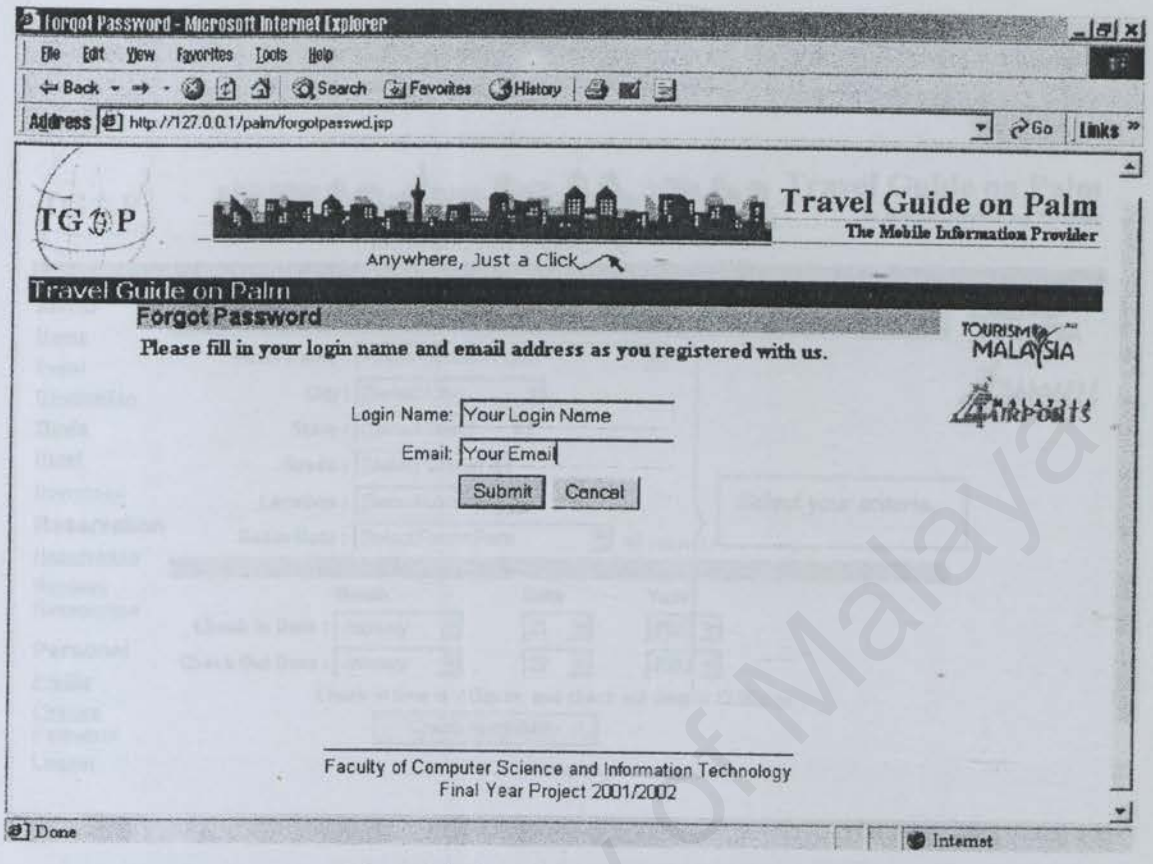

Figure A-3 Password Forgotten.

e) Making a Reservation

i. To do a reservation, click on the Hotel link (if you are not signing in) or Reservation link (if you have signed in) in the navigation bar.

ii. Select your criteria and click on Check Availability to find rooms that match your criteria (Please refer to Figure A-4).

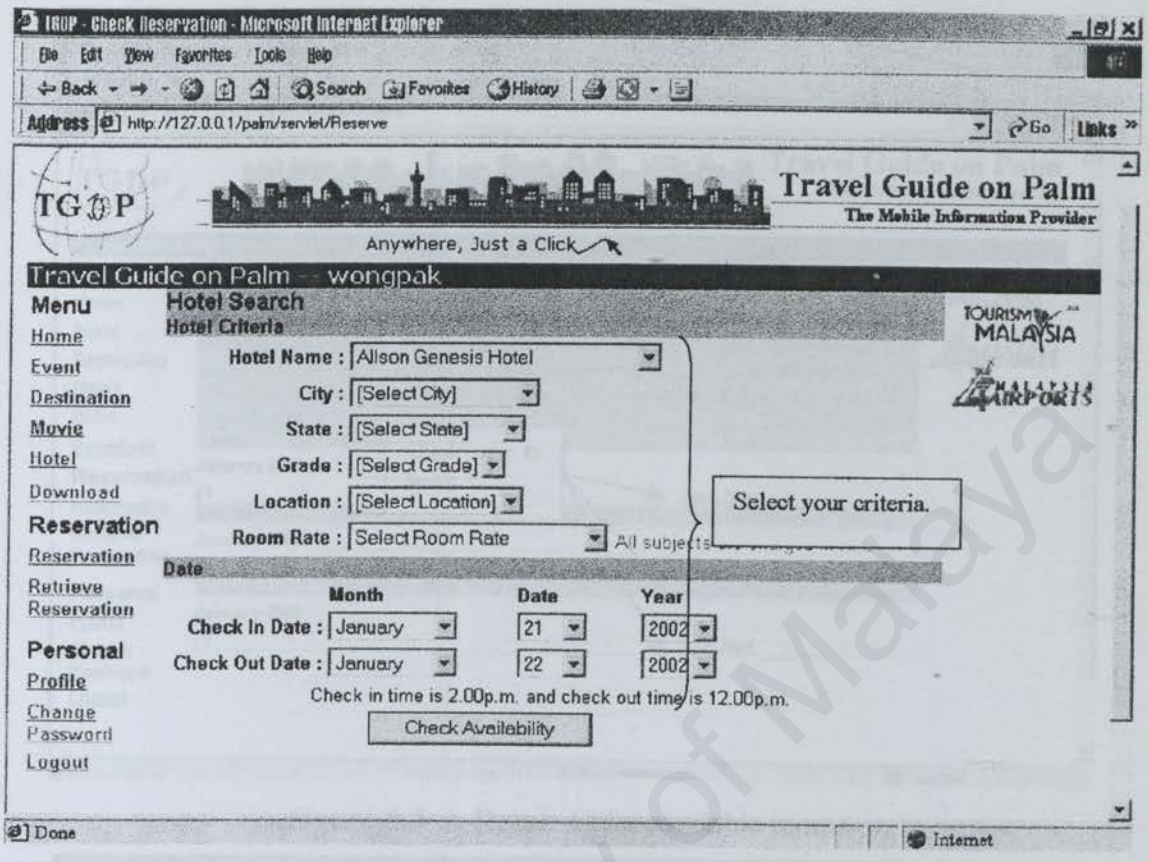

Figure A-4 Find available rooms.

iii. Click Book Now to book the room in the result set that returns (Figure A-5). (You may asked to sign in if you are not signing in)

iv. Fill up the form in Figure A-6 and click Continue to continue.

v. Complete the form in Figure A-7 and click Book to submit the reservation. Please read on the terms and conditions before submit the reservation.

vi. A email will send to you to inform you that your reservation has been received and in processed.

f) On Palm

Basically the navigation on palm is similar to the navigation on desktop except the navigation bar is at the bottom of the clipping page instead of at the side. More over, you are restricted to update your profile and change your password on Palm.

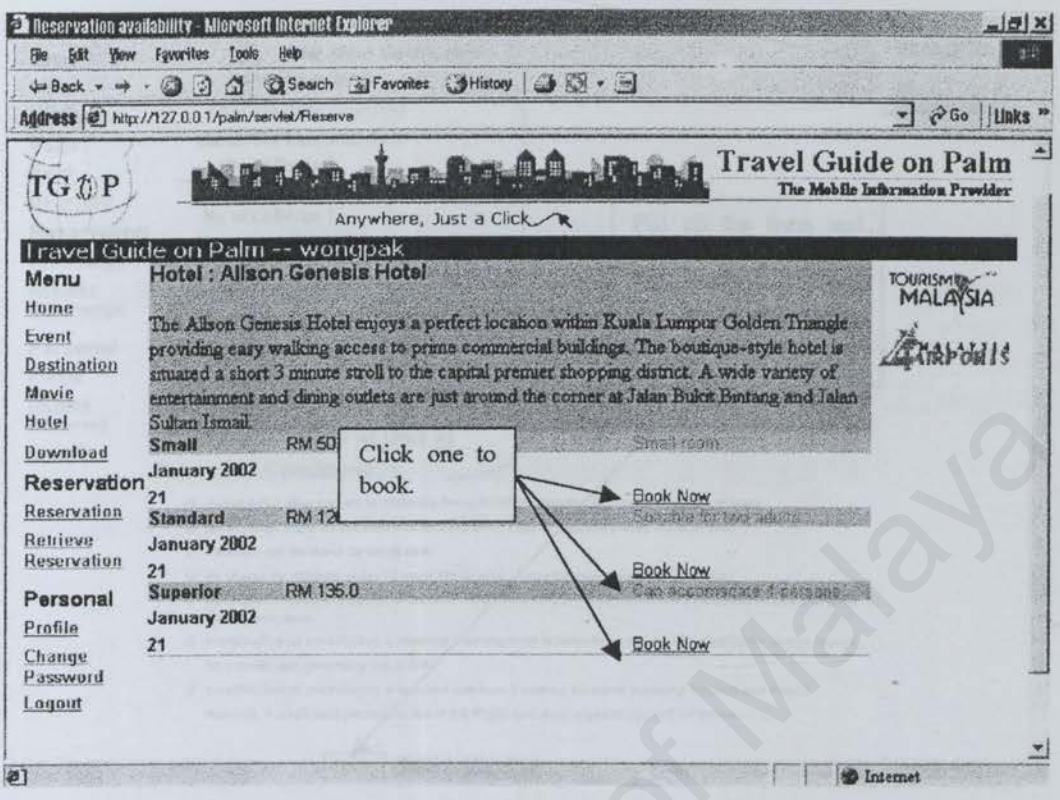

Figure A-5 Result set for available rooms.

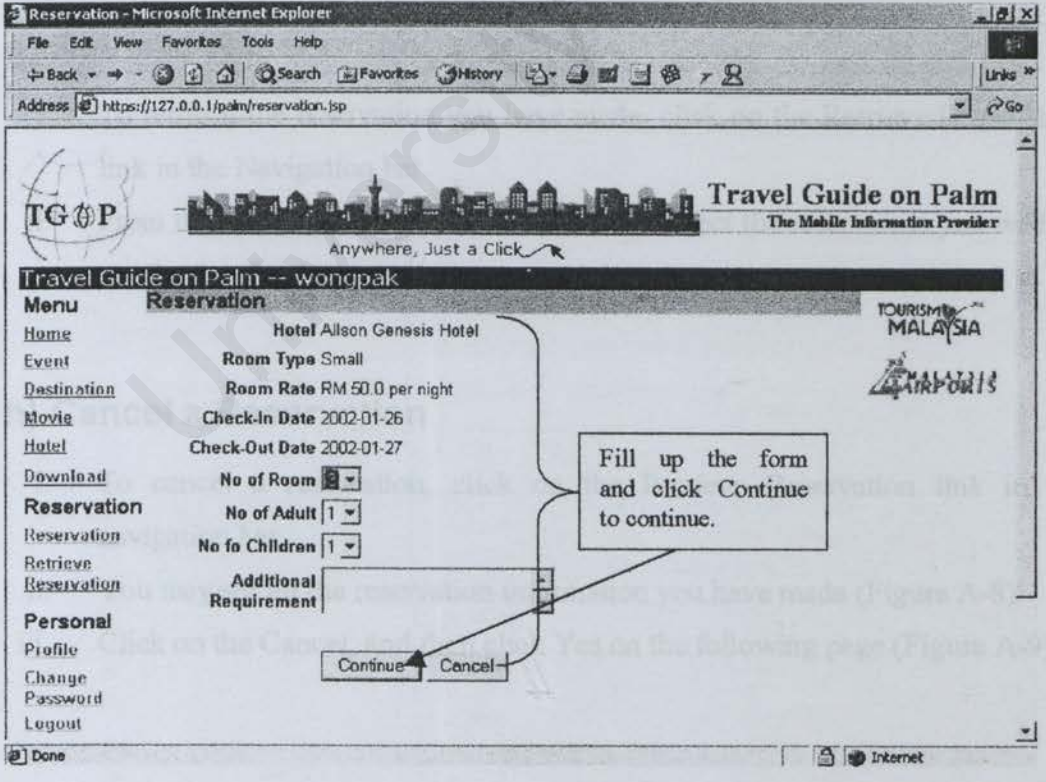

Figure A-6 Reservation Form 1.

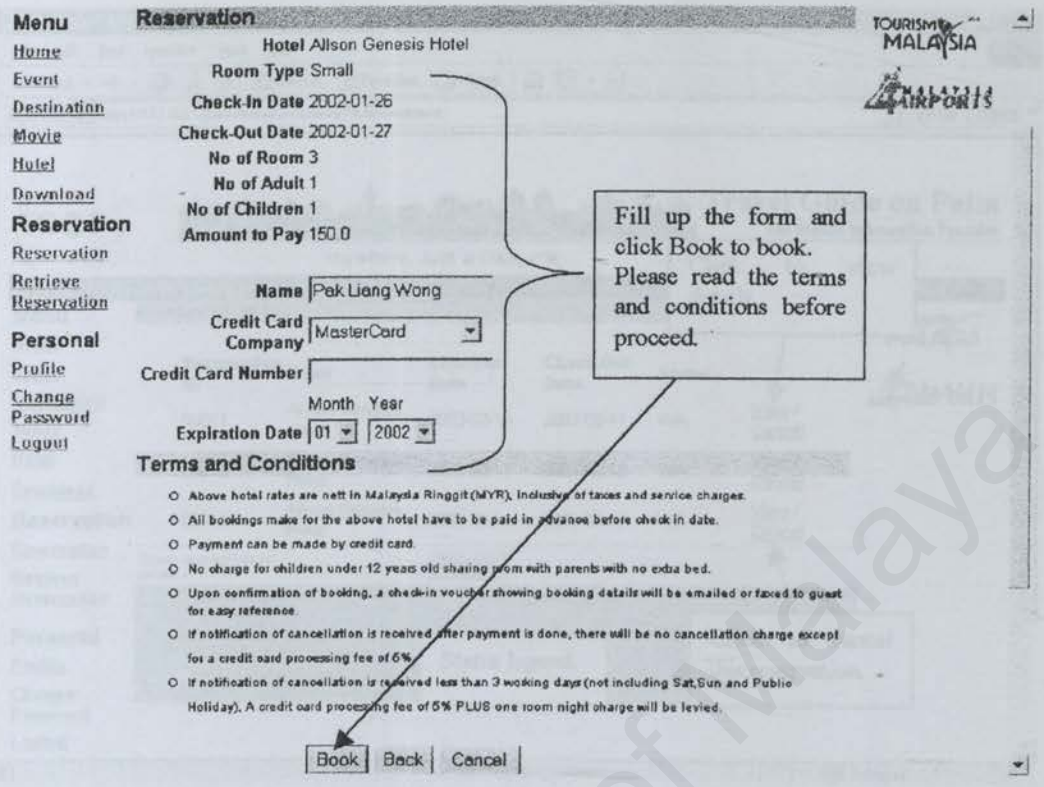

Figure A-7 Reservation Form 2.

g) Retrieve Reservation

i. To retrieve the reservation you have made, click on the Retrieve Reservation link in the Navigation bar.

ii. From the list (Figure A-8), click on view to select the reservation you wish to see the details.

h) Cancel a Reservation

i. To cancel a reservation, click on the Retrieve Reservation link in the navigation bar.

ii. You may see all the reservation information you have made (Figure A-8).

iii. Click on the Cancel, and then click Yes on the following page (Figure A-9).

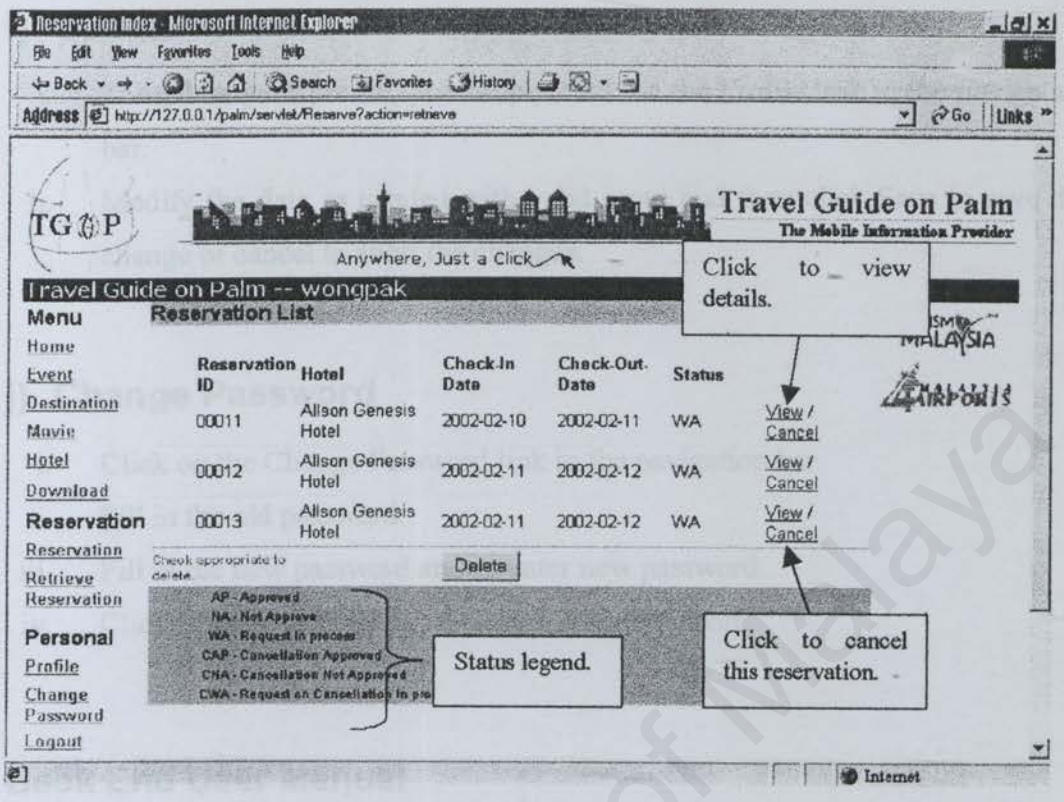

Figure A-8 Reservation List.

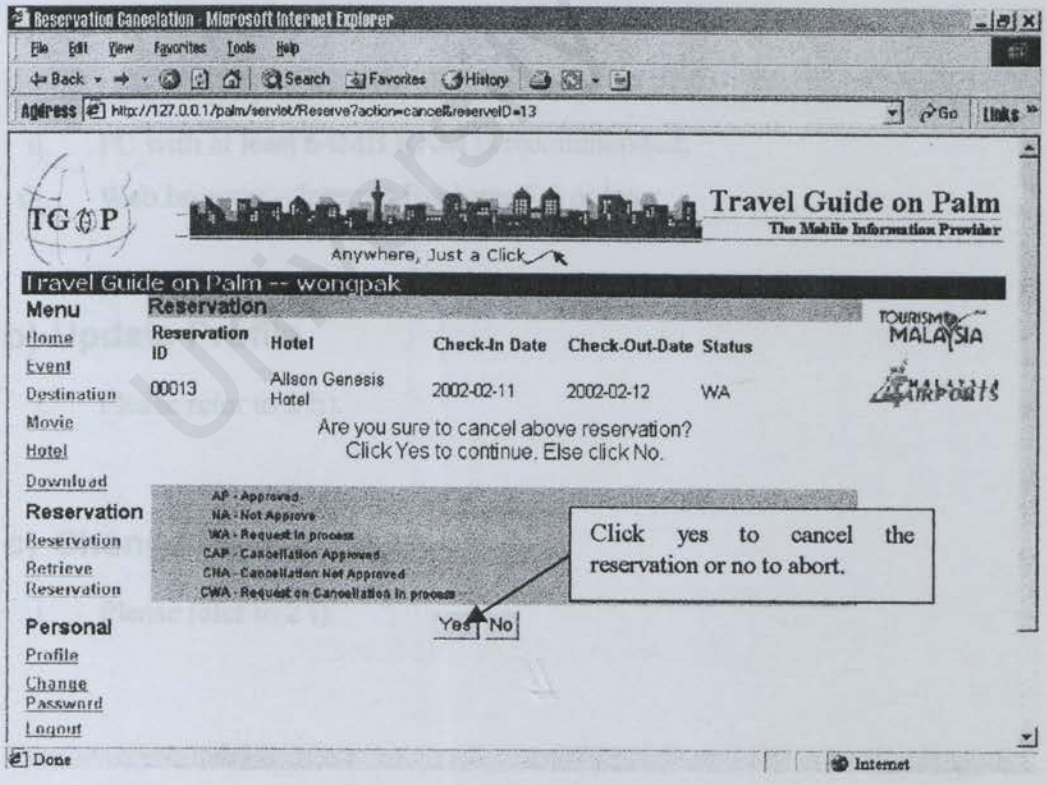

Figure A-9 Cancel a reservation.

i) Update Profile

i. To update your profile, just simply click on the Profile link in the navigation bar.

ii. Modify the data as needed with valid input and then click Save to save the change or cancel to abort the changes.

j) Change Password

i. Click on the Change Password link in the navigation bar.

ii. Fill in the old password.

iii. Fill in the new password and re-enter new password.

iv. Click Change to save the change or cancel to abort.

3) Back End User Manual

a) Back End User System Requirements

i. Operating System – MS Windows 95 or later;

ii. PC with at least 64MB RAM is recommended;

iii. Web browser – Internet Explorer 5.0 or later.

b) Update Profile

i. Please refer to 2 h).

c) Change Password

i. Please refer to 2 i).

d) Super User Manual

i) Login

1. The super user login as *root* with the password *palmroot*. Please change the password after first time login.

ii) Add Portal Admin

1. To add a portal administrator, click on Add Portal Administrator link in the navigation bar (Figure A-10).
2. Fill in the name, login name and password for the new portal admin.
3. Click Add to add the portal admin.

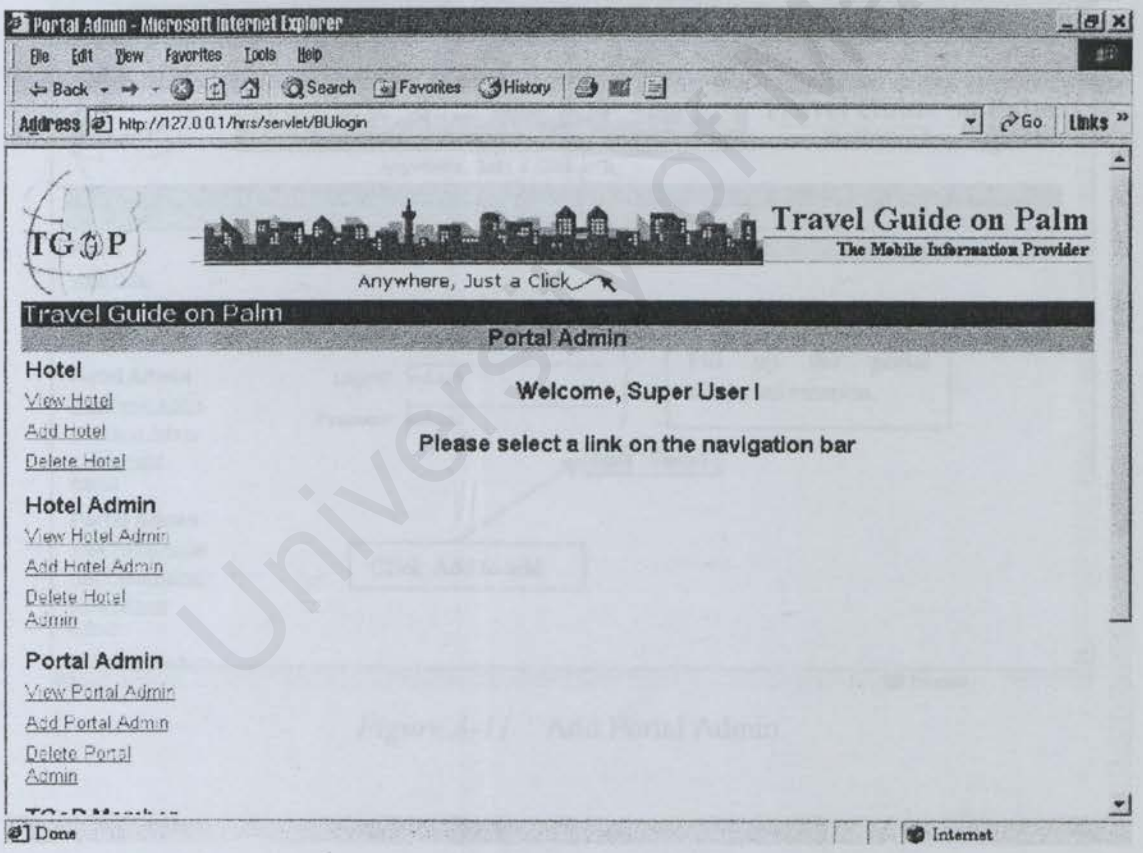

Figure A-10 Super user main page.

iii) View Portal Admin

1. Click on the View Portal Admin link to view portal admin list or click.
2. Click View on the list to view the portal admin details.

iv) Delete Portal Admin

1. Click Delete to select the portal admin to be deleted.
2. Check on the portal admin who will be deleted (Figure A-13).
3. Click Delete and OK on the message box to delete or cancel to abort the deletion.

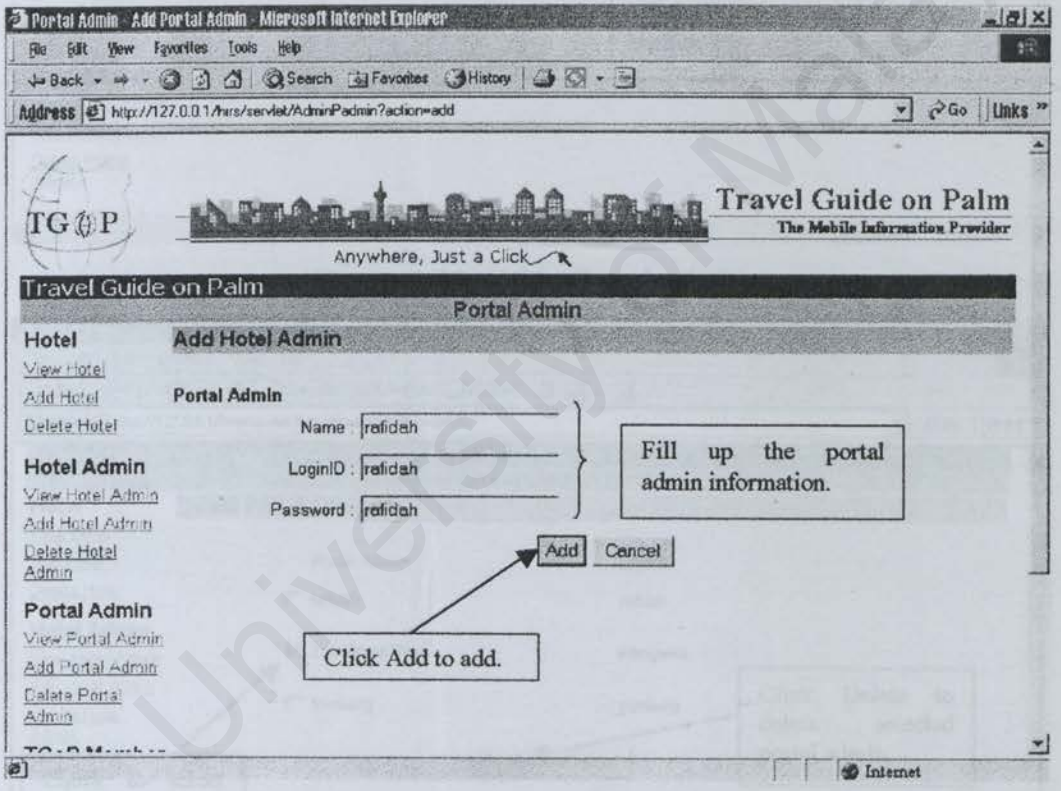

Figure A-11 Add Portal Admin.

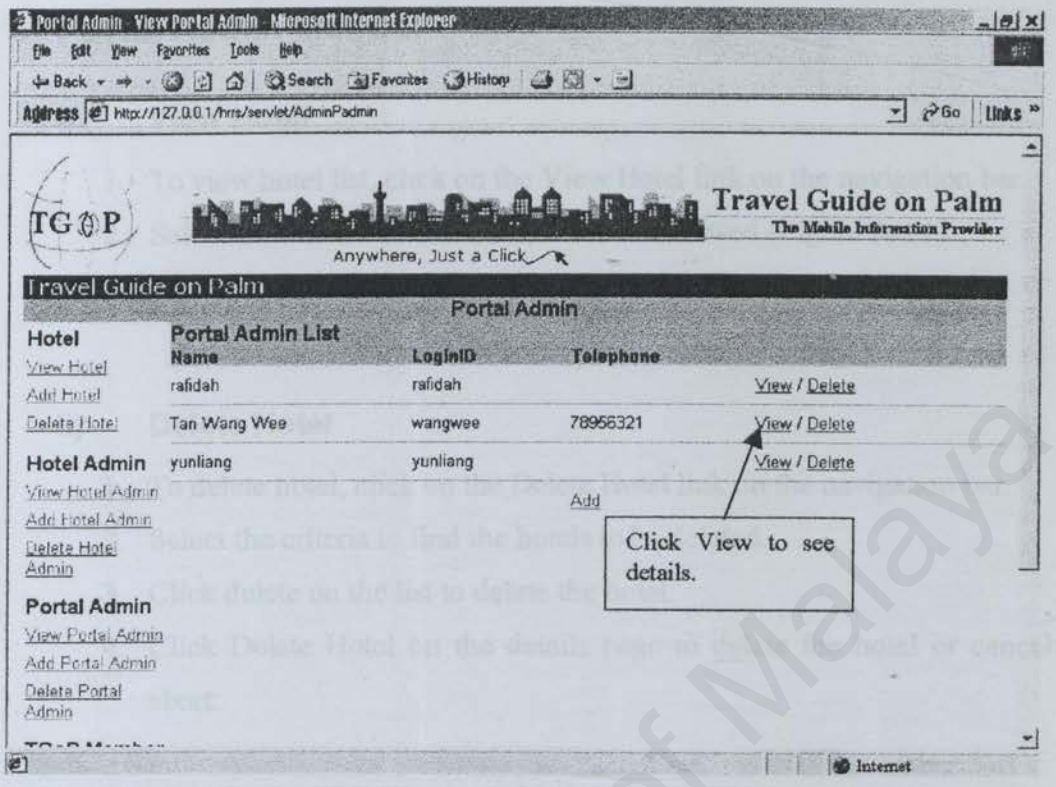

Figure A-12 View Portal Admin

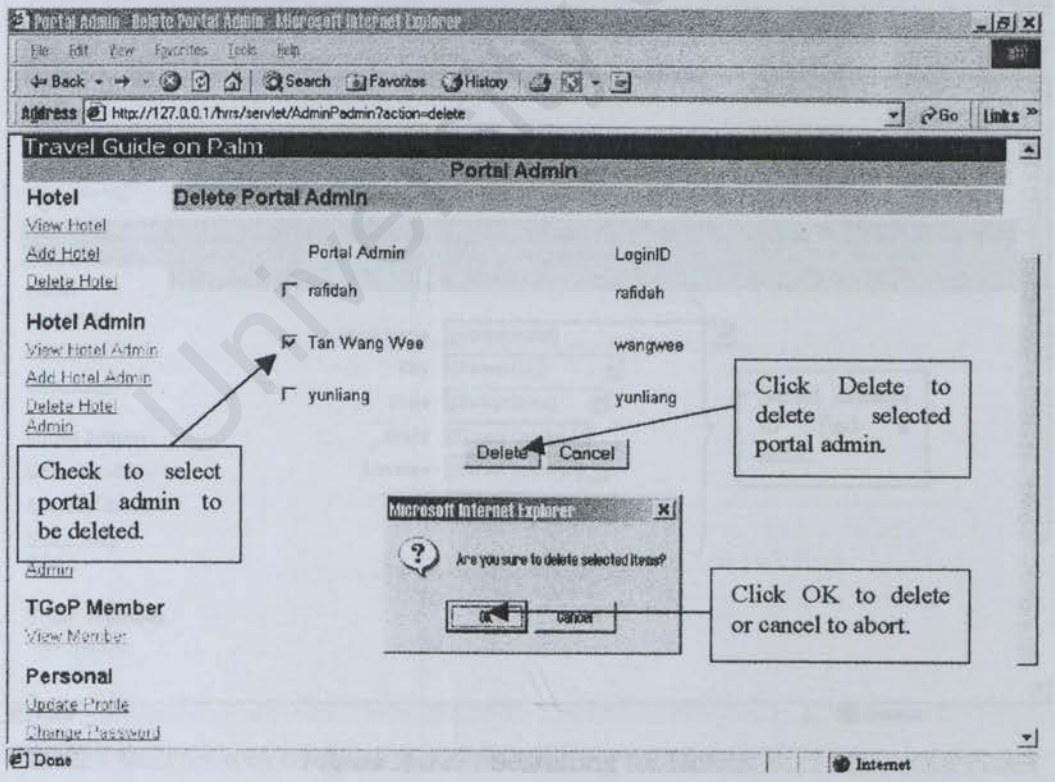

Figure A-13 Delete Portal Admin

e) Portal Admin Manual

i) View Hotel

1. To view hotel list, click on the View Hotel link on the navigation bar.
2. Select the criteria to find the hotels to be viewed (Figure A-14).
3. Click view on the list to view details (Figure A-15).

ii) Delete Hotel

1. To delete hotel, click on the Delete Hotel link on the navigation bar.
2. Select the criteria to find the hotels to be deleted.
3. Click delete on the list to delete the hotel.
4. Click Delete Hotel on the details page to delete the hotel or cancel to abort.

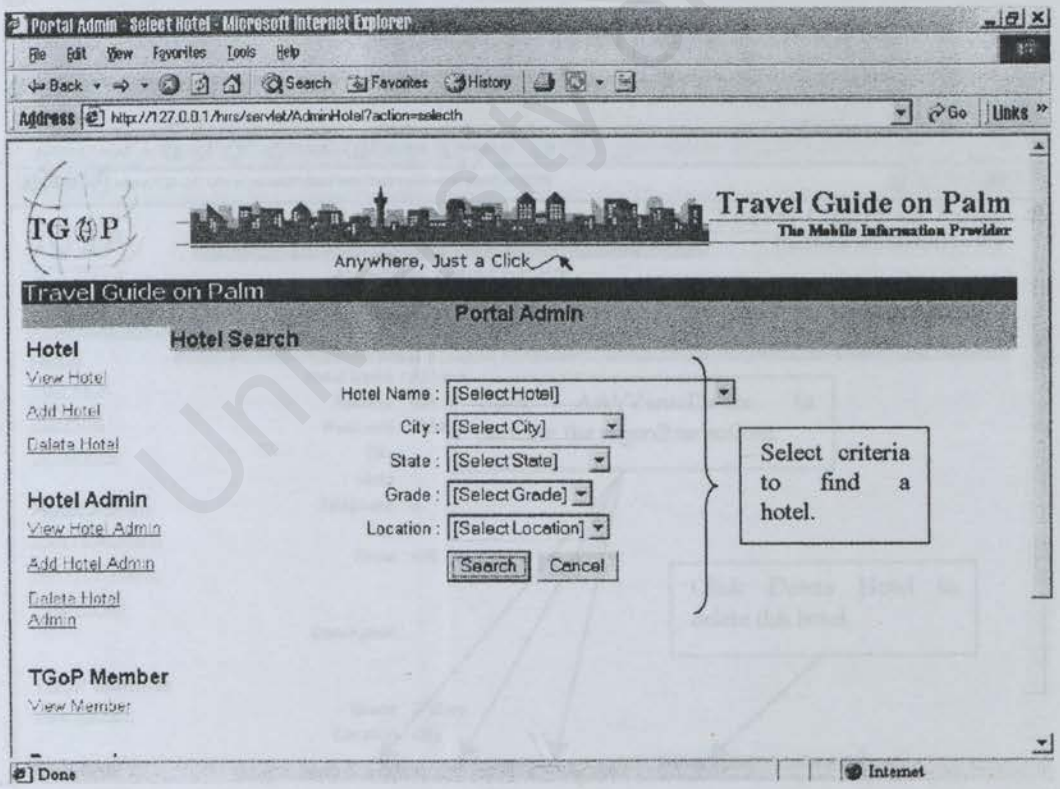

Figure A-14 Searching for Hotels.

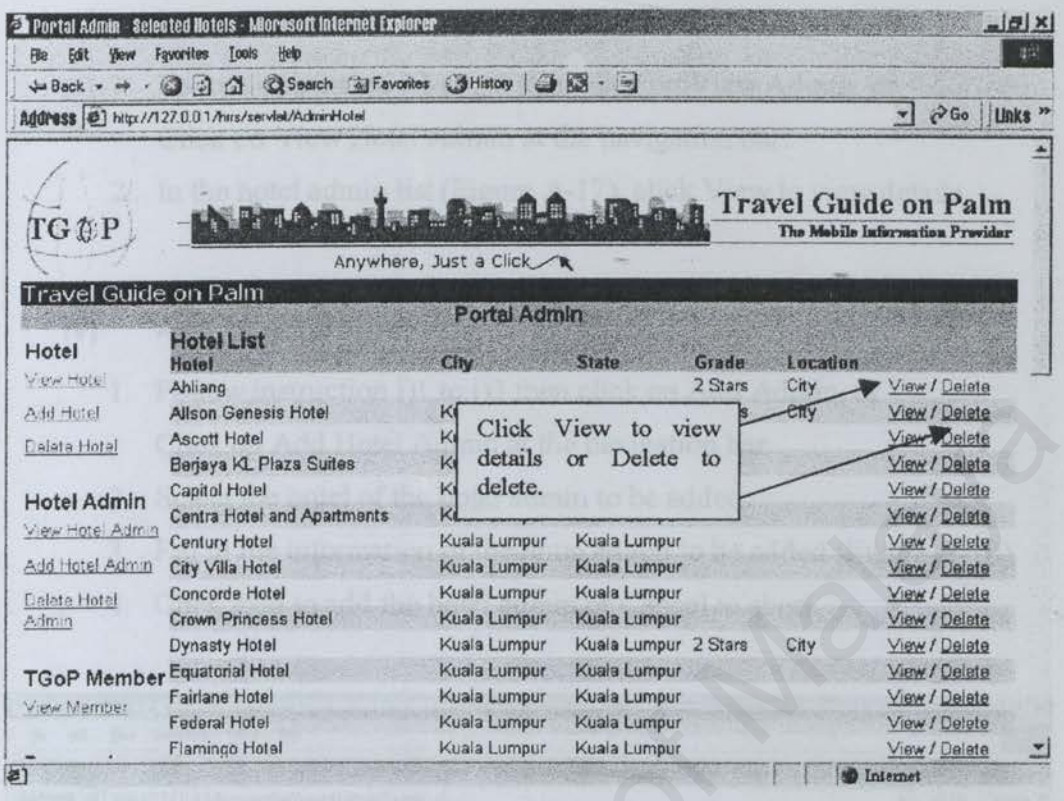

Figure A-15 Hotel List.

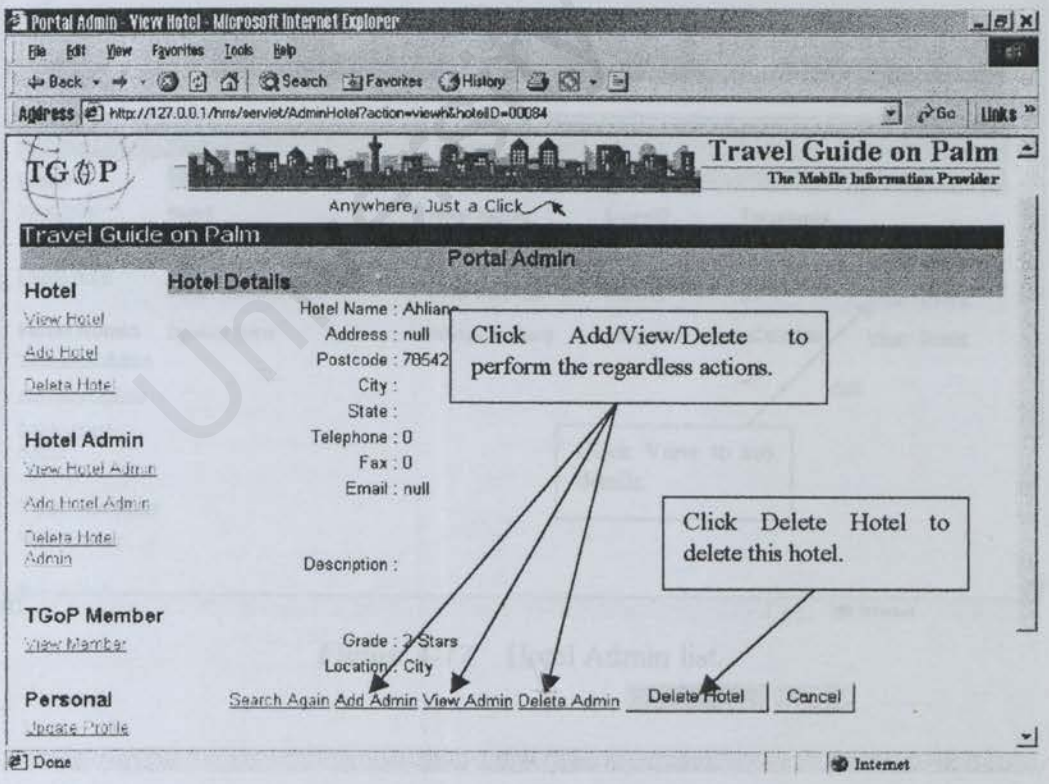

Figure A-16 Hotel details.

iii) View Hotel Admin

1. Follow instruction i)1 to i)3 then click on View Admin, or Click on View Hotel Admin at the navigation bar.
2. In the hotel admin list (Figure A-17), click View to view details.

iv) Add Hotel Admin

1. Follow instruction i)1 to i)3 then click on Add Admin, or Click on Add Hotel Admin at the navigation bar.
2. Select the hotel of the hotel admin to be added.
3. Fill in the information of the hotel admin to be added (Figure A-18).
4. Click Add to add the hotel admin or Cancel to abort.

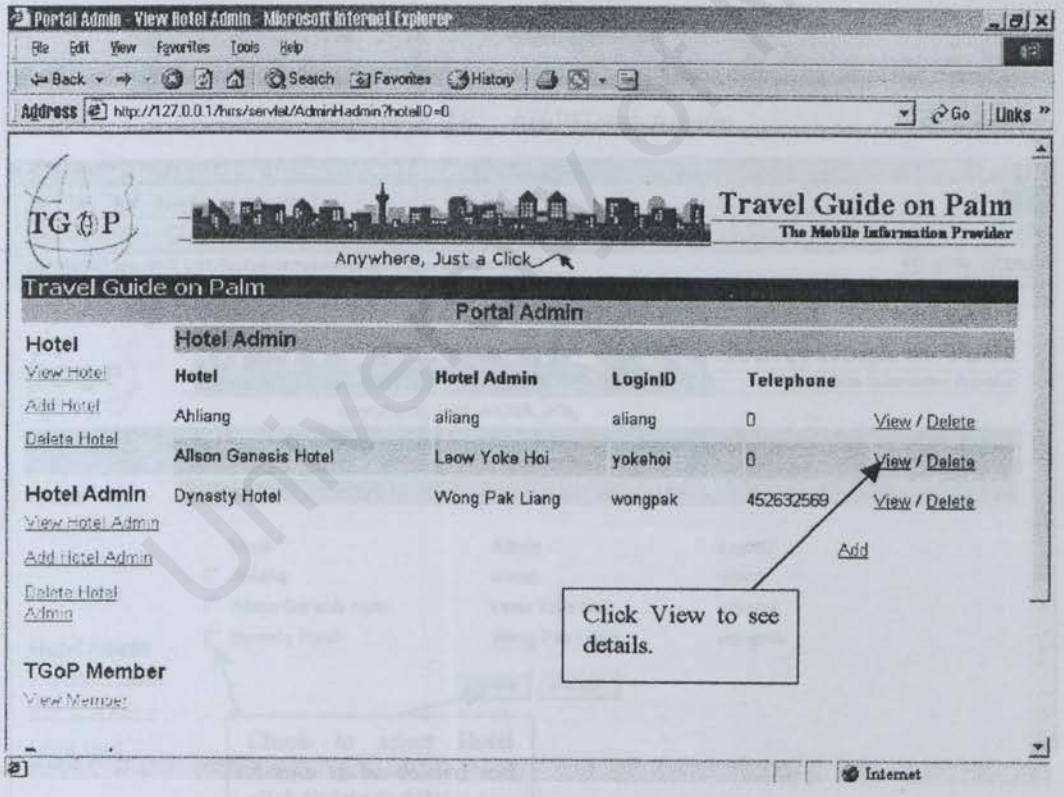

Figure A-17 Hotel Admin list.

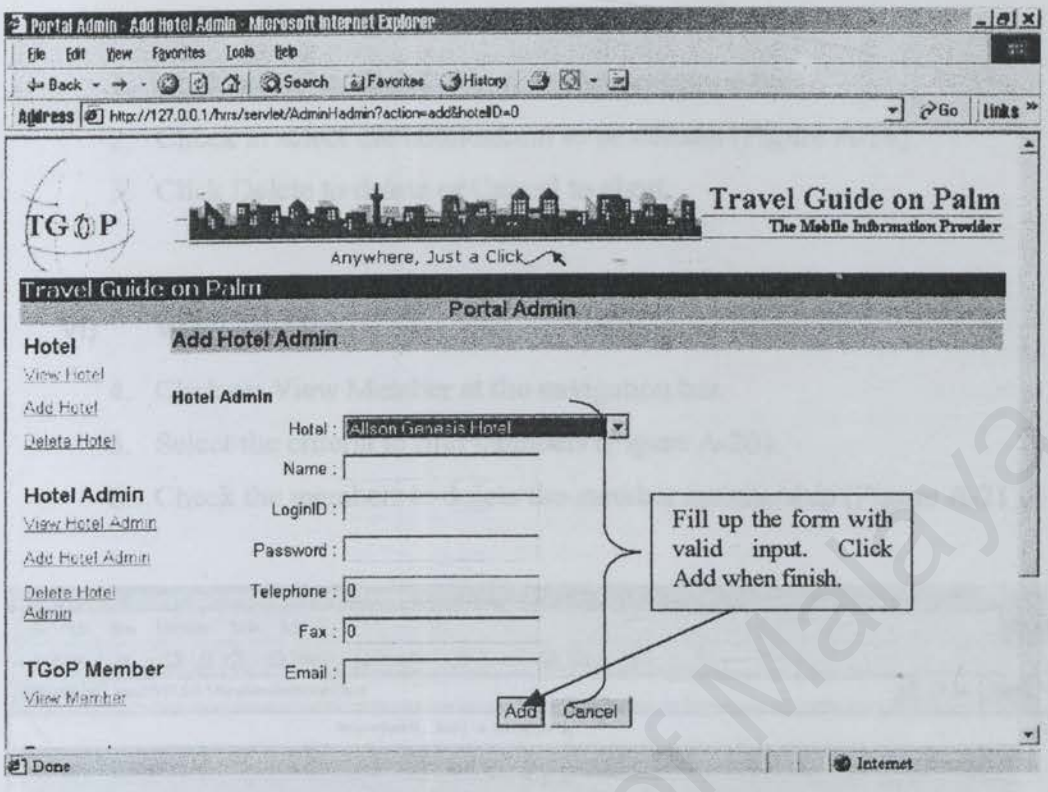

Figure A-18 Add Hotel Admin.

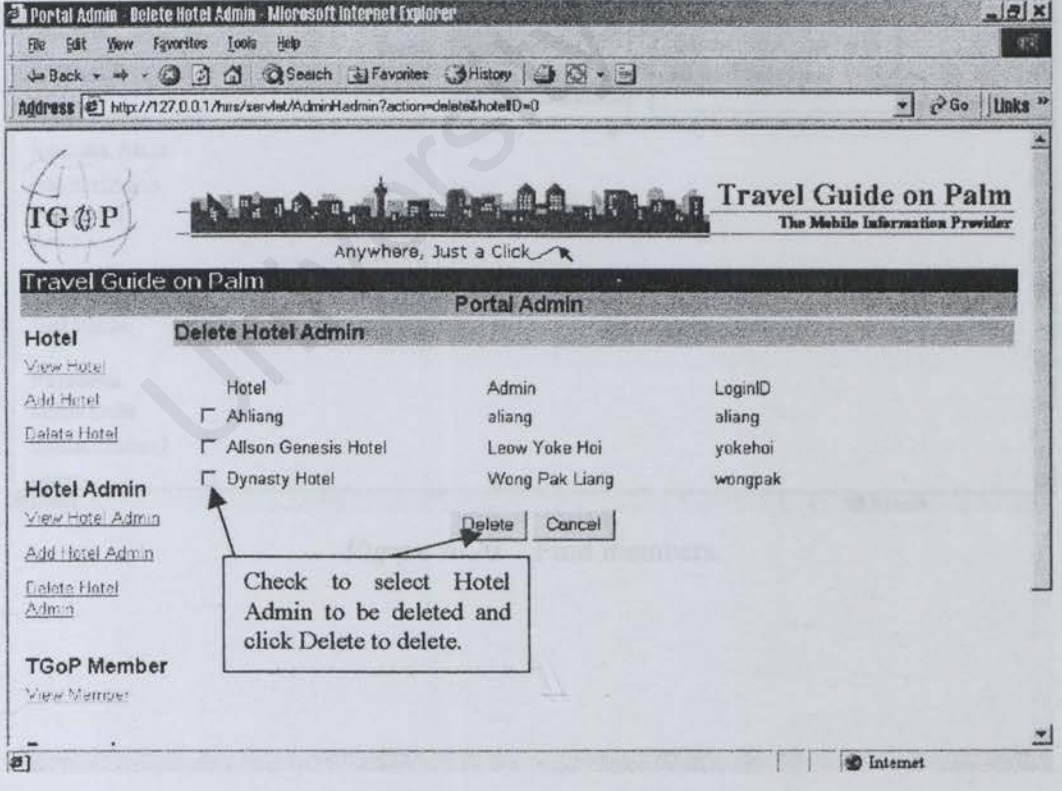

Figure A-19 Delete Hotel Admin.

v) Delete Hotel Admin

1. Click on Delete Hotel Admin at the navigation bar.
2. Check to select the hotel admin to be deleted (Figure A-19).
3. Click Delete to delete or Cancel to abort.

vi) View Member

4. Click on View Member at the navigation bar.
5. Select the criteria to find members (Figure A-20).
6. Check the members to delete the member membership (Figure A-21).

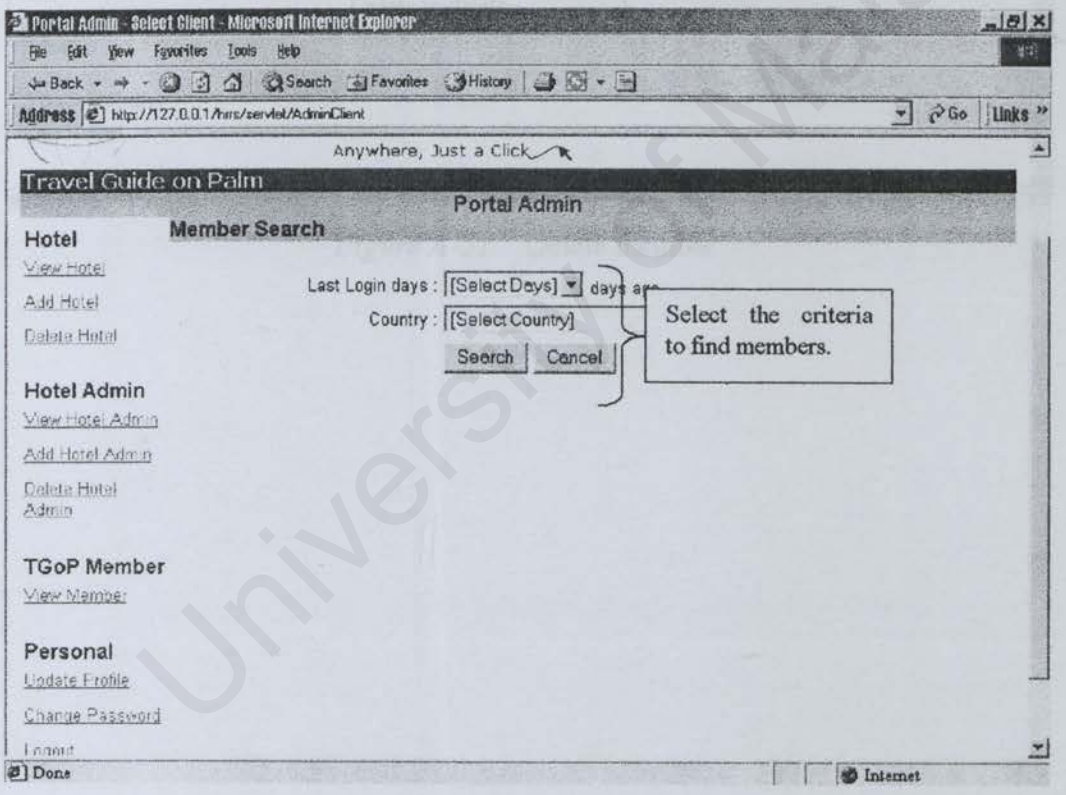

Figure A-20 Find members.

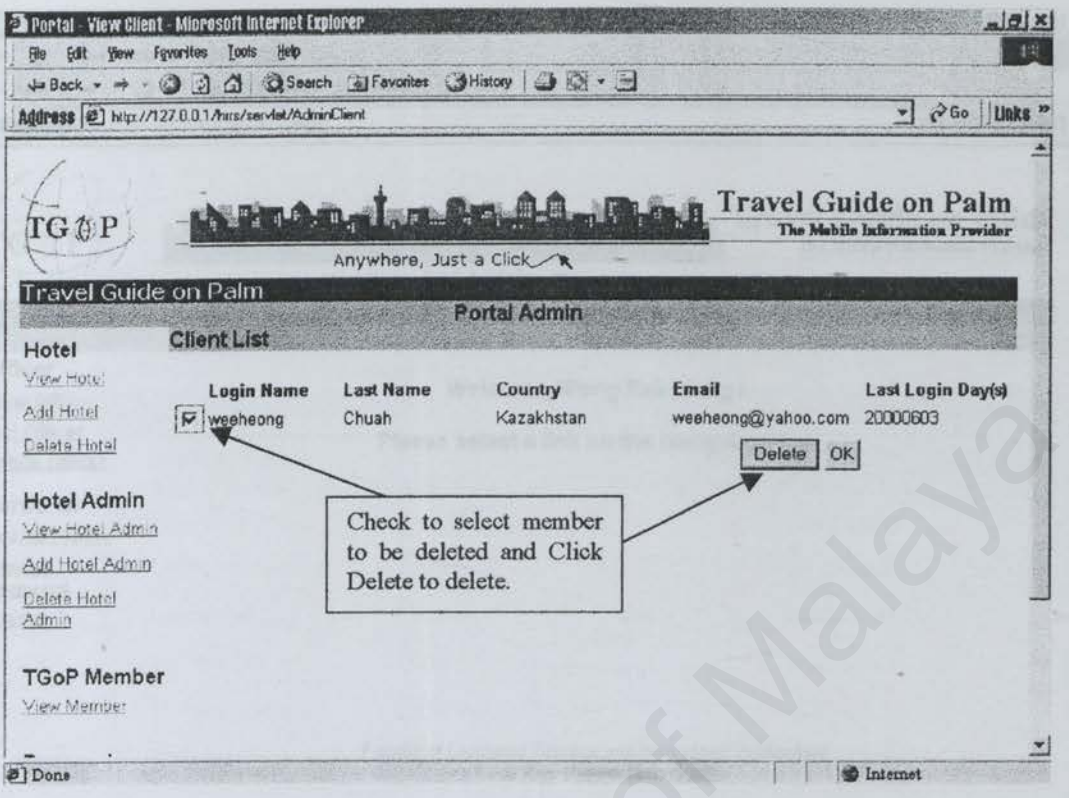

Figure A-21 Delete member.

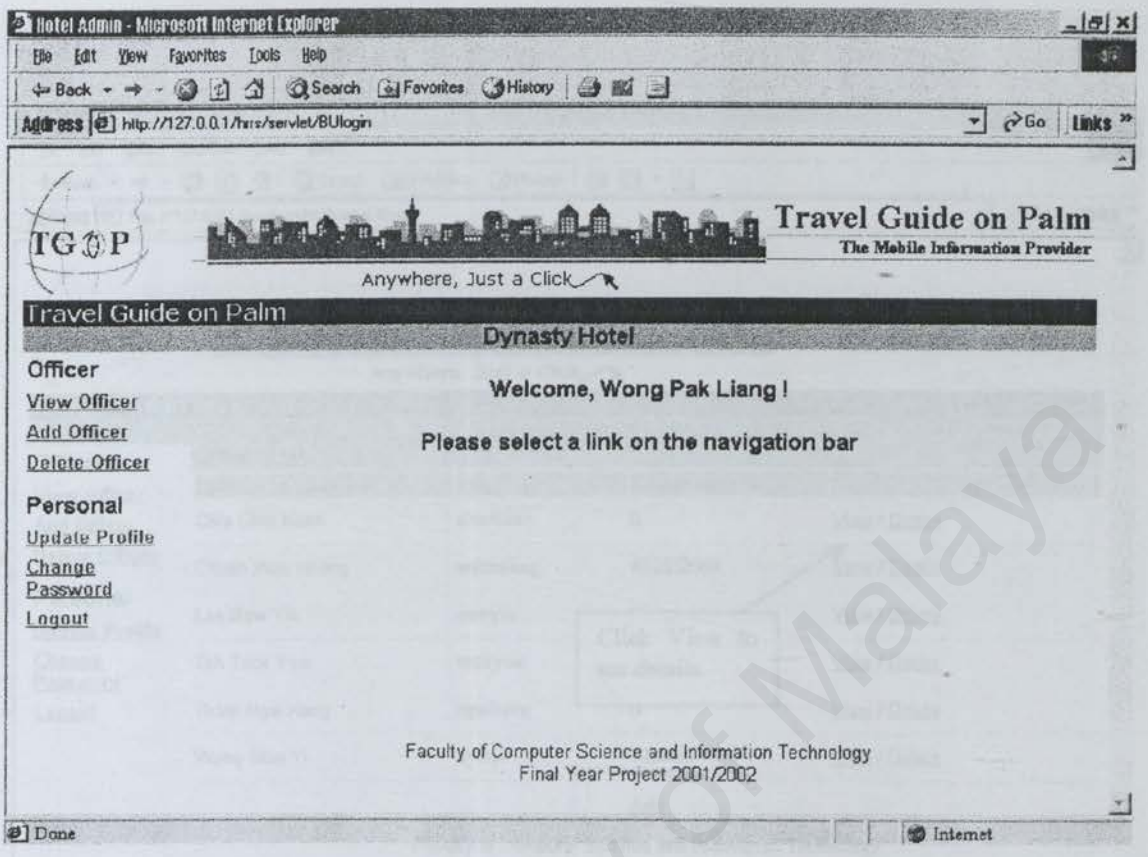

Figure A-22 Hotel Admin main page.

f) Hotel Admin Manual

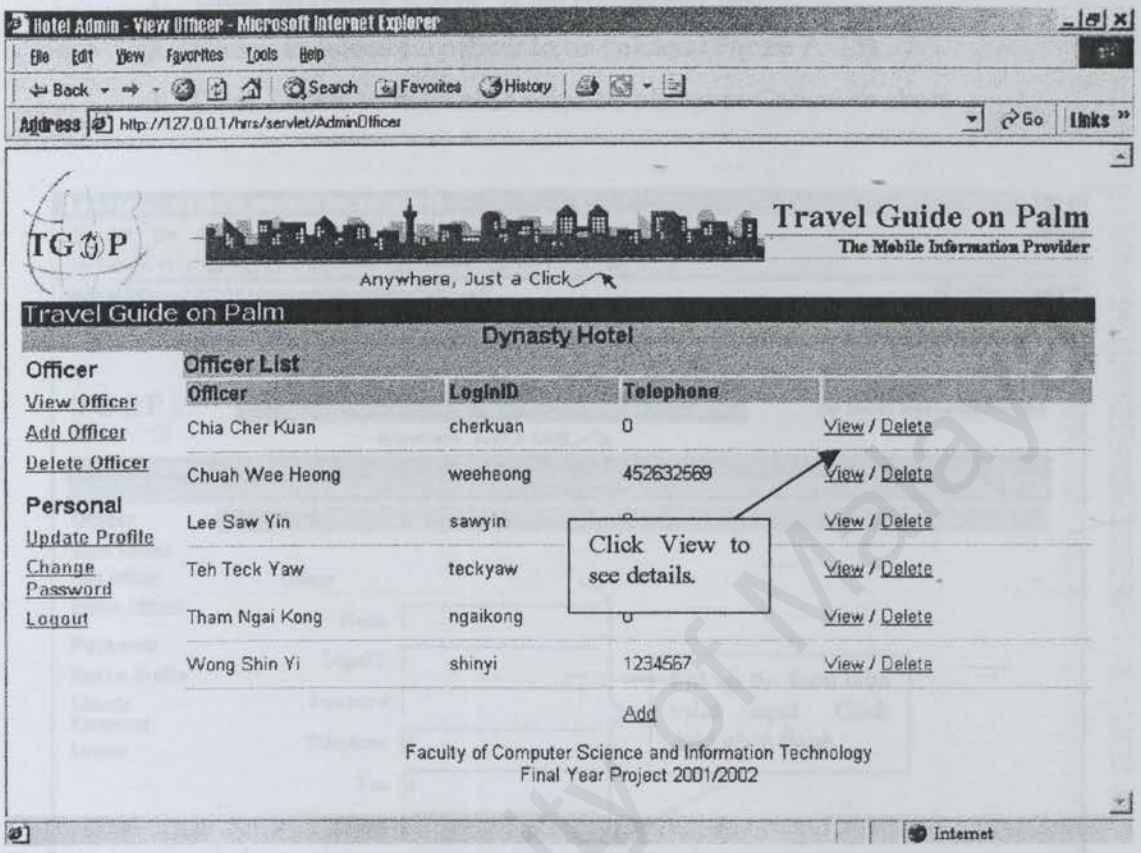

Figure A-23 Officer list.

i) View Officer

1. Click on View Officer at the navigation bar (Figure A-22).
2. Click View to view the details (FigureA-23).

ii) Add Officer

1. Click on Add Officer at the navigation bar.
2. Fill in the form with the Officer name, login name and password.
3. Fill in other details if possible.
4. Click Add to add the officer or Cancel to abort.
5. Click View Officer on the navigation bar.

iii) Delete Officer

1. Click on Delete Officer at the navigation bar.
2. Check to select the officer to be deleted (Figure A-25).
3. Click Delete to delete the selected officer or Cancel to abort.

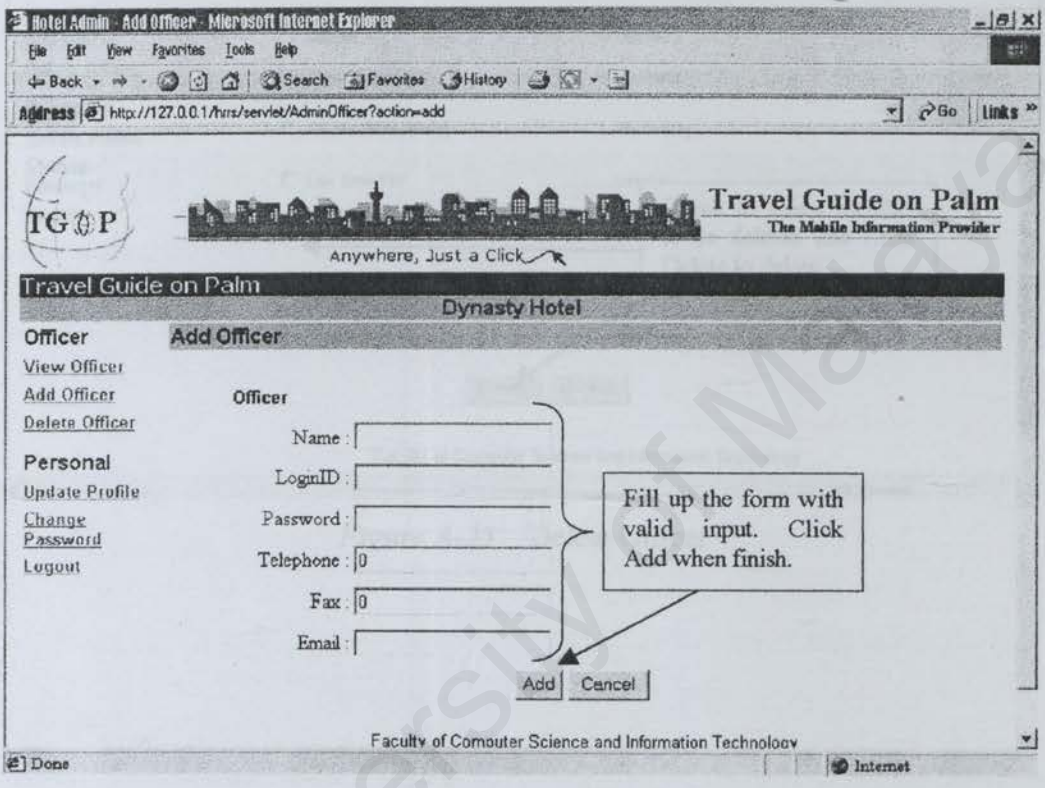

Figure A-24 Add Officer.

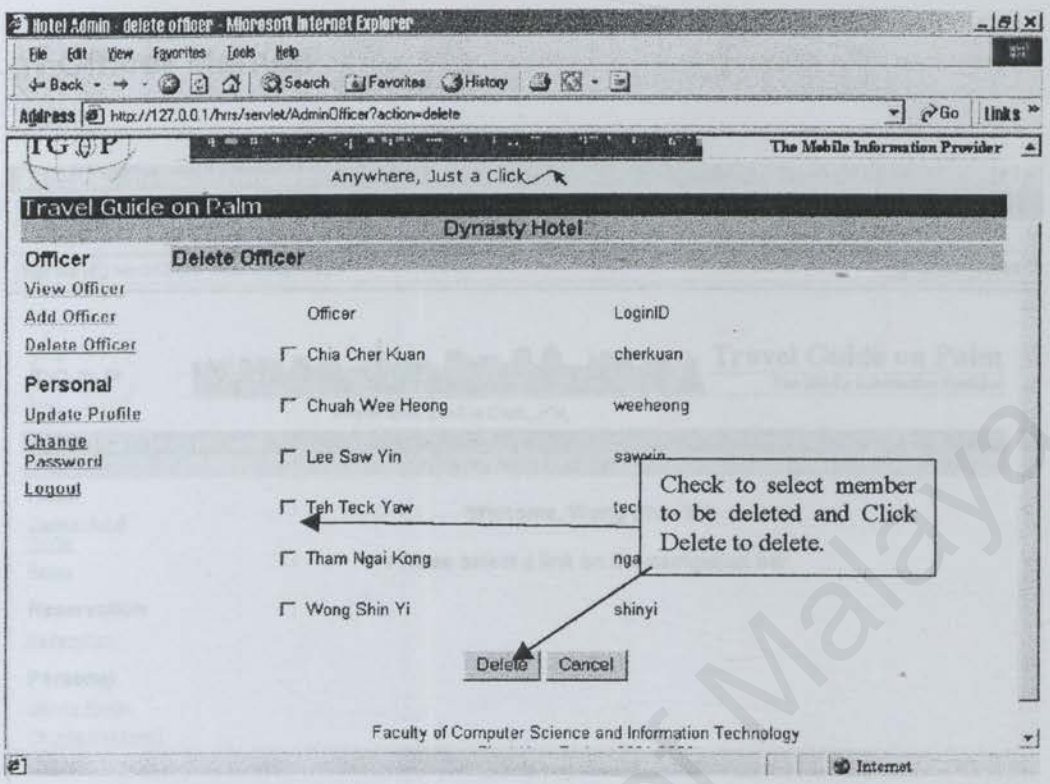

Figure A-25 Delete Officer

g) Officer Manual

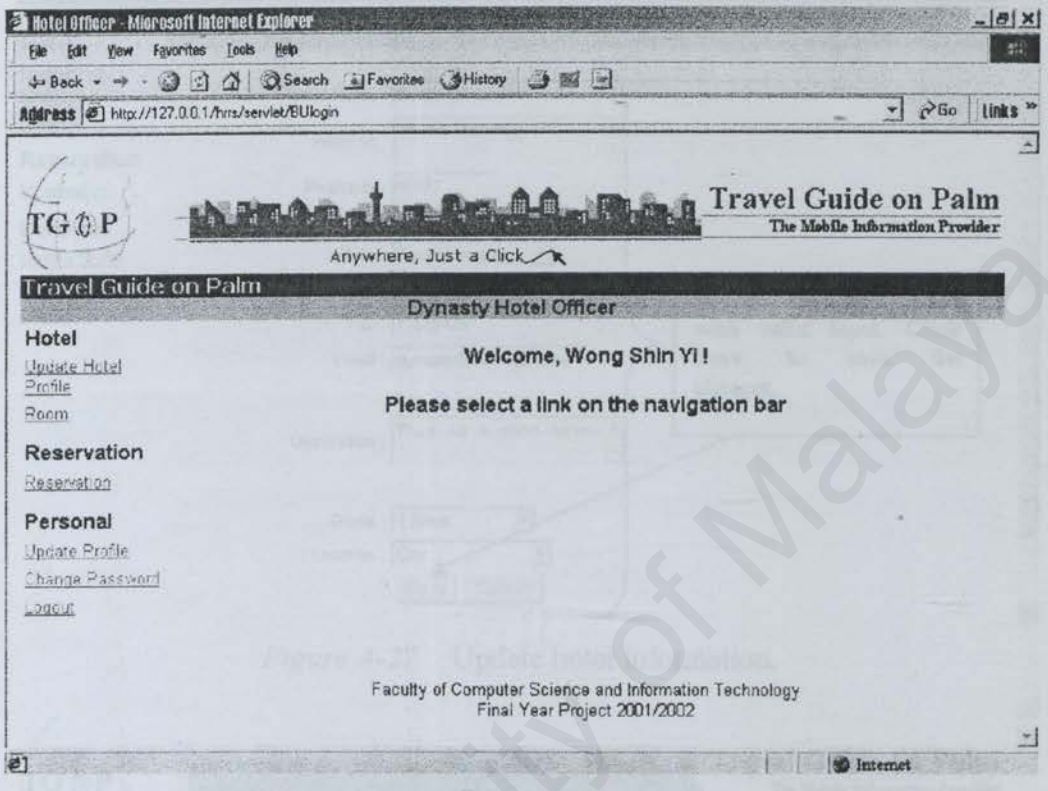

Figure A-26 Officer main page.

i) Update Hotel Profile

1. Click on Update Hotel Profile at the navigation bar (Figure A-26).
2. Modify the hotel information as needed with valid input (Figure A-27).
3. Click Save to save changes or Cancel to abort.

ii) Add New Room Type

1. Click on Room at the navigation bar (Figure A-26).
2. Fill up the fields at the top division with title bar *Add New Room Type* (Figure A-28).
3. Click Add Room to add the room.
4. The new room type will be displayed under the title bar *Room Type List*.

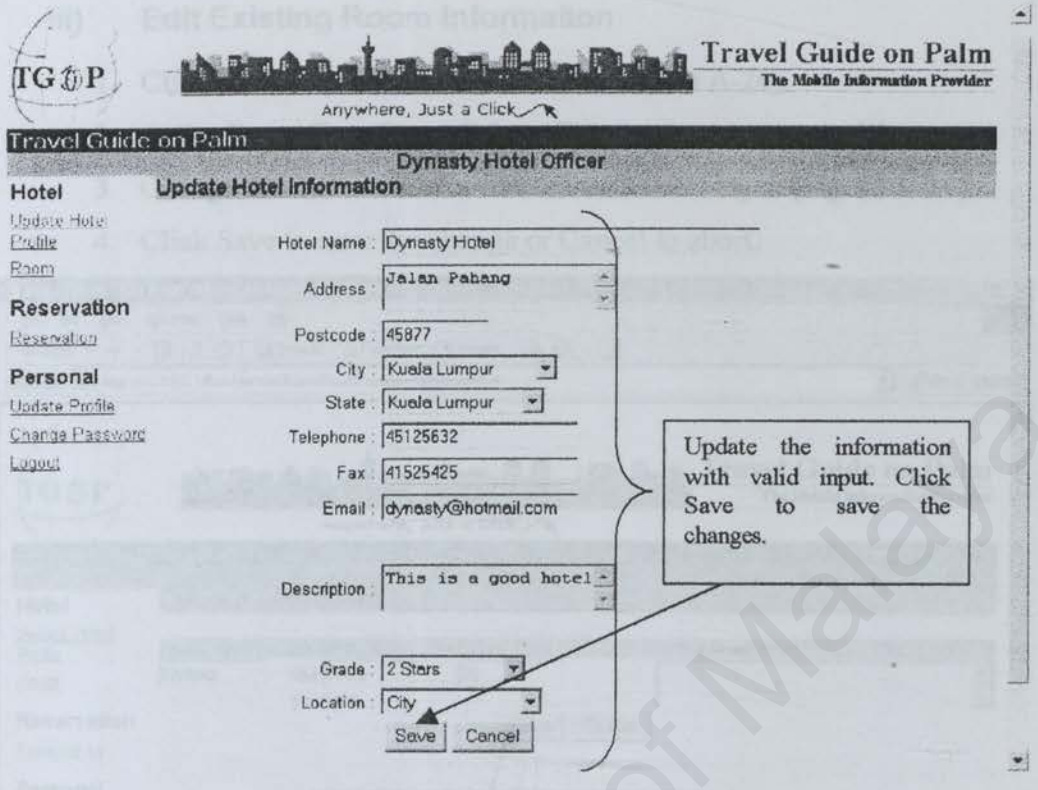

Figure A-27 Update hotel information.

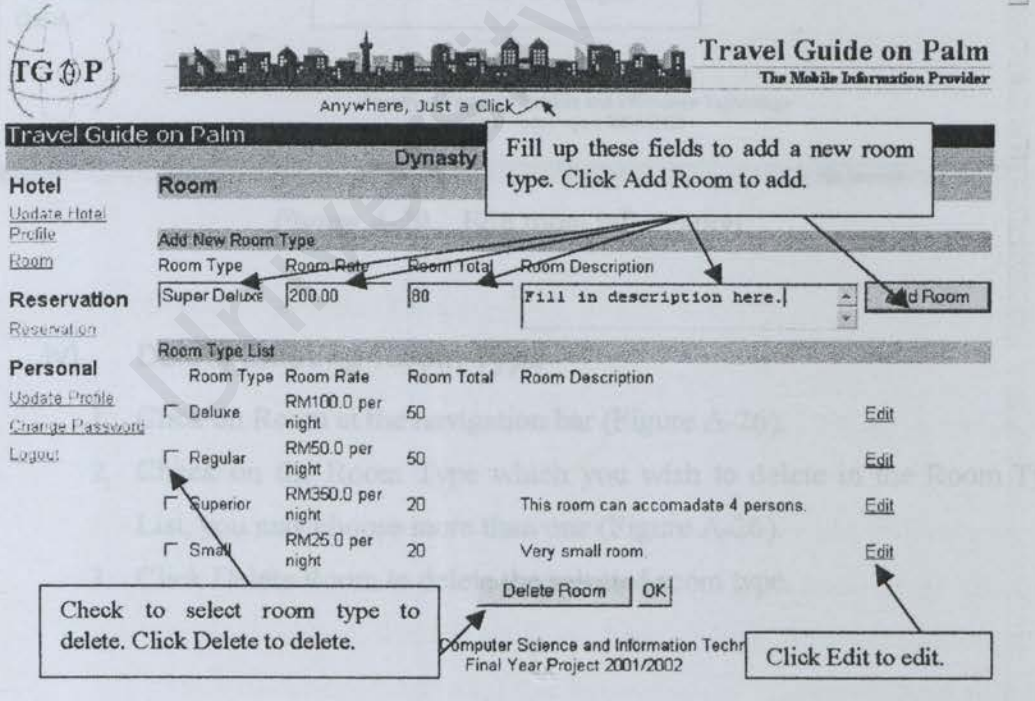

Figure A-28 Room Managing.

iii) Edit Existing Room Information

1. Click on Room at the navigation bar (Figure A-26).
2. At the Room Type List, click on Edit for the room you wish to edit.
3. Change the information as needed with valid input (Figure A-29).
4. Click Save to save the change or Cancel to abort.

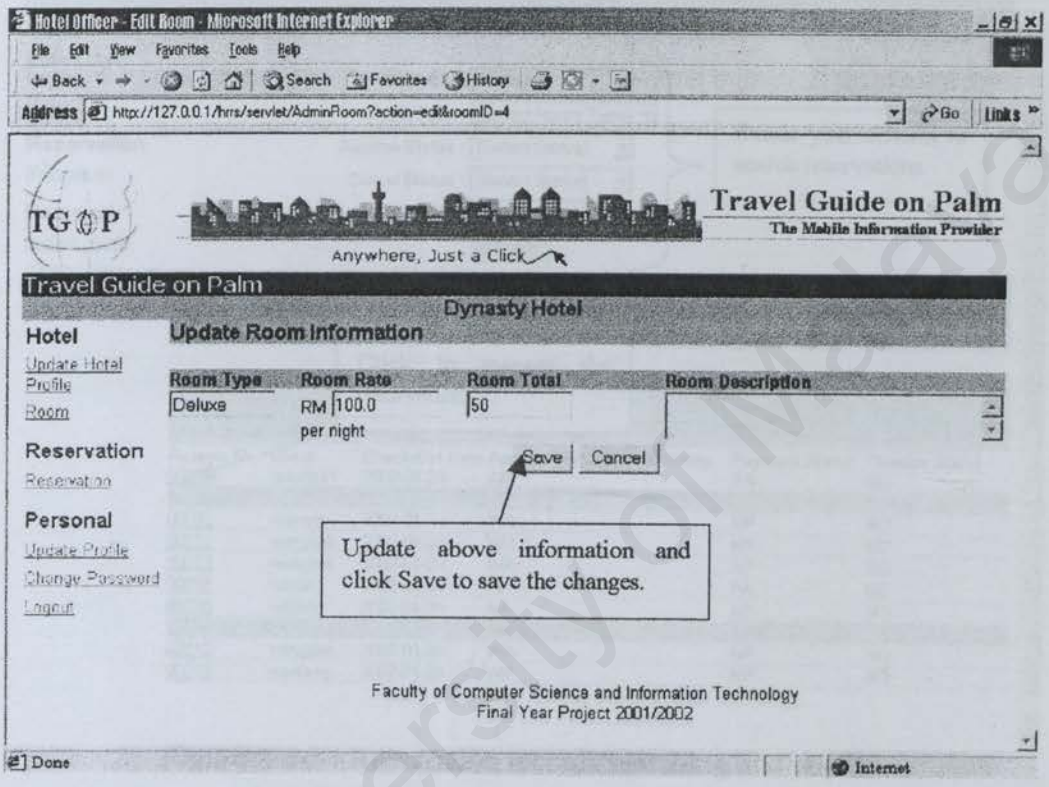

Figure A-29 Edit room information.

iv) Delete Existing Room Type

1. Click on Room at the navigation bar (Figure A-26).
2. Check on the Room Type which you wish to delete in the Room Type List, you may choose more than one (Figure A-26).
3. Click Delete Room to delete the selected room type.

v) Reservation Records Searching

1. Click on Reservation at the navigation bar (Figure A-26).
2. Select the criteria to search for the reservation records (Figure A-30).
3. The results will be displayed under the Reservation List (Figure A-30).

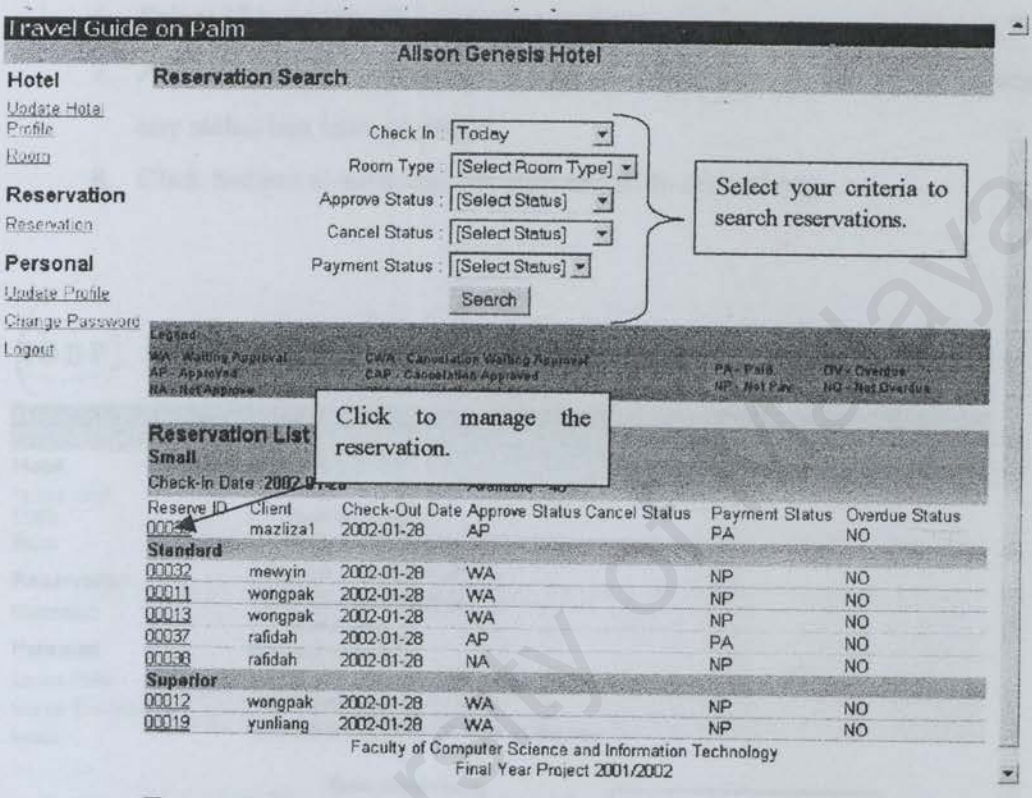

Figure A-30 Reservation records searching and results.

vi) Managing Reservation

1. Follow the instruction v)1 to v)3.
2. Click on the Reserve ID which you wish to view or approve the reservation/cancellation requests (Figure A-30).
3. If you wish to approve the reservation request, select Approve in the Reservation Approve Status list (Figure A-31). Please check from the bank to make sure that the credit card information is valid and transaction is allowed. Check on Paid if the transaction done. Please make sure that the request is made three days before check-in date.
4. Select Not Approve if you want to reject the reservation.

5. If you wish to approve a cancellation request, select Approve in the Reservation Cancel Status List (Figure A-32, this list will be shown if only the Client wish to cancel the reservation). Please make sure that the request is made three days before check–in date.
6. Select Not Approve to reject the request.
7. A email will be sent to the Client automatically by the server whenever any status has been changed.
8. Click Submit to save the changes or OK to do nothing.

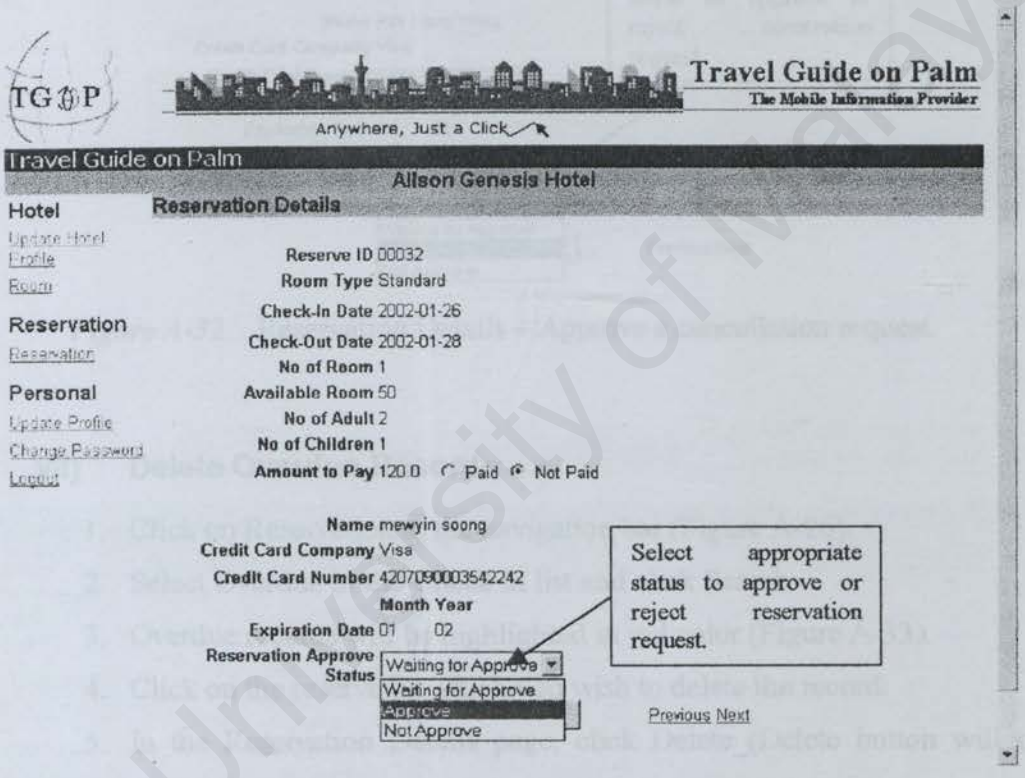

Figure A-31 Reservation details – Approve reservation request.

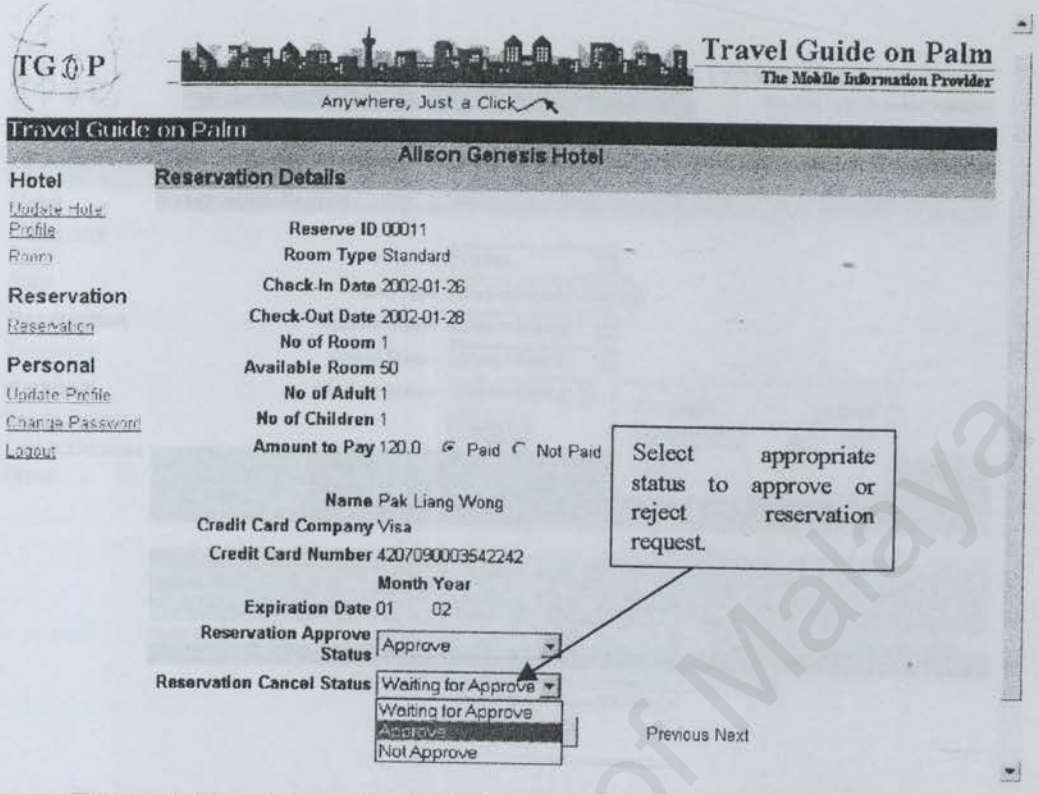

Figure A-32 Reservation Details – Approve a cancellation request.

vii) Delete Overdue Records

1. Click on Reservation at the navigation bar (Figure A-26).
2. Select Overdue in the Check In list and click Search.
3. Overdue records will be highlighted in red color (Figure A-33).
4. Click on the reserve ID which you wish to delete the record.
5. In the Reservation Details page, click Delete (Delete button will only enabled if the record is overdue) to delete this record.

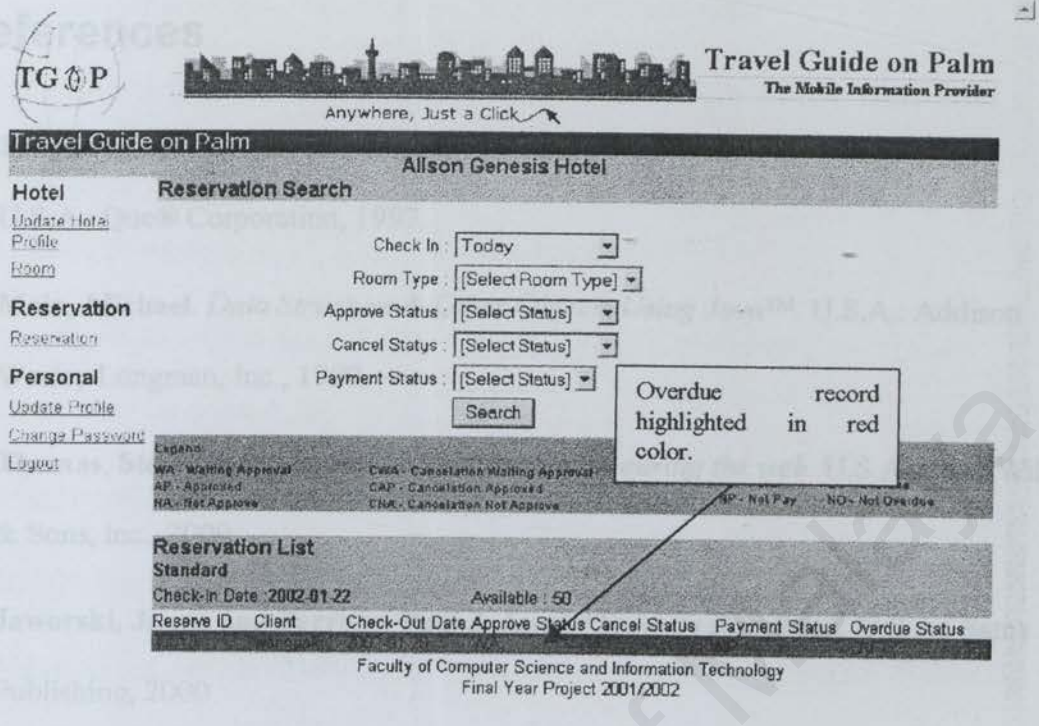

Figure A-33 Overdue record.

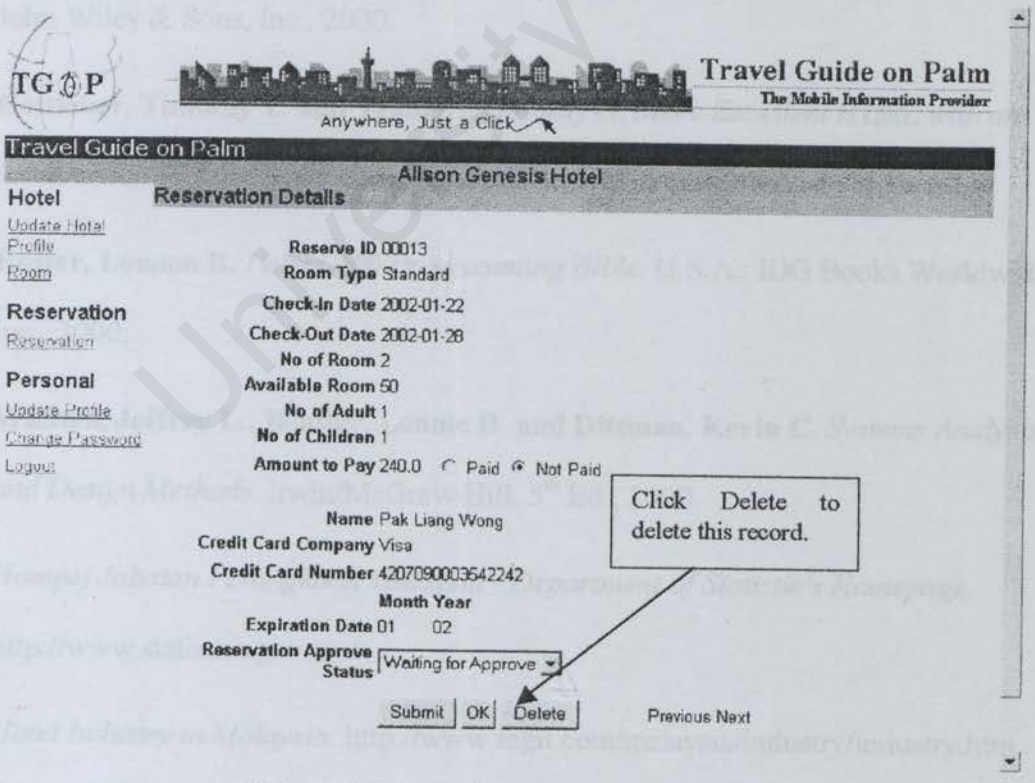

Figure A-34 Delete reservation record.

References

1. **Schneider, Jeff and Arora, Rajeev.** *Special Edition Using Enterprise Java™.* U.S.A.: Que® Corporation, 1997.

2. **Main, Michael.** *Data Structure & Other Objects Using Java™.* U.S.A.: Addison Wesley Longman, Inc., 1999.

3. **Thomas, Stephen A.** *SSL and TLS Essentials : securing the web.* U.S.A.: John Wiley & Sons, Inc., 2000.

4. **Jaworski, Jamie and Perrone, Paul J.** *Java™ Security Handbook.* U.S.A.: Sams Publishing, 2000.

5. **Giguere, Eric.** *Java™ 2 Micro Edition: Professional Developer's Guide.* U.S.A.: John Wiley & Sons, Inc., 2000.

6. **Gottleber, Timothy T. and Trainor, Timothy N.** *More Excellent HTML with an introduction to JavaScript.* Singapore: McGraw-Hill Co., 2000.

7. **Roster, Lonnon R.** *Palm OS® Programming Bible.* U.S.A.: IDG Books Worldwide, Inc., 2000.

8. **Whitten, Jeffrey L., Bentley, Lonnie D. and Dittman, Kevin C.** *Systems Analysis and Design Methods.* Irwin/McGraw-Hill, 5th Ed., 2000.

9. *Hompej Jabatan Perangkaan Malaysia - Department of Statistic's Homepage.* http://www.statistics.gov.my/

10. *Hotel Industry in Malaysia.* http://www.regit.com/malaysia/industry/industry.htm

11. *Internet and Messaging Applications*. http://www.palm.com/

12. *All-malaysia-Hotels.Com - Malaysia Hotels - KualaLumpur Hotels*. http://www.all-malaysia-hotels.com/

13. *Malaysia Tourist Arrivals 2000*. http://tourism.gov.my/

14. *Hotels and Resorts in Kuala Lumpur, Malaysia - Marimari.com*. http://www.marimari.com/

15. *Asia Travel Mart*. http://www.asiatravelmart.com/

16. *MySQL Manual Reference*. MySQL AB, 1997-2001.

17. **Maas, Brian and Hillerson, Gary.** *Palm OS® Emulator: Excerpt from Palm OS Programming Development Tools Guide*. U.S.A.: Palm, Inc., 2001.

18. **Pressman, Roger S.** *Software Engineering: A Practitioner's Approach*. Singapore: McGraw-Hill Higher Education, 5th Ed., 2001

19. **Pfleeger, Shari Lawrence.** *Software Engineering: Theory and Practice*. U.S.A.: Prentice-Hall, Inc., 2nd Ed., 2001