

**CAR RENTAL MANAGEMENT SYTEM WITH SMS
ALERT**

FAZREENY BINTI ADNAN

THESIS SUBMITTED IN FULFILMENT OF THE
DEGREE OF COMPUTER SCIENCE

FACULTY OF COMPUTER SYSTEM AND SOFTWARE
ENGINEERING

2013



UNIVERSITI MALAYSIA PAHANG

BORANG PENGESAHAN STATUS TESIS

JUDUL: CAR RENTAL MANAGEMENT SYSTEM WITH SMS ALERT

SESI PENGAJIAN: 2013/2014

SAYA FAZREENY BINTI ADNAN (HURUF BESAR)

Mengaku membenarkan tesis/laporan PSM ini disimpan di Perpustakaan Universiti Malaysia Pahang dengan syarat-syarat kegunaan seperti berikut:

1. Tesis/Laporan adalah hakmilik Universiti Malaysia Pahang.
2. Perpustakaan Universiti Malaysia Pahang dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institut pengajian tinggi.
4. **Sila tandakan (✓)

☐

SULIT

(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972) *

☐

TERHAD

(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan) *

☐

TIDAK TERHAD

Disahkan Oleh

.....

Alamat tetap:

.....

Penyelia:

Tarikh:

Tarikh:

*Sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis/laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

Declaration

I hereby declare that the idea and information in this report is my own concept for quotations and summaries without any plagiarism which have been recognize by my supervisor and PSM community.

Date: 24.12.2013

FAZREENY BINTI ADNAN

CA10034

SUPERVISOR DECLARATION

I hereby declare that I have read this thesis and in my opinion this thesis/report is sufficient in terms of scope and quality for the award of the degree of Bachelor of Computer Science ()

Signature :.....

Supervisor Name:

Date :.....

ACKNOWLEDGEMENTS

I am deeply indebted to many people who, directly or indirectly, are responsible for this project coming into being. I am most grateful to Dr. Norrozila Binti Sulaiman as my supervisor for her constructive criticism of my research in this project, and most of all her guidance and support for the whole project. I also thank all Faculty of Computer System & Software Engineering lecturers for their guidance and helps in developing the system. Finally, thanks to parents, and numerous friends who endured this long process with me, always offering guidance and supports.

EXECUTIVE SUMMARY

Technology is an absolute need we cannot escape from. It has an important role in most aspects of our lives. In other words, it is solution of every mankind problems. The importance of technology is aiming for comfort of use in whichever form it is. In this paper, it discuss about changing convenient method in managing car rental to standalone system. Before this, commonly car rental manages by record manually in a paper. These methods are not systematic and time consuming during searching and updating data. Other than that, when record data in file will risk of losing information as burnt in fire, ink split and so on. By proposed Car Rental Management System with SMS Alert will promotes better management in future. In this system, user can log on into the system and key in detail and wait for the admin response with rent a car status. There are several functions such as record information, update car status, and deleting data. In addition, it consist of special function where customer will receive SMS reminder to return a car on time and customer should not worried if forget password because SMS alert function able to sent recovery password to user. As a result, this system will make sure overall management process done efficiently and reduce time and cost in future.

RINGKASAN EKSEKUTIF

Teknologi merupakan keperluan yang sangat penting pada masa kini. Ia mempunyai peranan penting dalam kebanyakan aspek kehidupan kita. Dalam erti kata lain, ia adalah penyelesaian setiap masalah manusia. Kepentingan teknologi adalah untuk memudahkan manusia dalam melakukan pekerjaan harian. Dalam projek tahun akhir ini, ia membincangkan mengenai perubahan kaedah yang mudah dalam menguruskan kereta sewa bagi menggantikan pengurusan secara konvensional sebelum ini. Pada masa kini, pihak syarikat menguruskan segala urusan mereka dengan rekod secara manual iaitu merekod di dalam borang. Kaedah ini tidak sistematik dan memakan masa untuk mencari dan mengemaskini data. Selain daripada itu, data rekod di dalam fail mempunyai risiko kehilangan maklumat akibat daripada bencana alam sebagai contoh kebakaran, tertumpah dakwat dan sebagainya. Dengan cadangan Sistem Pengurusan Sewa Kereta dengan SMS Peringatan akan menggalakkan pengurusan yang lebih baik pada masa akan datang. Dalam sistem ini, pengguna boleh melayari ke dalam sistem dan menunggu jawapan admin dengan sewa status kereta. Terdapat beberapa fungsi seperti maklumat rekod, status kereta kemaskini dan memadam data. Di samping itu, ia terdiri daripada fungsi khas di mana pelanggan akan menerima peringatan SMS untuk kembali kereta pada masa dan pelanggan tidak harus bimbang jika terlupa kata laluan kerana fungsi amaran SMS dapat kata laluan pemulihan dihantar kepada pengguna. Hasilnya, sistem ini akan memastikan proses pengurusan keseluruhan dilakukan dengan cekap dan menjimatkan masa dan kos pada masa akan datang.

TABLE OF CONTENTS

Content	Pages
DECLARATION	3
SUPERVISOR DECLARATION	4
ACKNOWLEDGEMENTS	5
EXECUTIVE SUMMARY	6
RINGKASAN EKSEKUTIF	7
CONTENTS	8
LIST OF FIGURES	10
LIST OF ABBREVIATIONS	11

Section	Content	Pages
1	Introduction	12
1.1	Problem Formulation	14
1.1.1	Problem Statement	14
1.1.2	Objectives	14
1.2	Review of previous works / researches and relationship to current project	14-22
1.3	Explanations of the current system and its limitation	22-23
1.4	Explanations of terminology	23-24
1.5	Method(s) of approach in previous system.	24-25
1.6	Indications of scope and limitations of the study	25
1.7	Outline of material presented in previous system.	25-26
2	Report Body	27
2.1	System Requirement	27-28
2.1.1	Comparison between the previous system and system approaches	28-30
2.2	Technic/theory/flow chart/all relevant diagram	30-34
2.3	Methods and materials used.	34-36
2.4	Implementations	37

2.4.1	Introduction to system interfaces	37
2.4.2	Modem Configuration Management setup	38-42
2.4.3	System Interfaces Using Visual Basic.Net	42-52
2.5	Input and Output Design	53
2.5.1	Input Design	53
2.5.2	Output Design	54
2.5.3	Database Design	55-56
2.5.4	System Interface Conclusion	56
2.6.	Testing	56
2.6.1	Test data and descriptions	56-72
2.6.2	Testing Conclusion	72
3	Conclusion	72
3.1	System Strengths and Limitations	72
3.1.1	Strength of the systems	72-73
3.1.2	Limitations of the systems	73-74
3.2	Suggestions for future work	74
3.3	Significant contributions of the research	74-75
3.4	Summary	75
References		76-78
Appendix	The Gant Chart of the Completion of My PSM 1.	79
	The Gant Chart of the Completion of My PSM 2.	80
	SYSTEM MANUAL & CODINGS	81-100

LIST OF FIGURES

Figure Number	Content	Page
1.2.1	Block Diagram of Proposed System	15
1.2.2	Library Management System use-case diagram	16
1.5.1	Book Search Flow Chart	24
1.5.2	Flow Chart of Book Placement Algorithm	24
2.2.1	Flow Chart of Innovative Use of SMS Technology for Efficient Library Management System	30
2.2.2	Regain password flow chart	31
2.2.3	Send Reminder Flow Chart	32
2.2.4	Borrow Book Flow Chart	33
2.2.5	Library Management System Use Case diagram	34
2.3.2	COMPARISON TABLE WITH WAVECOM	35-36
2.4.1	Introduction to system interfaces	37
2.4.2(1)	Add references	39
2.4.2(2)	Object Browser	40
2.4.2(3)	Add new item.	41
2.4.2(4)	MobitekSMSAPIv5	41
2.4.3 (1)	login form for library management system.	42
2.4.3 (2)	connect modem interface.	43
2.4.3 (3)	Staff login interface.	4 3
2.4.3(4)	Member login interface.	4 4
2.4.3(5)	Menu form.	4 5
2.4.3(6)	User Registration form.	4 6
2.4.3(7)	Gain password form.	4 7
2.4	Book managing form.	4

.3(8)		8
2.4.3(9)	Borrow Book form.	49
2.4.3(10)	Return Book form.	50
2.4.3(11)	Check List form.	51
2.4.3(12)	Staff Profile form.	52
2.4.3(13)	Member profile form	52
2.5.1	Input Design	53
2.5.2	Output Design	54
2.5.3(1)	Book Database design	55
2.5.3(2)	Member Information Database design	55
2.5.3(3)	User Database design	55
2.5.3(4)	Book Borrow Database design	56
2.6.1(1)	Login form	57
2.6.1(2)	Register form obtains access into the system.	58
2.6.1(3)	Connect modem interface	59
2.6.1(4)	login interfaces for staff.	60
2.6.1(5)	Menu form which allow staffs to select which form to use	61
2.6.1(6)	login interface for system administrator.	61
2.6.1(7)	Menu form for admin.	62
2.6.1(8)	Book Management interface	63
2.6.1(9)	The list of database contain in the system.	64
2.6.1(10)	Process of saving new book(s) data into system database.	64
2.6.1(11)	Result after searching process is complete	66
2.6.1(12)	Borrow book form for user to verify their information	67
2.6.1(13)	Receipt form that user will receive after borrow book from library.	68
.6.1(14)	Return book form to check borrower records.	68
2.6.1(15)	Staff profile form where to update and save staff information.	70
2.6.1(16)	Member login interface.	70
2.6.1(17)	Member profile form where to update and save member information.	71

LIST OF ABBREVIATIONS

- i. Radio Frequency Identification (RFID)
- ii. Global Positioning System (GPS)
- iii. Telemetry Transmitters (TT)
- iv. Return on Investment (ROI)
- v. Application Programming Interfaces (API)
- vi. Global System for Mobile (GSM)

1.0 INTRODUCTION

A car rental is a vehicle that can be used for a short periods of time by paying a fee. The individual are able to rent a car that suitable with them in affordable rate. Normally, the car rental business conducted in major cities and strategic places, airport or an island. With the car rental service, customers do not have to worry about moving from one destination to another without bringing own car. Car rental agencies provide car rental services to customers for the period required by the user. However, users must have a license before the vehicle is allowed to rent a car from the agency. With the car rental services enable users who do not have vehicles, have emergency cases, outstation rent a car to easier moving to target destination without worry to find public transport. However, some car rental agencies are still using manual system for managing rental car operations. No doubt they use the internet to advertise their space agencies but there were no specific systems to storing customer data or business process in the system.

CarRental2U is a car rental agency located in Seri Kembangan, Selangor. The agency is owned by a husband and wife and handled by them. Management of rental car is manually operated. The agency is open from 10am to 10pm. they do not have a rental car management system online. However, to make them known to the agency, Facebook is one medium to promote their agency.

Develop a system that will is a standalone system which has two interfaces, namely staff and admin, and customers. Some of the functions that are present in the system are the database of customers, staff, train, and car status. In addition, there are special features added to the system of sending SMS to the customer if you forgot your password and SMS reminders borrowing the car rentals.

En Sahlan Bin Muhammad Majid an agency owner and his wife are staff at the agency. Every morning, before the launch of the operating agency, his wife, Puan Halizah Bint Kamal will check all the car rented record. This is because to make sure customer will return a car on time. Pn. Halizah will also check the status of their agency's rental car that is free from police summonses after one week from the date of return of the rental car to ensure that each suit must be paid by the tenant in accordance with the agreement of both parties to the contract that was signed.

As usual, most of the customers have to contact the agency by phone in advance to check whether the date for the rent it's available or not available. After receiving confirmation that the vehicle is available on the required date, the customer will come to the agency for the rental of vehicles. CarRental2U always had been the choice for customers to rent a vehicle even if the company is just a small sized stores and management manually. The purpose of its customers rent vehicles in the agency is to use the size of MPV vehicles to wedding event, their car broke down and was in the workshop, a holiday with the family and so on.

To rent a vehicle, customers are required to fill out the form provided in advance. All car rental records are compiled in a file. For regular customers, they also need to fill out the form as to save time to rediscover the form that was previously in the file. Then, Pn. Halijah will record the detail date of the loan and the date the vehicle returnk in the record books. Receipt will be given to customers after making the payment during the process of renting a vehicle. Money deposit will be returned after the customer returns the vehicle.

1.1 Problem Formulation

1.1.1 Problem Statement

In this paper, I will propose a Car Rental Management System with SMS Alert to changing from manual system to standalone system. This is because the manual system is not systematic and time consuming during process of searching and any update done quite long time. Furthermore, by using manual system may risk losing of information such as by natural disaster, burning and losing of files. The system proposed will consist of adding, deleting, updating information in the system. The system is providing with security protection where only authorized people can log in into a system. Car Rental Management System with SMS Alert (CRMSS) consist of special feature GSM technology connect with the system. GSM Technology is a device that similar function with hand phone by insert sim card to running a function. It can send a SMS or MMS to hand phone. By using this technology in this system for sending reminder message to make sure customer returned a car on times. In addition, hand phone one of the important thing they bring everywhere. So of course by using SMS alert will be the best option to make sure return a car on times. Agencies may loss customer if

the previous customers do not return a car by on time. Other function of SMS alert is user can regain password in a short period of time if they forgot a password by sending direct to hand phone.

- 1) Manual system takes a long time to make a review of the list of records of customers who want to rent a vehicle in the future.
- 2) Staff must record the details of vehicles that have been hired each time renting a car.
- 3) Possibility to record the customer's name over and over again or missed the name of customers who want to rent a car will result in a loss to the agency.

1.1.2 Objectives

The objectives of this research are:

- 1) To develop a standalone system can record all the staffs and members information. Two types of interface: interface for staff and admin, and interface for customers.
- 2) To enhance current system become staff do not have to key in the customers basic information again because all the information is already stored in database.
- 3) To give protection for all the data is safely because only authorised people can access into this system.

Target users:

- 1) Staffs in car rental agency
- 2) Admin
- 3) Customers who rent a car from this company

1.2 Review of previous works / researches and relationship to current project

In this chapter, discusses about literature review of the project. It comprises of description about existing system and the problem faced by the users. Literature review will study and relates the journal or article about the business management and how far impact of technology in company profit. Several existing system also will discuss briefly in this chapter about the characteristic, advantages, and disadvantages the system. Different types of framework, technology and method use by different car rental agency in order to compete between each other. In this chapter, by study and analyze all the data is important to develop and proposed better system to replace existing manual car rental management system. Mostly the agency that using latest technology to manages their business is large company and already well known at that area. So, the company afford to invest in improvement of management by promote latest technology in the business management.

The rapid growth of interactive Internet services has led to constantly increasing Web sites (Berners-Lee, T., et. al, 1992). We have also experienced an increase in their functionality, which, in turn, makes them more complicated to use (Nielsen, J., 2000). Thus, a useful web application should be based on traditional graphical user interface designs which rely on menu selection and navigation. But these interfaces require a considerable cognitive overhead. This may be tolerable to frequent users, but will in many cases deter casual users, especially those who are not yet used to computers. (Nielsen, J., 1993)

Management Information System is a system or process that provides information needed to manage organizations efficiently and effectively. Management information systems involve three primary resources: technology, information, and people. It's important to recognize that while all three resources are key components when studying management information systems, the most important resource is people. Management information systems are regarded as a subset of the overall internal controls procedures in a business, which cover the application of people, documents, technologies, and procedures used by management accountants to solve business problems such as costing a product, service or a business-wide strategy. Management information systems are distinct from regular information systems in that they are used to analyze other information systems applied in operational activities in the organization. Academically, the term is commonly used to refer to the group of information management methods tied to the automation or support of human

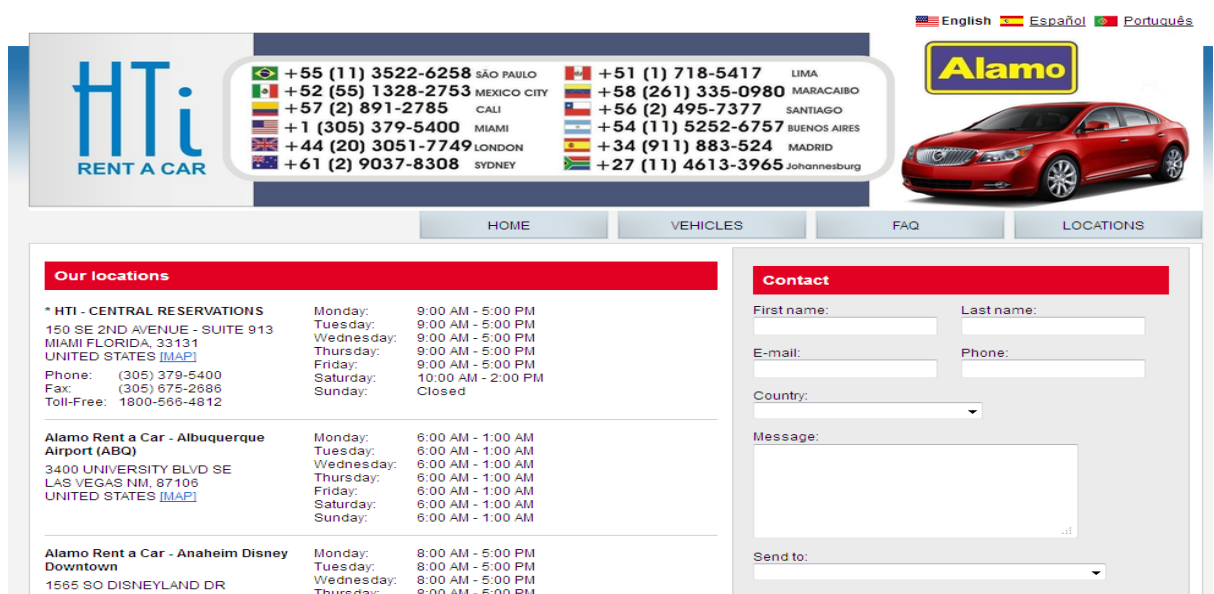
decision making, e.g. decision support systems, expert systems, and executive information systems (Wikipedia, Online).

Meanwhile a web service is a collection of protocols and standards used for exchanging data between applications or systems. Software applications written in various programming languages and running on various platforms can use web services to exchange data over computer networks like the Internet in a manner similar to inter-process communication on a single computer.

Furthermore, the meaning of software engineering is the computer science discipline concerned with developing large applications. Software engineering covers not only the technical aspects of building software systems but also management issues such as directing programming teams, scheduling and budgeting. Software engineering has several documentation that must done with the development of the system. There are Software Development Plan (SDP), Software Requirements Specification (SRS) and Software Design Development (SDD).

1.3 Explanations of the current system and its limitation

Before proceed with developing the new system for CarRental2U, deeply study about the current system is done. All the description about the process in car rental management is record to make sure knows about the overall flows of management very well. The manual system process before, when customer walk in to the agency, staff will fill the information about user first. This is because to make sure if anything happen such as accident or lost, can report to the police and claim for insurance. Other than that, customer should wait about 15 minutes for processing their request. Other method to rent a car by SMS to agency first for booking first before rent a car. By this way can reduce time waiting for processing but to make sure customer is serious to rent a car, transaction should be done by bank in deposit to the agency first. After that, staff from the company will sent a car direct to the customer and brings along the form to fill in by the customers. All the information will stored in a file by manually. By this method will cause time consuming because should wrote by pen all the customer information by manually. For the return customer also, staff will fill in again the information because if searching back the record from the file, it will take time and sometimes previous record is lost and cannot be found.



The screenshot displays the HTi RENT A CAR website. At the top, there are language options: English, Español, and Português. Below this is a header with the HTi logo and a list of international phone numbers for various cities including São Paulo, Mexico City, Cali, Miami, London, Sydney, Lima, Maracaibo, Santiago, Buenos Aires, Madrid, and Johannesburg. To the right of the header is an Alamo logo and a red car. The main navigation bar includes links for HOME, VEHICLES, FAQ, and LOCATIONS. The 'Our locations' section lists three locations with their addresses and operating hours. The 'Contact' section contains a form with fields for First name, Last name, E-mail, Phone, Country, and a Message box, along with a 'Send to' dropdown menu.

Our locations		
* HTi - CENTRAL RESERVATIONS 150 SE 2ND AVENUE - SUITE 913 MIAMI FLORIDA, 33131 UNITED STATES [MAP] Phone: (305) 379-5400 Fax: (305) 675-2686 Toll-Free: 1800-566-4812	Monday: 9:00 AM - 5:00 PM Tuesday: 9:00 AM - 5:00 PM Wednesday: 9:00 AM - 5:00 PM Thursday: 9:00 AM - 5:00 PM Friday: 9:00 AM - 5:00 PM Saturday: 10:00 AM - 2:00 PM Sunday: Closed	
Alamo Rent a Car - Albuquerque Airport (ABQ) 3400 UNIVERSITY BLVD SE LAS VEGAS NM, 87106 UNITED STATES [MAP]	Monday: 6:00 AM - 1:00 AM Tuesday: 6:00 AM - 1:00 AM Wednesday: 6:00 AM - 1:00 AM Thursday: 6:00 AM - 1:00 AM Friday: 6:00 AM - 1:00 AM Saturday: 6:00 AM - 1:00 AM Sunday: 6:00 AM - 1:00 AM	
Alamo Rent a Car - Anaheim Disney Downtown 1565 SO DISNEYLAND DR ANAHEIM CA, 92802	Monday: 8:00 AM - 5:00 PM Tuesday: 8:00 AM - 5:00 PM Wednesday: 8:00 AM - 5:00 PM Thursday: 8:00 AM - 5:00 PM	

Contact	
First name:	Last name:
E-mail:	Phone:
Country:	
Message:	
Send to:	

HTIRentACar is a company that provides car rental services online. The interface is a simple website can provide comfort to the user to use it. The function that provided on the website is to make a reservation, and price list of vehicles for rent, car service branch address. To make a reservation, customers should put the date of borrowing and returning, location to pick up a rental vehicle and then select the desired vehicle. After that, the customer must fill out the personal details and the booking process was successful without having to pay the cost of the car first.

The screenshot displays the Enterprise website's car rental reservation interface. At the top, the Enterprise logo is on the left, and navigation links for 'Language: English', 'About Us', 'Contact Us', and 'Help' are on the right. Below this is a horizontal menu with options: 'Rent a Car', 'Business Rentals', 'Rent a Truck', 'Buy a Car', 'Manage Your Fleet', and 'Careers'. The main content area is divided into three columns. The left column contains links for 'Buy a Car', 'Used Car Deals', 'Join Our Business Rental Program', 'Long Term Rentals', and 'email extras'. The middle column, titled 'Make a car rental reservation', contains a 'Rent a car in:' dropdown set to 'US', a 'Pick Up Location' field, 'Pick Up Date & Time' and 'Return Date & Time' dropdowns, a 'Vehicle Class' dropdown, a 'Renter's Age' dropdown, and a 'Search' button. The right column features a 'Login to Enterprise Plus' section, a promotional banner for 'Weekend Special \$9.99 per day', and an 'Introducing Enterprise CarShare' section. The bottom of the page has a green footer.

Enterprise car rental is a large car rental agency and offers a wide range of other services in the same company. For example, customer may rent a car for a few days and also for a long time in an affordable price. The company promote a various attractive deals on offer special for their customers. The website packed with info and also briefly explanations makes the user easy to use the website. The service provided on the website are: Rent a Car , Used Cars for Sale , Manage Your Fleet, Careers, About Us ,Contact Us, Reservation , Vehicles , Locations , Business Rentals , Print Your Receipt , Help, Travel Agent ,Site Map ,Customers with Disabilities , and Global Franchise Opportunities .

For users who want to change the date of booking the car after signing up, they can do so without incurring any additional charges. When customers have been convinced by the change, they may print the receipt detail rent it.

CarRentals.com is a car rental agencies located in the United States guarantees the lowest car rental prices in addition to the promotion of interest. The users are also now able to browse the website of the car rental using a smart phone for user's android phone and iPhone app. Order of the functions contained in the website is organized and the use of the color orange as a theme is a perfect choice for showing the website attractive and convincing customers.

HAWK Rent A Car is a car rental company that is known and has many branches, both in and outside the country. Between the countries involved are Malaysia, Singapore and Hong Kong. The company is certainly well known and various services offered by the

company as an hourly car rental, monthly car rental, and service or rent a car and personal driver. Types of vehicles offer also consists of a variety of brands to provide satisfaction to our customers. On the website, customers do not need to go to another page for all information included on the main page. Function list offered on the website are:

- 1) Home
- 2) Fleet guide
- 3) Services
- 4) Corporate information
- 5) Regional offices
- 6) Worldwide booking

For this chapter, we will cover all the relevant advantages, disadvantages and limitations contained in the four car rental website. This is important because as a reference to create a new system more organized and meet the needs of customers and staff in the management of a rental car in future.

Company	Advantages	Disadvantages	Limitations
HTIRentACar	-A simple home page and complete the user easily make use of the website.	-Vehicle rental rates are not mentioned on the website when the customer reviews of vehicles offered	-Lack of service options offered on the website makes customers wonder of new vehicles offered by the company and the type of promotion that will be offered.
	-Users can continue to make reservations by date are in want before selecting the vehicle they want.	-Use colours that are ambiguous and less interesting that makes website looks bland and unexciting.	-No information related to the company makes a bit less confident customers some background related companies.
Enterprise Car Rental	-Orderly arrangement of content and security features for which there are two interfaces, namely to members and non-members.	-The various services offered by the company other than car rental customers result in a bit confused to select the function button provided.	-No list of vehicles and rental prices shown in the website.
	-When booking, we can choose the type of vehicles demanded reservations in advance to ensure that the vehicle is available as a desired date.	-No explanation on the advantages gained to make car rental booking as member and non-member	

	-Arrangement of various functions in an organized and neat in a homepage display makes the user easy to use.		
	-Choice of language is also an advantage for the website because it is a bonus for some countries are still not familiar with foreign languages.		
CarRentals.com	- An attractive website, the use of bright colors allows it to appear cheerful and interest to the customer to explore the entire website.	-To know in detail related to car rental agencies, they must become a member first.	-No rental car rates specified in the site
	-Each function will be included also some explanation to facilitate customers to use each function contained in the website.		
	-Button function is also simple and straight to the next page without having to scroll down to find the desired page.		

	-Users do not need to make reservations in advance, but they can find a suitable date with the rental vehicle may be required in advance of the date desired.		
Hawk Car Rental	-Rental vehicle prices stated on the website.	-Features on the main page a bit stringy and irregular because there is ad space rented to outside parties.	-Confusing when making transactions money the company have to many branches around the world.
	-All promotions mentioned in the main page.		

1.4 Explanations of terminology

Web based- it is an application that is usable and addressed over a computer network such as the internet or an intranet and uses HTTP as its primary communication protocol.

Platform- it is the combination of hardware architecture and a software framework that allow both to run together.

1.5 Method approach in previous system

Hypertext Preprocessor (PHP) is used to design the web based car rental management system. PHP is chosen as a language because allows web page builds, and flexible for complex programming tasks.

1.6 Indications of scope and limitations of study

Nowadays, using manual systems management performance rental car is less relevant because it will reduce productivity in management. Besides, by doing it manually managing rental car will take a long time and storing documents in the file are at risk of missing information. Users who will use the system consist of admin, staff and clients. The user requirement and existing manual procedures of management process will be collected through interview and observation for applying into the future system.

1.6.1 Project scope

Modules:

- 1) This application consist two types of framework: staff and admin, and customer only.
- 2) The application is design in simple user interface and easy to use by all background of user.
- 3) The application allow user to fill in a form before staff will approve or reject due to availability of car rental.
- 4) The application is developed by using Microsoft Visual Studio 2008

Chapter 2 Report Body

This section describes the general function and requirement of the project. It also includes the product constraint and assumption. The specific requirement will be defined in details in part 3 of the technical report.

The following subsections consist in this section are:

- 4) product perspective
- 5) product functions
- 6) user characteristics
- 7) constraints
- 8) assumptions and dependencies
- 9) apportioning of requirements

2.1 System requirement:

GSM

GSM is a 2nd generation cellular mobile system to cater voice services and data delivery using digital modulation. The chosen device in this project is Q24 GSM MODEM to provide sending SMS in our project.

Microsoft Visual Studio 2008

Microsoft Visual Basic 2008 as the language used in the develop system because it is simple and easy when compared to the other language. If we are having trouble coding used, can easily find the solution on the internet with detailed descriptions of the main reasons the function that is built does not work.

Microsoft office 2007

Microsoft Access 2007 is a relational database management system that's combines the relational Microsoft Jet Database Engine with a graphical user interface and software-development tools. Microsoft Access as database use is because it suits with VB2008 and has the ability to link data in its existing location and use it for viewing, querying, editing and reporting.

2.1.1 Comparison between the previous system and system approaches

To ensure that the resulting system is a competitive system and has a market value of its own, some web-based system was used as the reference. This is important to ensure CRMSS can meet staff and customer satisfaction CarRental2u. Studied various aspects of the existing system in terms of advantages, disadvantages and limitations is importance in order to making better system. Existing system of reference is important because for sure meet the staff and customers. For example, the logical functions that should be present in the system are existed. In addition, the arrangement and color also play a role in the production of a good system. After studying the functions available in four web-based systems, we are able to make the evaluation and improvement of the resulting system. In essence, the resulting system is a web-based system, but for reasons only suitable GSM modem code in VB2008 and to avoid the resulting system experiencing problems, we use a standalone system. There is the advantage of this system is the GSM modem is used to enable the user knows their password if forgotten and staff can send reminders to customers to remind them to return the vehicle on time.

2.2 Technic/ theory/ flow chart/ all relevant diagrams

In this report, my systems have been classified into four types of flow chart. One overall system flowchart, three functions flow chart, and one use case diagram which represent the types of forms that can be accessed by each categories of users.

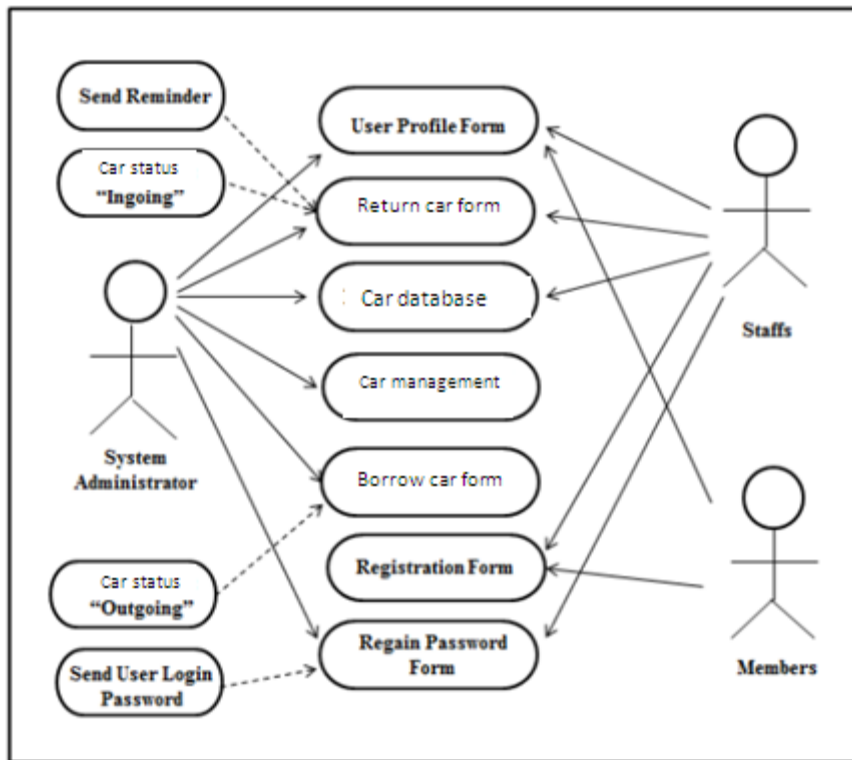


Figure 2.2.1 shows: Car Rental Management System Use Case Diagram

Use case diagram above shows the type of form that different category of users able to log on in car rental management system.

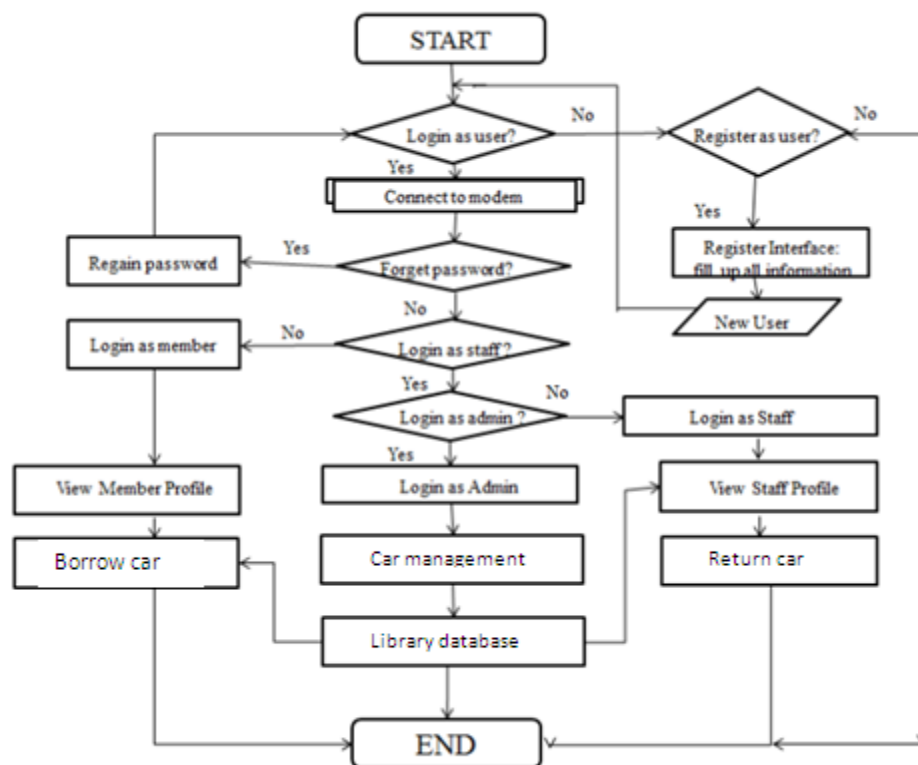


Figure 2.2.2 shows: Flow Chart of Car Rental Management System with SMS Alert

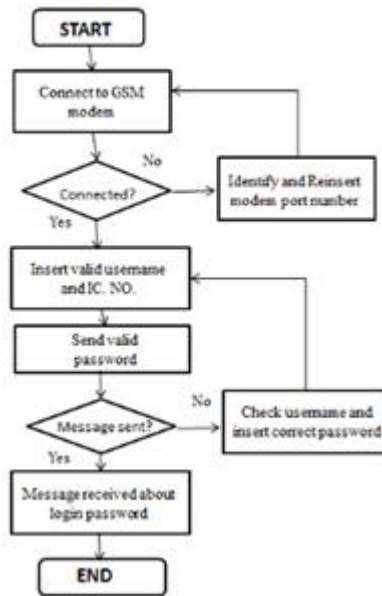


Figure 2.2.3 shows: Regain password flow chart

Based on the figure 2.2.3, the function above is focus on the staff and admin only as a trial for this system. For the staffs who forget their password should not worry because staff can key in the IC no and name to regain password. Just a few second staff will receive SMS to get the password.

2.3 Methods and materials used

The materials that used in this project are based on the objectives of my system. I had used the sending and receiving sms algorithm when handling users passwords and their books borrowed records.

GSM Modem

My chosen device in my system is Q24 GSM modem.

MOBITEK[®] Q24 GSM Modem is a messaging device. This device is able to fulfill the demand of heavy traffic and reliable data collections tool. It contains a USB cable that uses 2 USB ports which is in Y shape cable. The design of the cable allows the modem to draw

sufficient power from the computer, thus improves stability when sending and receiving text messages from the users.

The general specifications of the modem is listed as follows:

Radio Frequency: GSM and GRPS class 10

1. Quand Band: 850 / 900 / 1800 / 1900 Mhz -- it can be deployed in all GSM networks around the world
2. Core Module: Wavecom Q24 Plus -- using a reputable and reliable manufacturer, result is reliability and stability
3. Connectivity: USB or Serial -- allows connection with all lagacy and modern hardware.

2.4 Implementations

Implementation will be discussed in the relevant planning of data analysis, / model / process / hardware into Implemented been selected algorithm. the processes involved in the development of the system will be discussed in full to enable users to understand the whole connection CRMSS this.

2.4.1 Introduction to system interfaces

In this chapter, will be discussing on each interface along with comprehensive functions. This system consists of:

- i.Login (login interface as main page)
- ii.Agreement (user should agree before proceed register interface)
- iii.Borrow car (form to rent a car)
- iv.Car list (details cars and status either available or not available)
- v.Car manage (interface to manage all cars)
- vi.Car types (collection of cars with price description)
- vii.Contact (address and contact number company)
- viii.Customer promo (promotion for this month)
- ix.Gain password (interface to regain their password)
- x.Menu (option staffs and users can choose)
- xi.Policy
- xii.Receipt (detail after fill a form for rent a car)
- xiii.Register user (fill basic information)

- xiv.Return car (record car that has been return to company)
- xv.Term
- xvi.Testimonial (comment from previous customers)

2.4.2 Modem Configuration Management setup

In this sub-chapter, the setup of GSM modem is explained in more details.

Installation Guide

1. Screw the antenna. Warning: warranty does not cover antenna. Do not over-screw the antenna connector, otherwise the internal cable will be broken!
2. Press the button to eject the SIM tray. Before inserting SIM card into modem, please disable the SIM lock by using hand phone. Insert SIM card.
3. Install USB driver first! DO NOT plug the GSM modem into the USB port.
 - a) install the driver (version 2.0.0.18) for Windows 98 / 2000 / XP / 2003
 - b) install driver (version 2.0.2.8) for Windows XP / Vista
 - c) install driver (version 3.3.0.1) for Windows Vista / 2008
4. The USB cable is “Y” shape. Your computer must have 2 available USB ports. Attach one end of the USB cable to the modem. At the other end, plug into 2 USB ports.
5. Right click on "My computer", click "Manage", then "device manager > Ports (COM & LPT)", look for "Prolific USB-to-Serial Comm Port"
6. Look for the COM port number in the bracket "Prolific USB-to-Serial Comm Port (COM3)" the modem is now connected to PC at COM port no. 3.
7. The red LED light beside the antenna indicates the following status:
 - a)on – power is turned on;
 - b)on, and blinks – power is turned on, and connected to GSM network, modem is ready to be used for SMS application.
8. For SMS API – COM ActiveX (one of the SMS Gateway Development Kit), always remember to specify the COM port number when calling the "ModemInit()" . Eg. to initialise modem at port number 3, call the function ModemInit(3) . If successfully connected, it will return "True".
9. The next step will be the setup of MorbitekSMSAPIv5.
 1. Click SMS API version 5 -> Install SMS API (ActiveX DLL) version 5.1 - MobitekSMSAPI51.dll. The ActiveX DLL component (MorbitekSMSAPI51.dll) is automatically registered upon installation.

2. The MobitekSMSAPI51.dll is installed into "Windows\System" or "Windows\System32" folder. Later, if you want to uninstall it, just goto "Control Panel - > Add/Remove Programs" to do so.

3. Connect the GSM modem to serial port or USB port. For serial port, it is either COM Port 1 or COM port 2. For USB port, please view the video demo files below:

- My Computer
- Control Panel

4. Before inserting SIM card into GSM Modem, you must disable to SIM lock using your mobile phone.

10. Add MobitekSMSAPI5 functions into the visual studio.

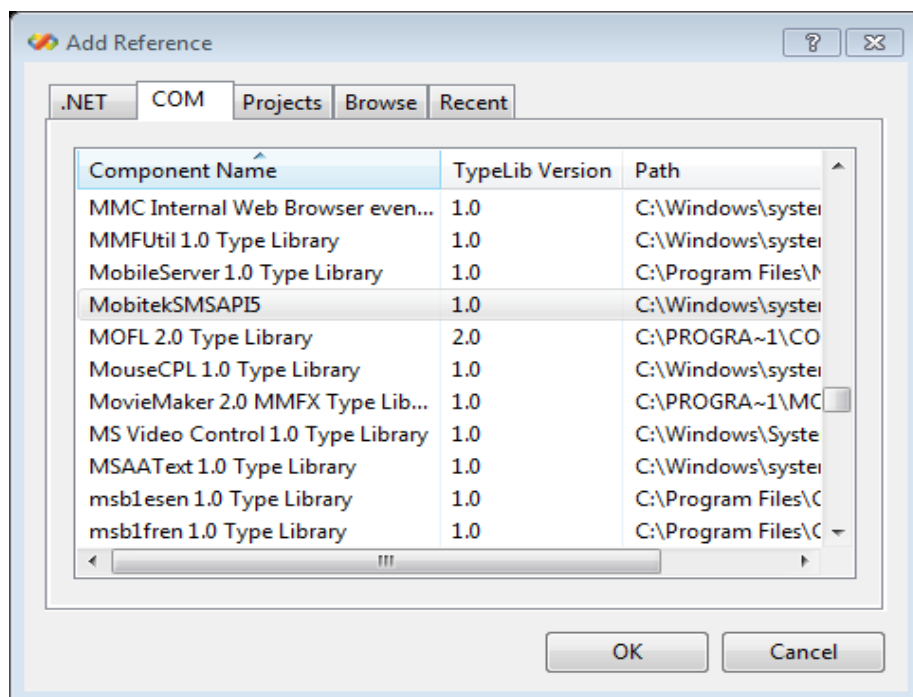


Figure 2.4.2(1) shows: Add references

11. Then, verify the installations by check up on the object browser. If there is a drop down list called Interop.MobitekSMSAPI5. You have successfully enabled that functions.

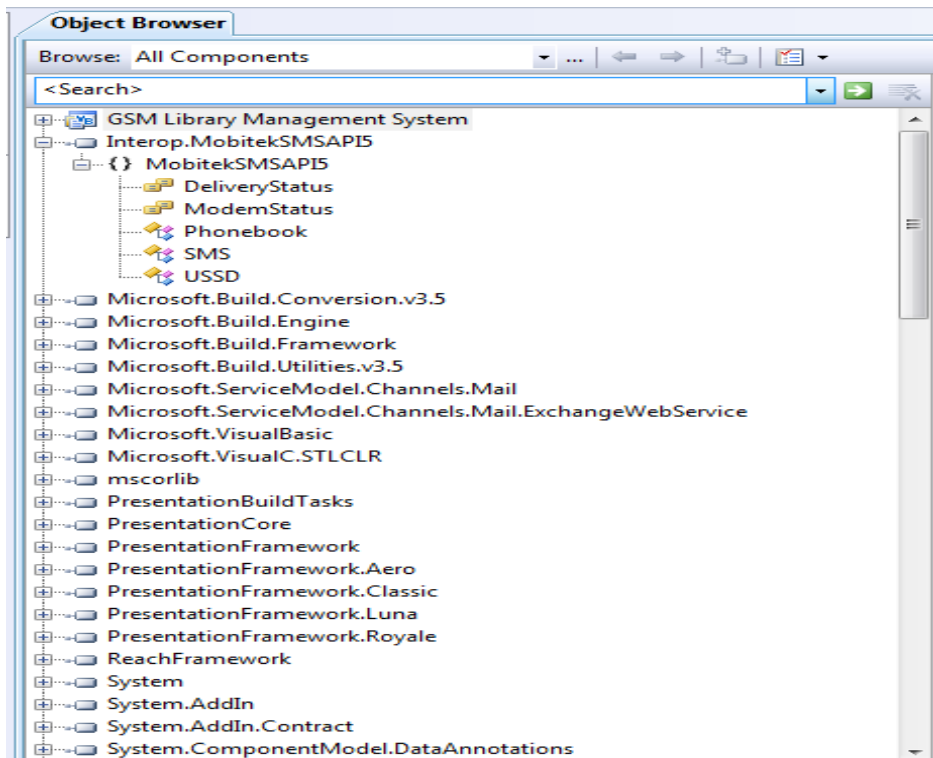


Figure 2.4.2(2) shows: Object Browser

12. Create a module to store and write the coding of SMS notifications.

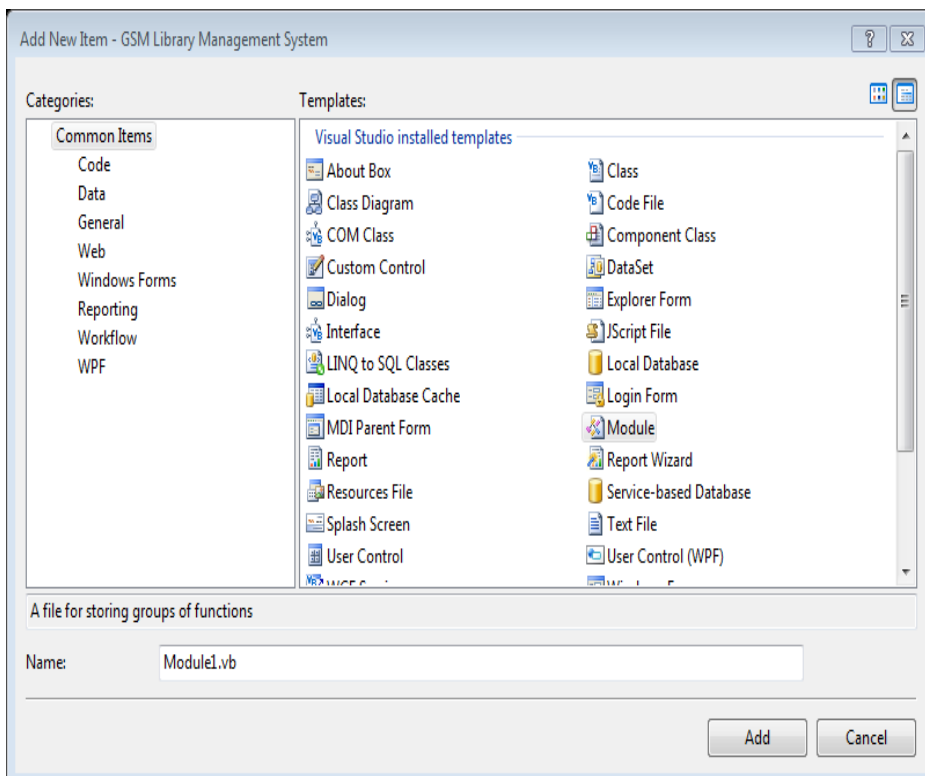


Figure 2.4.2(3) shows: Add new item

Modem status can be categorized into 6 types which are:

1. Fail: Indicate that gsm modem not connected
2. OK: Indicate that gsm modem is connected
3. PINRequired: Indicate that need to insert the PIN for the Sim card.
4. PINWrong: Indicate that the insert PIN is incorrect and after 3 tries, the Sim card will be blocked.
5. SIMBlocked: Indicate that Sim card had been blocked
6. SIMError: Indicate that the Sim card had problem and fail to detect.

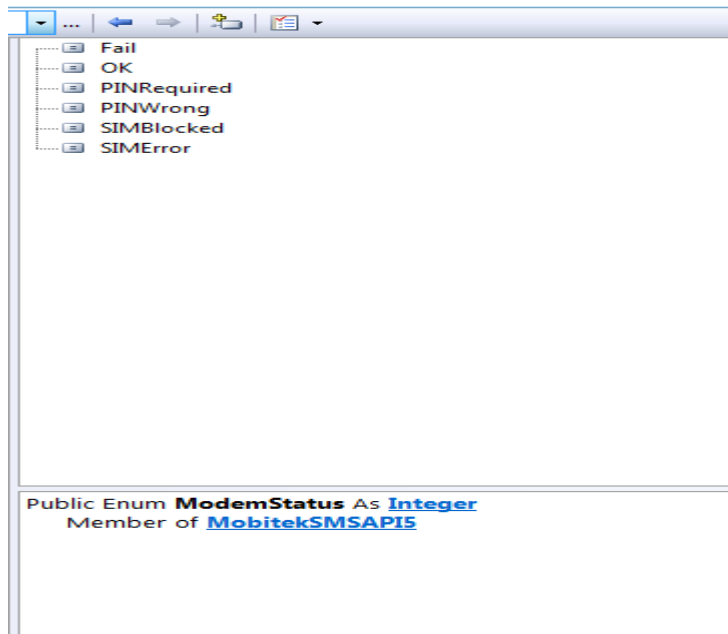


Figure 2.4.2(4) shows: MobitekSMSAPIv5

13. After that, you may continue your own coding that fit and able to connect with the GSM modem devices.

2.4.3 System Interfaces Using Visual Basic.Net

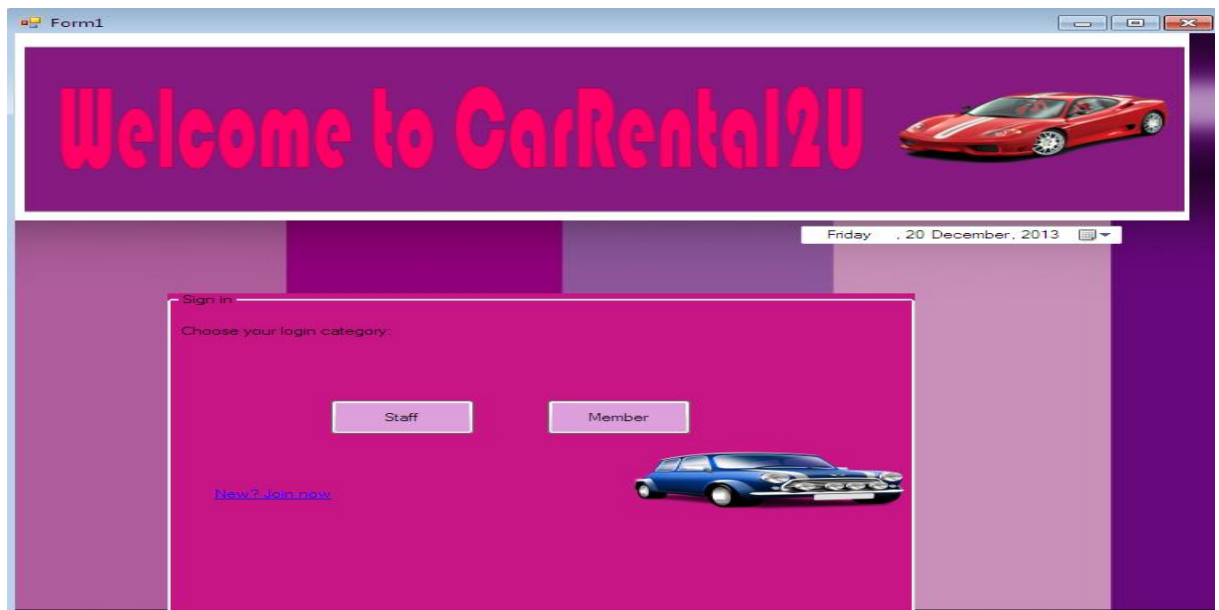


Figure 2.4.3(1) shows: Login interface for staff, admin and member

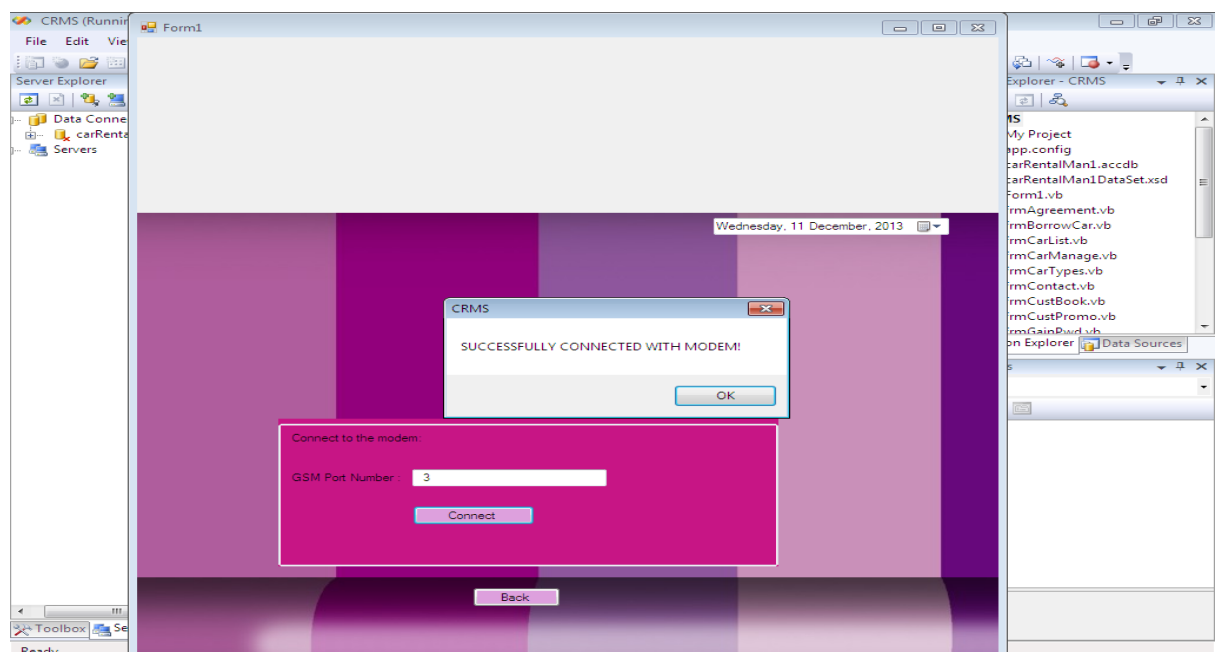


Figure 2.4.3(2) shows: GSM connection interface

the staff and the admin should ensure GSM modem connect before logging in for GSM functions such as sending SMS reminders to customers and to enable staff to retrieve your password.

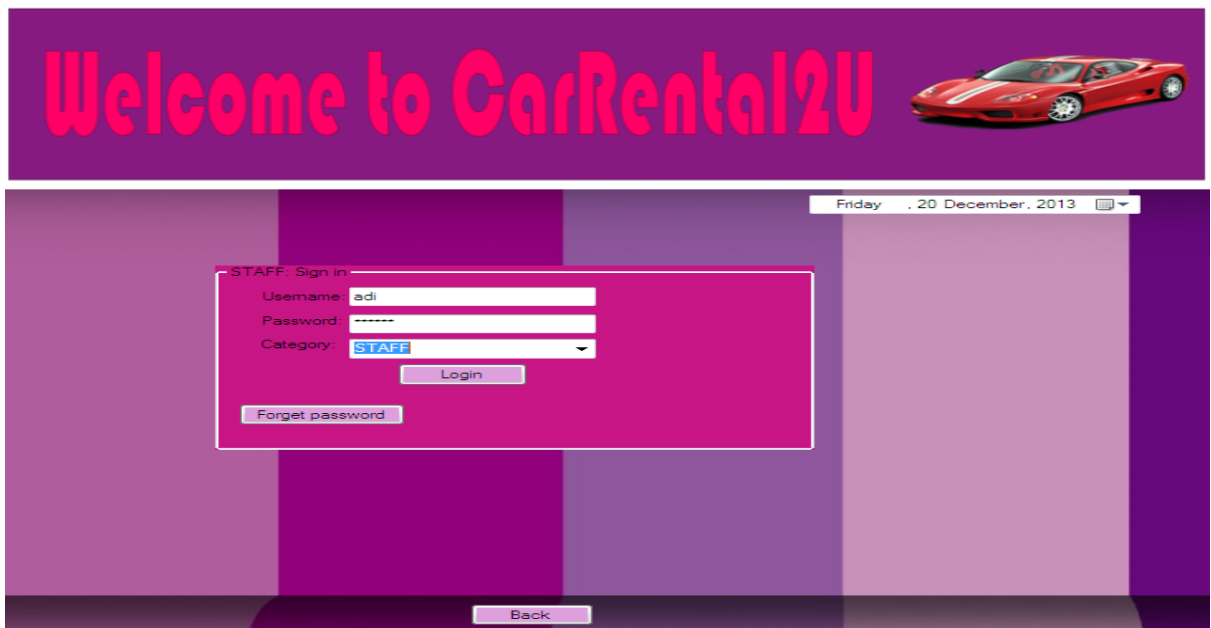


Figure 2.4.3(3) shows: Staff log in interface

Staff login interface has two categories namely staff and admin. In general, admin accounts owned by the owner of the agency that has full authority to control all the functions contained in the system. Admin also reserves the right to update the maintenance car rental listings.



Figure 2.4.3(5) shows: Menu interface for staff

By providing staff menu, staff easier to manage car rental process by clicking the button function only to proceed to the next form.

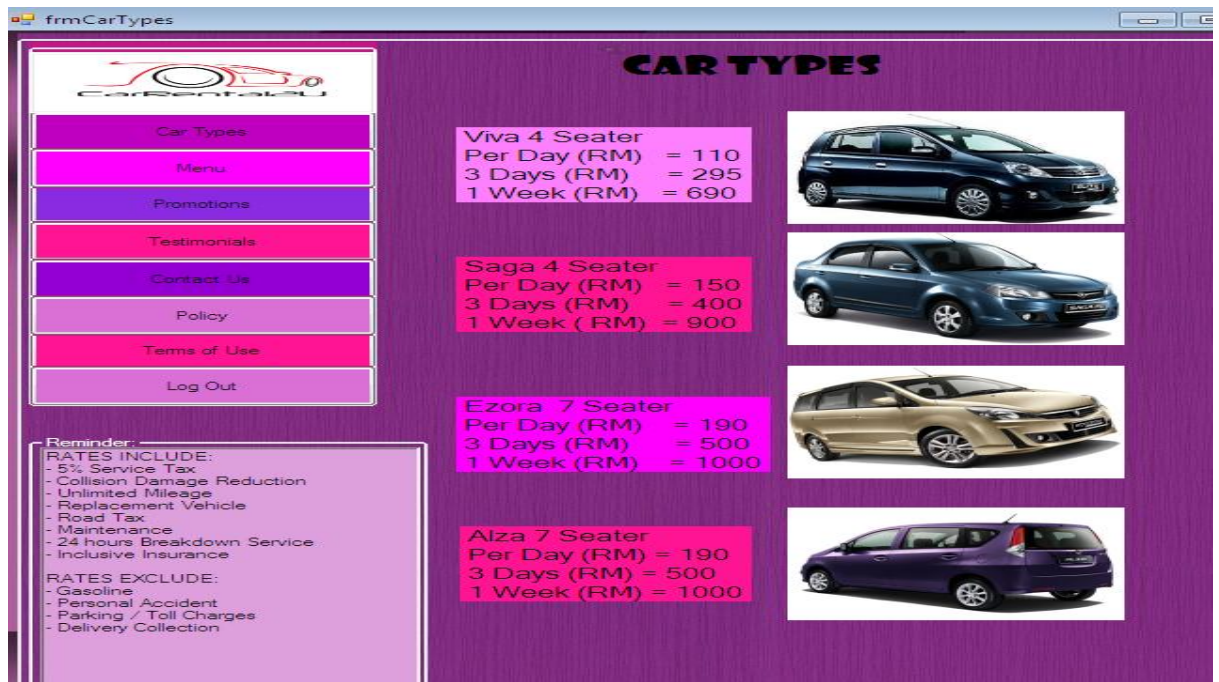


Figure 2.4.3(6) shows: Car types interface

Based on the figure above, details of car with prices is showing to making a customer can decide the right choice of car depend on their budget.

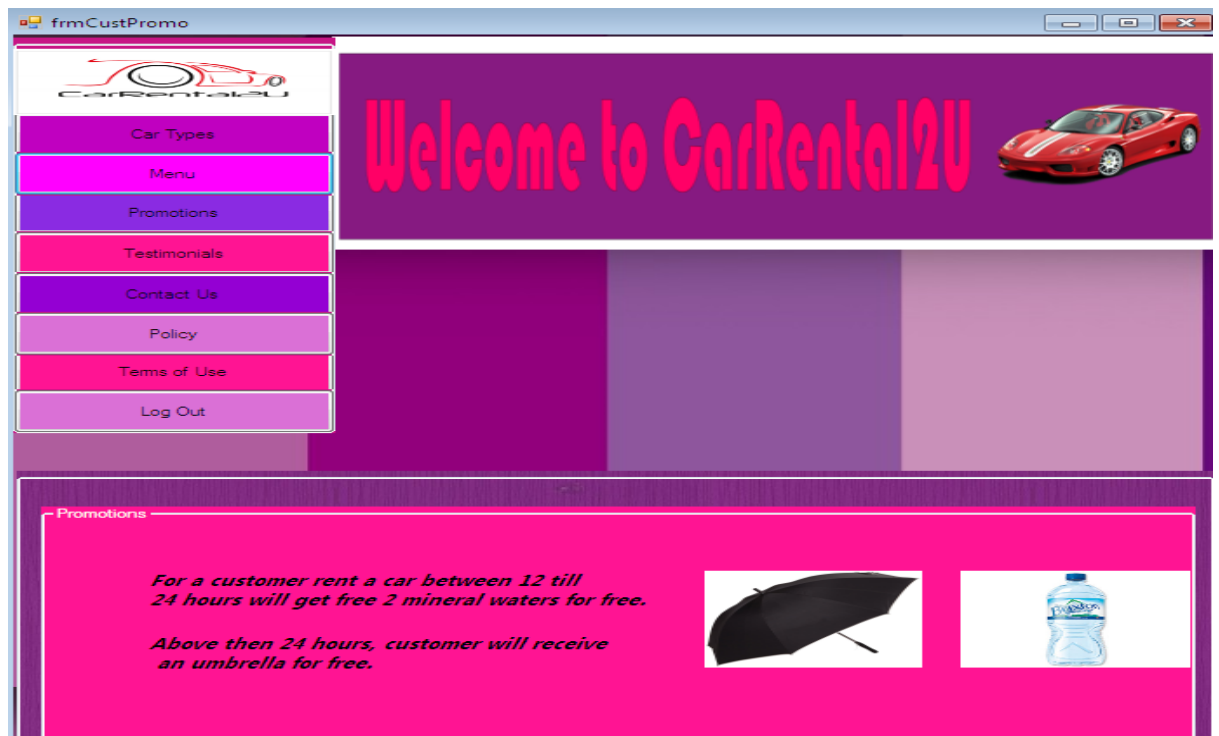


Figure 2.4.3(7) shows: Promotion interface

The main reason provides promotion form in this system to attract customer to become loyal customer. This is important to maintain company in stable condition and can compete with other car rental agency.



Figure 2.4.3(8) shows: Testimonials interface



Figure 2.4.3(9) shows: Contact interface

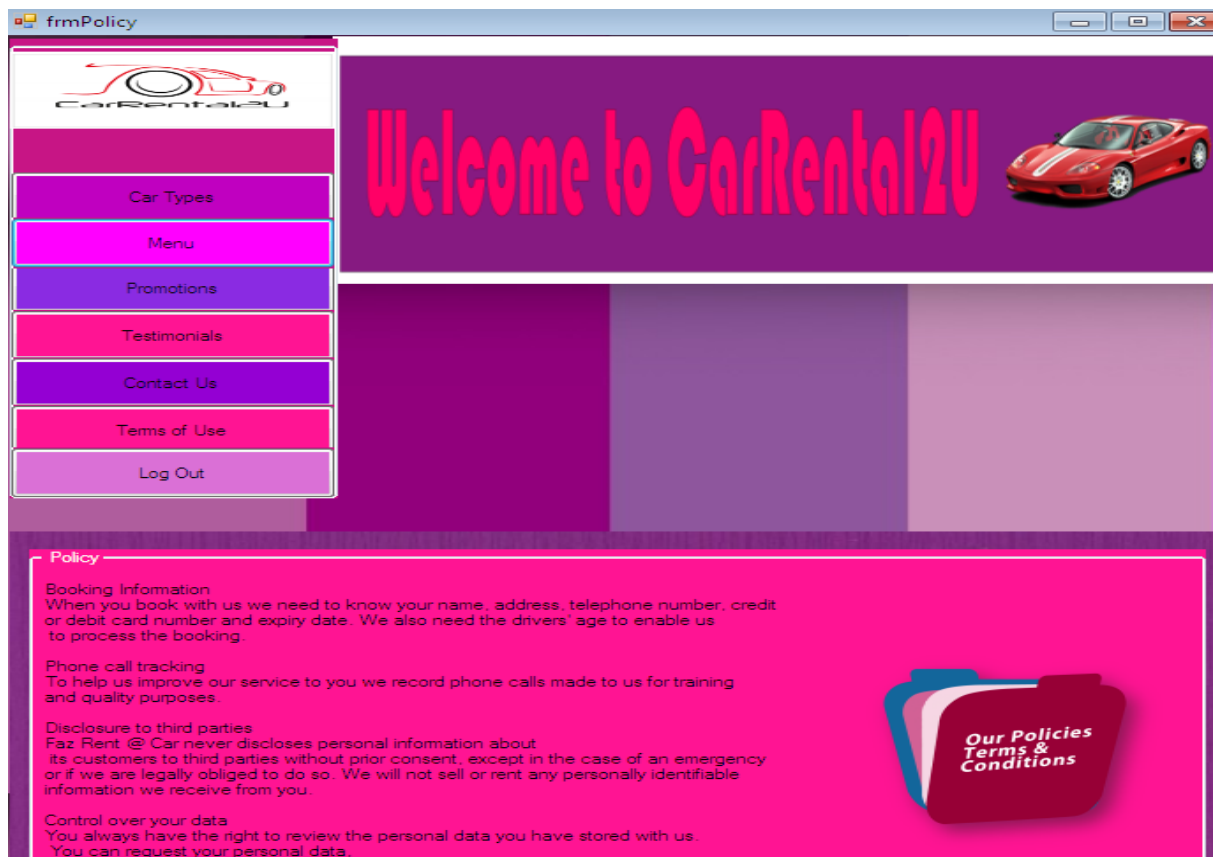


Figure 2.4.3(10) shows: Policy interface

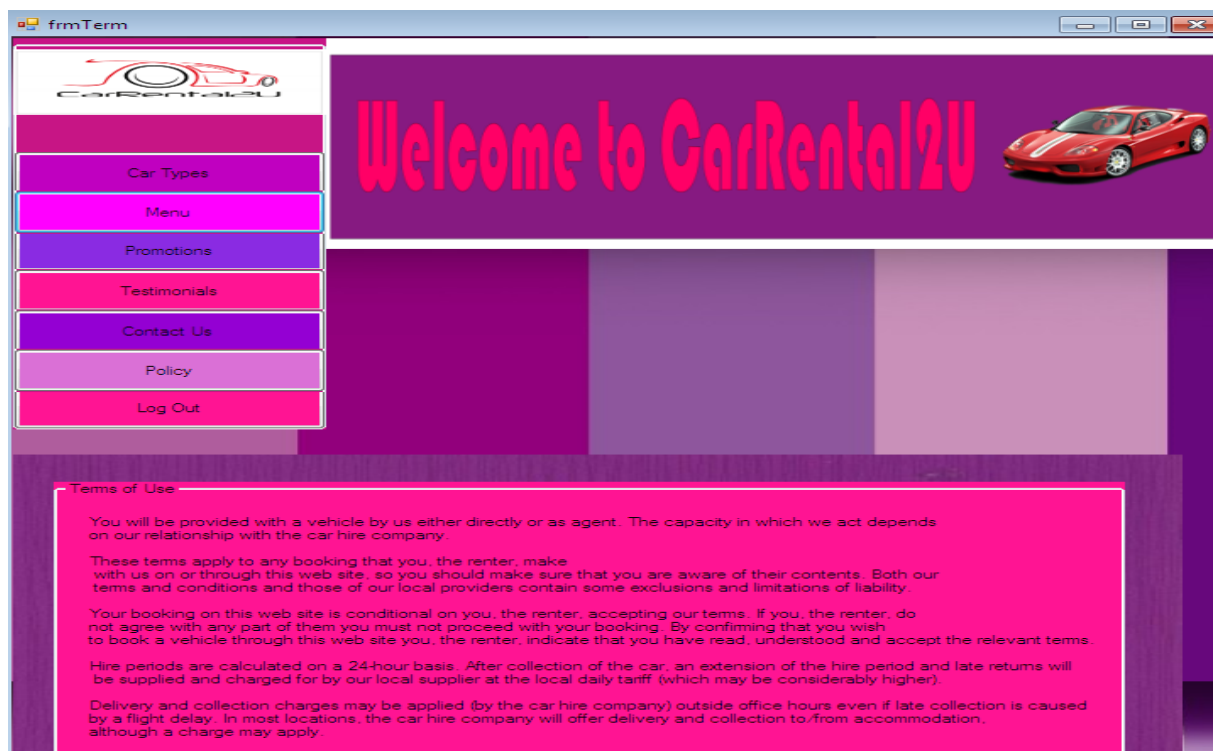


Figure 2.4.3(11) shows: Terms of use interface

Figure 2.4.3(12) shows: Return a car interface

By providing only one form, multifunction is embedded in this form such as checkout, generate reminder and send reminder. This is one of the important form where every customer will be frequent be remind to return a car to make sure car is returned on time.

Staff Name	staffGender	staff_IC_NO	PhoneNO	Email	staffAddress	staffID
adi	male	145800	0178737719	adi@gmail.com	no 13,ump,43300,....	a111
ALI	MALE	900058596695567	01234567890	ALI@GMAIL.COM	12,JLN UMP	AS112
ddddddddd	ccccccccccc		1111111111			
dddffffff	dddd	000000000000	3333333333	#####	ddddddddd	

Figure 2.4.3(13) shows: Database detail interface

The screenshot shows a web browser window titled "frmProfile". The header features the "CarRental2U" logo and a navigation menu with links: Car Types, Menu, Promotions, Testimonials, Contact Us, Policy, Terms of Use, and Log Out. The main content area displays a "Welcome to CarRental2U" banner with a red sports car image. Below the banner is the "Staff Profile" form for a user named "adi".

Staff Profile

Name: Address:

Gender:

IC No:

Handphone No:

Email Address:

Staff ID:

Password: Password should have 6-digits

A small icon of a person in a suit is visible next to the password field.

Figure 2.4.3(14) shows: Staff's profile interface

The screenshot shows a web browser window titled "Form1". The header features the "CarRental2U" logo and a navigation menu. The main content area displays a "Welcome to CarRental2U" banner with a red sports car image. Below the banner is a date display: "Friday , 20 December, 2013".

MEMBER: Sign in

Username:

Password:

A "CRMS" dialog box is open in the center of the screen, displaying "WELCOME TO THE SYSTEM" and an "OK" button.

Figure 2.4.3(15) shows: Member's login interface



Figure 2.4.3(16) shows: Member's menu interface

Figure 2.4.3(17) shows: Rent a car form interface

Member will be tracing the information first before able to proceed rent a car. This is because the hand phone number will be used to sending SMS reminder when rent a car.

frmProfile

CarRental2U

Car Types
Menu
Promotions
Testimonials
Contact Us
Policy
Terms of Use
Log Out

Welcome to CarRental2U

Member Profile

Name: siti Address: sdfgkjkh

Gender: female

IC No: 145000

Handphone No: 0178737719

Email Address: siti@yahoo.com

Member ID: m111

Password: 112233 Password should have 6-digits

Update Save

Figure 2.4.3(18) shows: Member's profile interface

Form1

Welcome to CarRental2U

Friday, 20 December, 2013

STAFF: Sign in

Username: adi

Password: *****

Category: ADMIN

Login

Forget password Gain Password Now

CRMS

ADMIN Additional Password :

654321

OK Cancel

Back

Figure 2.4.3(19) shows: Admin login interface

Additional admin password is compulsory for a staff who are admin because this is important for a security.

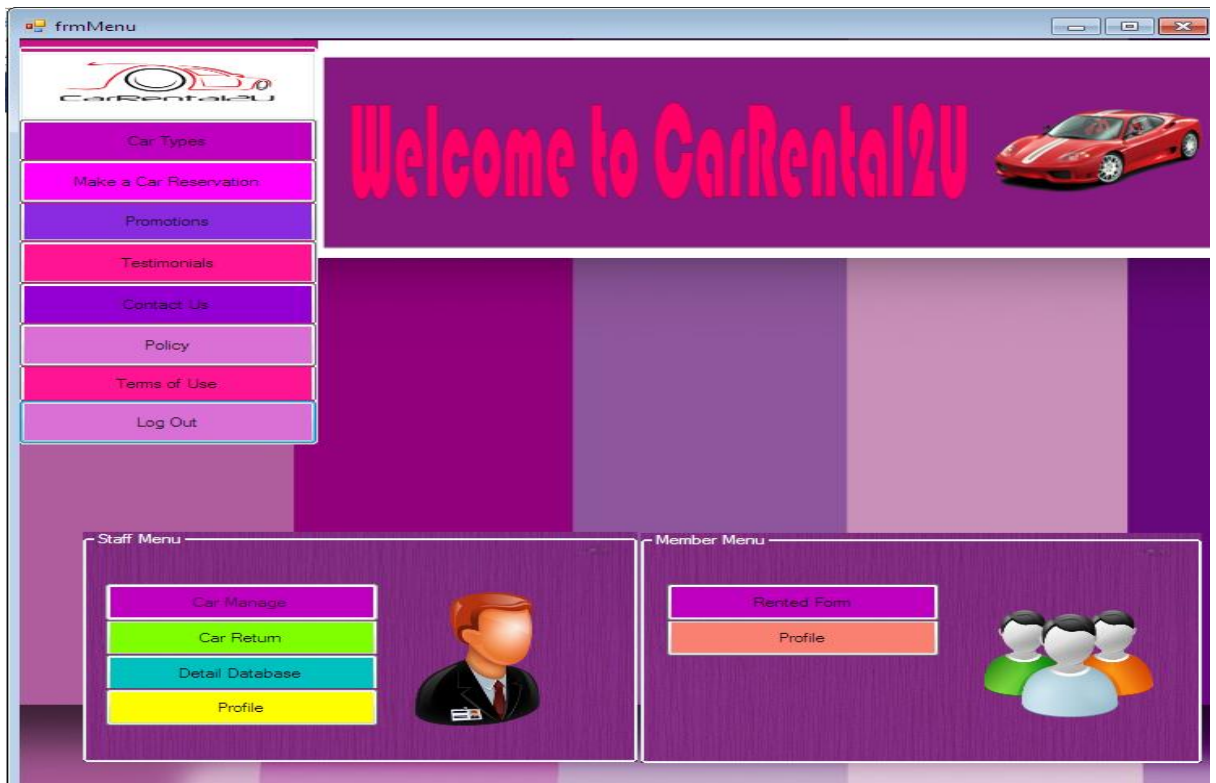


Figure 2.4.3(20) shows: Admin menu interface

Admin able to check all the function inside the system and only admin have a right to manage cars.

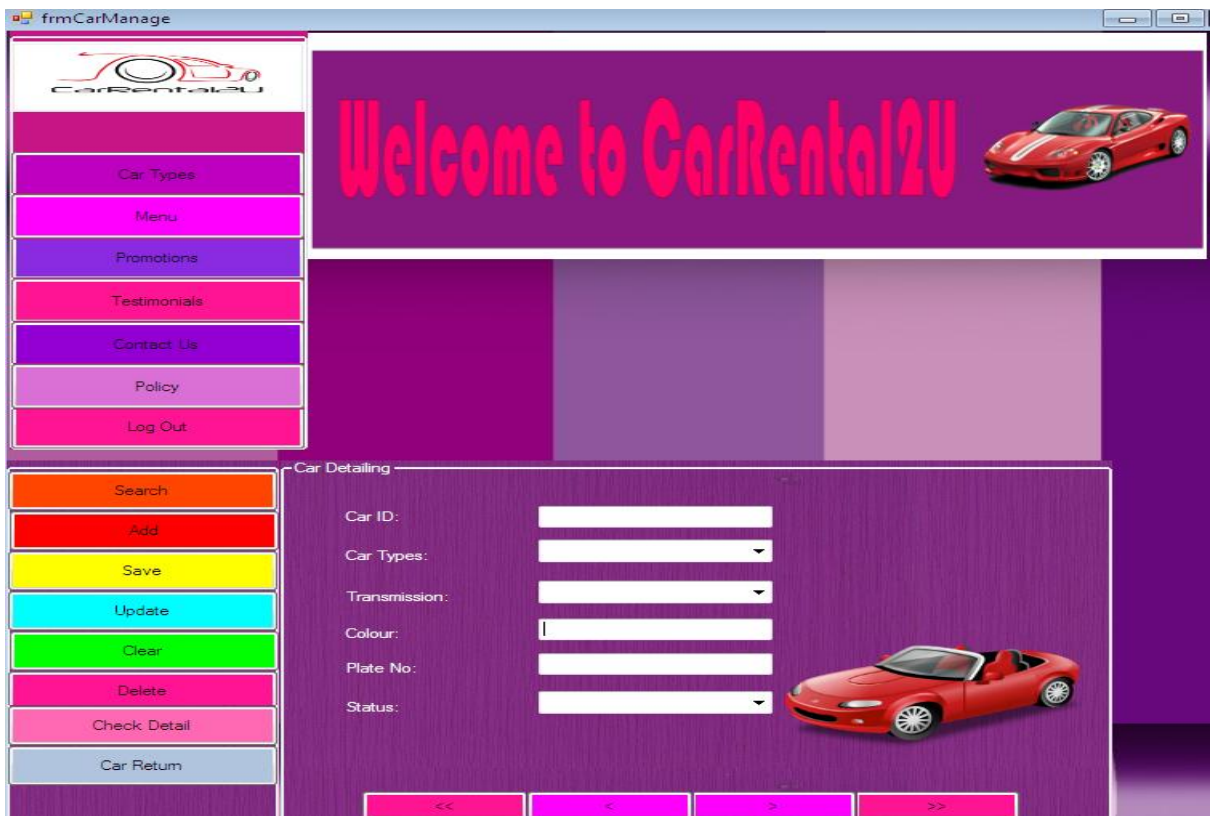


Figure 2.4.3(21) shows: Car managing interface

Car managing interface where all the record by adding, deleting or editing car lists inside a system manage by admin only.

2.5 input and Output Design

2.5.1 Input Design

Input interface is a data entry required by the system according to different functions meant to be kept in the database before generating the output interface as a result. The following is the input interfaces are available in this system as well as information for readers to better understand the process of data entry input.

Interface	Input	Input type
1. Login form	Entering name, category and password required	Input box
2. Menu form	Choose require form	Click on the selected button
3. Car Managing form	Add new information about rental car, edit or deleting previous car information	Insert input or click on selected button
4. Borrow car form	Should confirm first basic information in record before proceed key in data for rent a car	Insert details on input box
5. Return car form	Check the list borrower	Key in details on input box
6. Car list form	Listing all the database store in a system	Click on the selected button
7. Register/update user form	Insert the required data for users registrations process	Choosing user category
8. Profile form	Users able to	Showing all the

	review and update basic information	information regarding the user
9. Gain password form	User should enter IC no and name before automatic sent password to the hand phone.	Press on the button to obtain password

Figure 2.5.1 shows: Input design

2.5.2 Output design

The output interface is a key feedback received through the data in the input interface. The following is the output of the user interface rather than the result of filling in the feedback input.

Interface	Output	Output type
1. Login form	Confirmation identity of user	Message box
2. Menu form	The interface chosen is shown	Change to the form request
3. Car managing	Successfully update any changes	Message box appear if the process is successfully
4. Borrow car form	Car status will become not available	Auto date borrow shown and basic info after key in name
5. Return car form	Rental car changes become available and ready to be rent. SMS reminder is sent to borrower	Reminder message sent and message box appear when a car is returned
6. Car list form	Showing all the database that chosen	All the database shown in data grid view
7. Register/update user form	After complete register, user receive id for login	Message box appear shows user can proceed to login form
8. Profile form	Showing all the	User profile is listed

	information that key in during register before	in textbox
9. Gain password form	User login password	Message sent to user' hand phone

Figure 2.5.2 shows: Output design

2.5.3 Database Design

In this system, there are five categories of database according to their respective functions. For examples, tblUserName, tblMember, tblCustBook, tblCarDatabase and tblBorrowCar. The function of these databases is storing data updating by staff and customers in the system. These data will be linked to each other for matching true identity, the current status of rental vehicles and so on.

tblUserName	tblMember	tblCustBook	tblCarDatabase	tblBorrowCar			
StaffName	staffGender	staff_IC_NO	PhoneNO	Email	staffAddress	staffID	StaffPw
adi	male	145800	0178737719	adi@gmail.com	no13,ump,433	S111	123455
aida	female	123672	0114495590	aida@gmail.com	13,JLN 1, 42300	S444	123456
ali	male	156789	0123456789	ali@gmail.com	12,JLN UMP	S222	456787
aminah	female	103345	0130505966	aminah@yahoo	no11-1, jln 23b	S333	123476
sasa	female	118934	0134566788	sasa@yahoo.c	no1, jln tmn ri	S555	000000

Figure 2.5.3(1) shows: Staff database design (Primary key: StaffName)

tblUserName		tblMember		tblCustBook	tblCarDatabase	tblBorrowCar	
MemberID	MemberName	Gender	IC_NO	ContactNo	Email	Address	MemberPw
M222	siti	female	145800	0178737719	siti@gmail.com	11,jln34,tmn se	112233
M333	hani	female	112233	0133719649	hani@gmail.cc	12-4,jln kenany	123456
M444	ahmad	male	112222	3333333333	ahmad@gmail	43,lorong itik,	223335
M555	mad	male	678899	0113456777	mad@gmail.cc	no13,jln tamin	111111

Figure 2.5.3(2) shows: Member database design (Primary key: MemberName)

tblUserName		tblMember		tblCarDatabase		tblBorrowCar	
CarID	CarTypes	Transmissio	Colour	PlateNO	Status	Add New Field	
FZ123	PERODUA VIVA	MANUAL	GREEN	WWE5566	NOT AVAILABL		
FZ122	PROTON EZOR	AUTO	RED	BBM1145	NOT AVAILABL		
FZ124	PROTON SAGA	AUTO	BLACK	WWT1134	AVAILABLE		

Figure 2.5.3(3) shows: Car Database design (Primary key: CarID)

tblUserName		tblMember		tblCarDatabase		tblBorrowCar	
CarID	BorrowerName	BorrowDate	ReturnDate	PlateNo			
FZ122	siti	23/12/2013	25/12/2013	BBM 1145			
FZ123	ali	21/12/2013	23/12/2013	WWE 5566			

Figure 2.5.3(4) shows: Borrow Car design (Primary key: BorrowerName)

2.5.4 Conclusion

In this chapter, we are completely discussing about the function every form in the system. This is important for reader to know first the function before testing a system so user will not confuse the different form views inside the system of CRMSS.

2.6 Testing

For testing, the system will be run and all the process occur will showing clearly in this chapter. This is importance to test a system to makesure system is free error and able to stored in database.

2.6.1 Test data and descriptions

The figures below are running system with completely description of manual users.

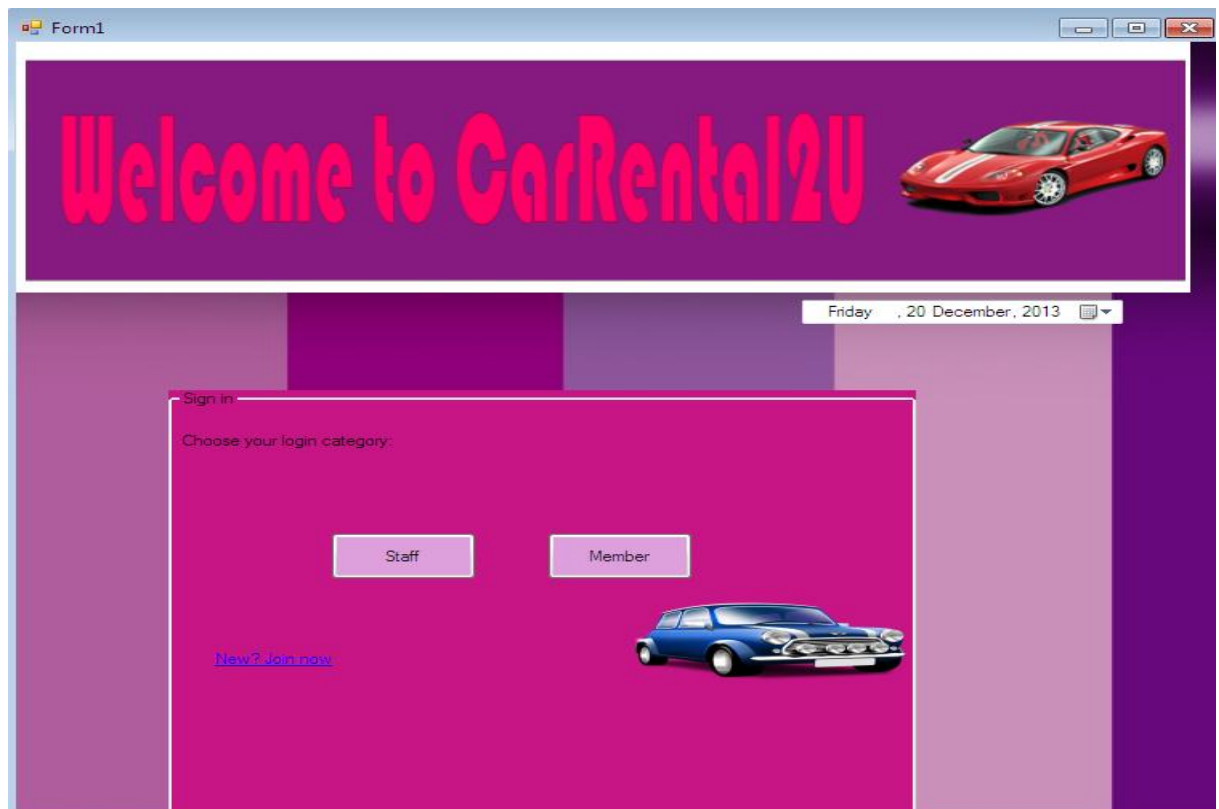


Figure 2.6.1(1) shows: Login form

First, user should select which category they are. If there are not user, then click join now button to proceed registration form.

The screenshot shows the 'Welcome to CarRental2U' website header with a red sports car. Below is a registration form titled 'Register:'. The form contains the following fields and labels:

- 'Your information will be submitted over a' (partially visible)
- '* Indicates Required Field' (asterisk icon)
- '* Name: liza' (text input)
- '* User ID: ' (text input)
- '* Gender: female' (text input)
- '* Handphone no: 0000000000' (text input)
- '* Email Address: (e.g faz@yahoo.com) dfgh@yahoo.com' (text input)
- '* IC No: 567777' (text input) with a label 'Insert last 6 digits IC No.'
- '* Address: ' (text input)

A modal dialog box titled 'CRMS' is overlaid on the form, containing the text 'Enter the Admin Password :', an 'OK' button, and a 'Cancel' button. The password field in the dialog contains '123456'.

Below the form are two buttons: 'Close' and 'Clear'.

At the bottom right, there is a section titled 'Please choose your login register category:' with two buttons: 'Staff' and 'Member'.

Figure 2.6.1(2) shows: Register form

For a member, registration will complete after key in the password to message box and able to sign in. But, for staff, they should key in admin password first before the registration process is completed.

The screenshot shows a modal dialog box titled 'CRMS'. The text inside the box reads: 'System administrator will assign a USER ID for you later. Please wait'. There is an 'OK' button at the bottom right of the dialog.

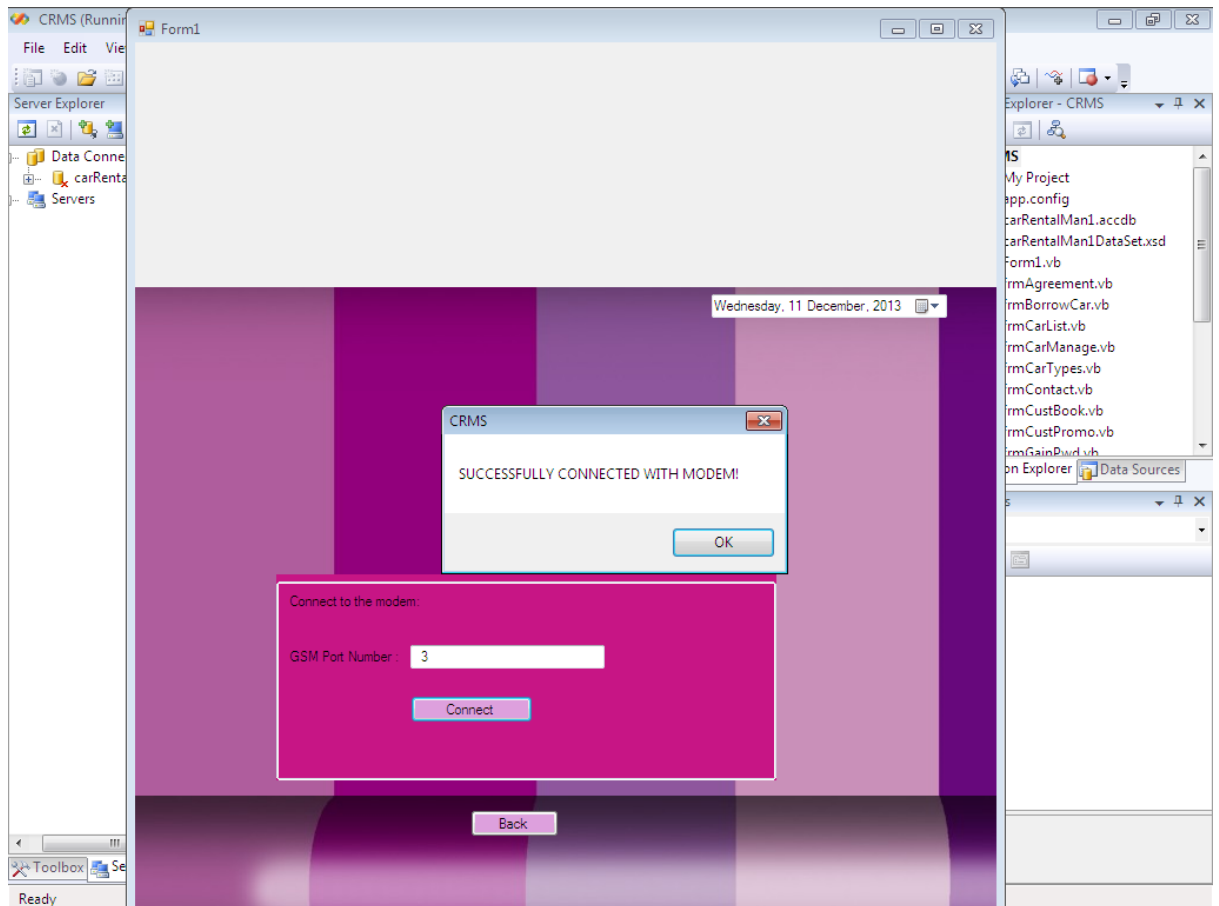


Figure 2.6.1(3) shows: Connect Modem Interface

Before staff start to use a system, they should key in no of port GSM Modem. Make sure the connection is successfully because if do not connect, it means that fail to sent SMS process.

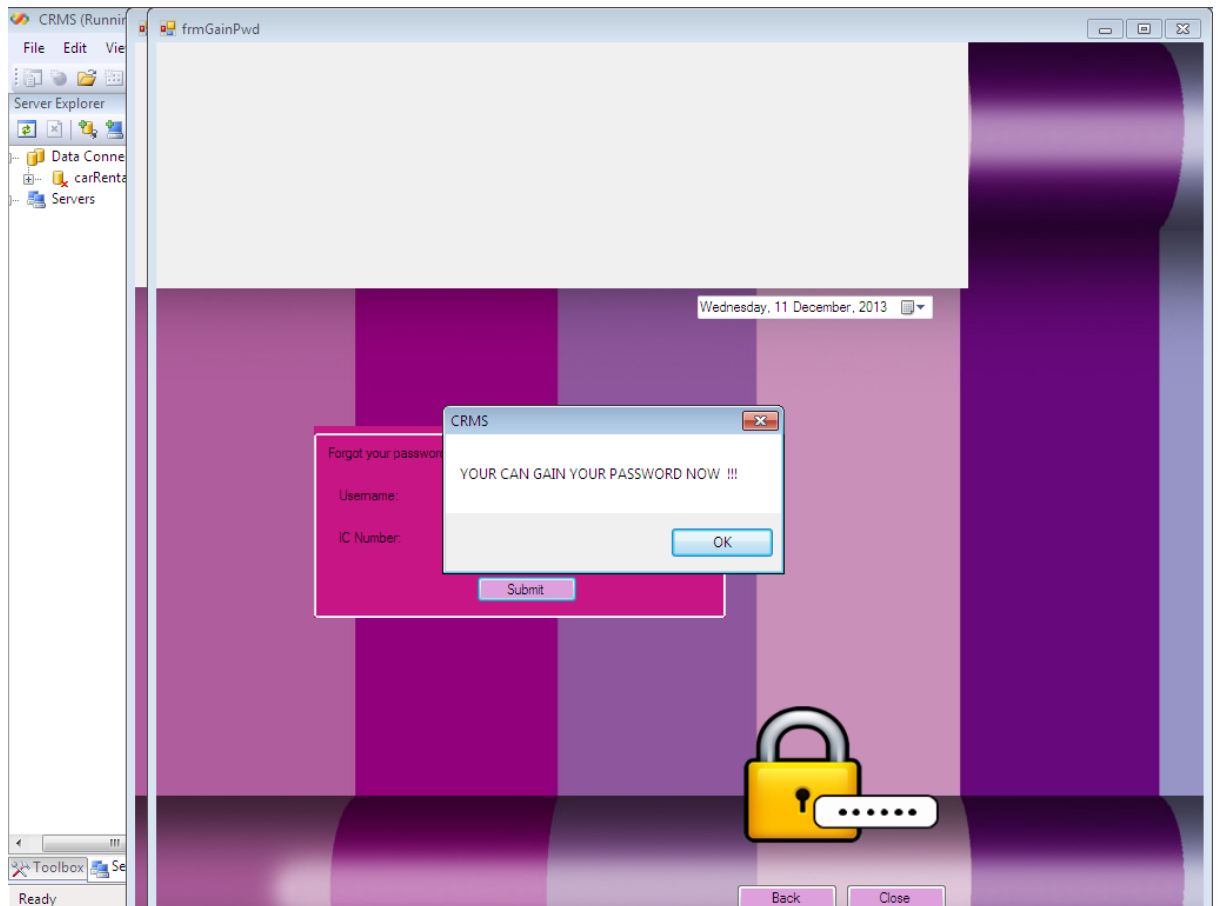


Figure 2.6.1(4) shows: Regain password form

If staff forgot a password, they should not worry because after inserting correct number port of modem, just insert name and IC no to regain password through SMS direct to owner's hand phone.

Form1

Welcome to CarRental2U

Friday, 20 December, 2013

STAFF: Sign in

Username: adi

Password: *****

Category: ADMIN

Login

Forget password Gain Password Now

CRMS

ADMIN Additional Password :

654321

OK Cancel

Back

Figure 2.6.1(5) shows: Admin login form

Staff who are become an admin should insert admin password also after key in their own password.



Figure 2.6.1(6) shows: Admin's menu form

Mostly, an admin is the company boss. So, admin have full right to control everything in the system.



Figure 2.6.1(7) shows: Staff menu form

Staff have four function that can be click such as car manage, car return, detail database and profile.

frmCarManage

CarRental2U

Welcome to CarRental2U

Car Types

Menu

Promotions

Testimonials

Contact Us

Policy

Log Out

Search

Add

Save

Update

Clear

Delete

Check Detail

Car Return

Car Detailing

Car ID:

Car Types:

Transmission:

Colour:

Plate No:

Status:

<< < > >>

Figure 2.6.1(8) shows: Car managing form

In this form, an admin are able to deleting, adding, or updating any changes to the car records.

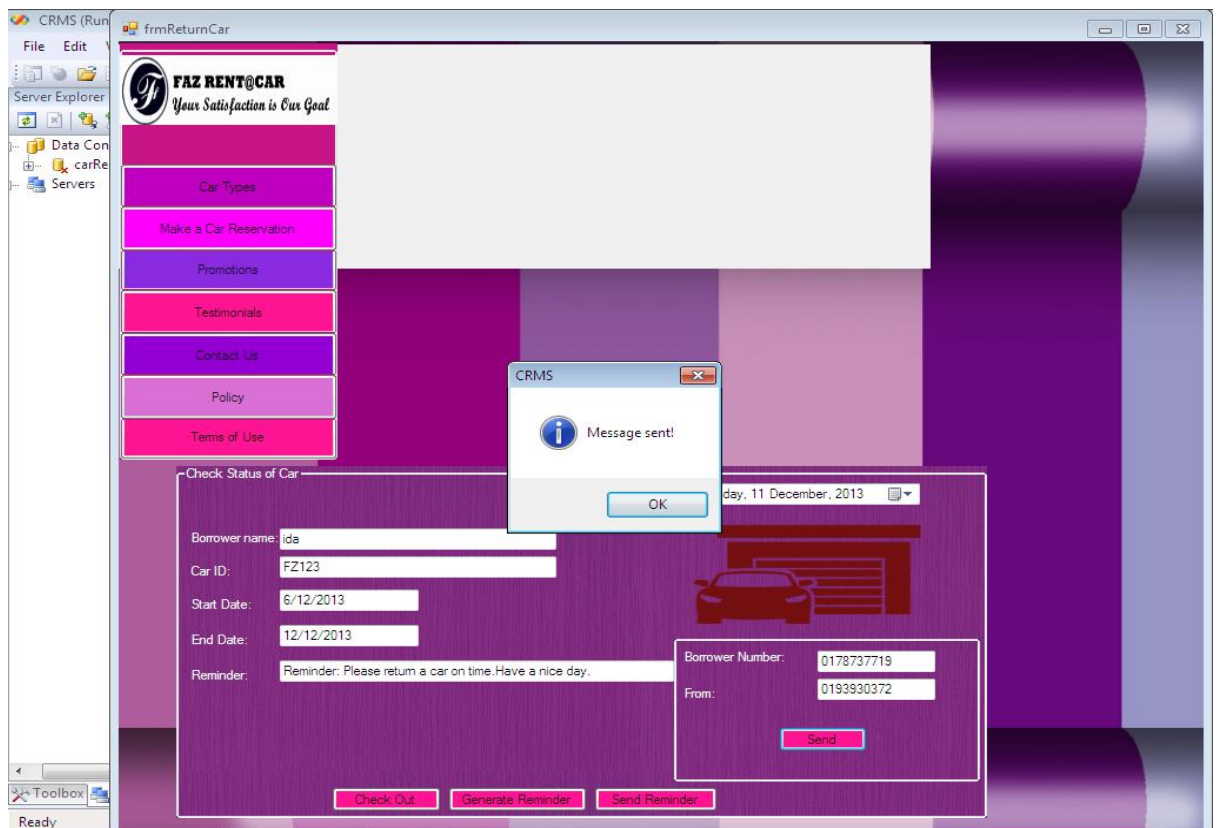


Figure 2.6.1(9) shows: Return car form

When user return a car, staff should check out a car to make sure car is on status “Available” and may rent by other customer. For a customer who still does not return a car, SMS is sending as a reminder to customer.

tblUserName	tblMember	tblCarDatabase	tblBorrowCar		
CarID	BorrowerName	BorrowDate	ReturnDate	PlateNo	Add New Field
#Deleted	#Deleted	#Deleted	#Deleted	#Deleted	
FZ123	ali	21/12/2013	23/12/2013	WWE 5566	

The screenshot displays the CarRental2U website interface. At the top left is the logo. A vertical menu on the left contains links: Car Types, Menu, Promotions, Testimonials, Contact Us, Policy, and Terms of Use. A large banner at the top right says 'Welcome to CarRental2U' with a red sports car image. Below the banner, a database table is visible, showing staff records. The table has columns: StaffName, staffGender, staff_IC_NO, PhoneNO, Email, staffAddress, and staffID. The data rows are as follows:

StaffName	staffGender	staff_IC_NO	PhoneNO	Email	staffAddress	staffID
adi	male	145800	0178737719	adi@gmail.com	no13,ump,43300,...	S111
aida	female	123672	0114495590	aida@gmail.com	13,JLN 1, 42300 ...	S444
ali	male	156789	0123456789	ali@gmail.com	12,JLN UMP	S222
aminah	female	103345	0130505966	aminah@yahoo.c...	no11-1, jln 23b, 4...	S333
sasa	female	118934	0134566788	sasa@yahoo.com	no1, jln tmn na, 4...	S555

Below the table, there is a large grey rectangular area. On the far left, a sidebar contains buttons: Manage Car Detail, Reload, Staff Detail, Member Detail, List Cars Not Available, Car Database Record, and Close. At the bottom left of the sidebar is a database icon.

Figure 2.6.1(10) shows: Car list form

In this form, staff and admin able to search all the record stored in database by clicking the button function at the left side.

The screenshot displays a web application interface for 'CarRental2U'. The top navigation bar includes links for Car Types, Menu, Promotions, Testimonials, Contact Us, Policy, Terms of Use, and Log Out. The main content area features a 'Welcome to CarRental2U' banner with a red sports car image. Below the banner is the 'Staff Profile' form, which contains the following fields and values:

Field	Value
Name	adi
Gender	male
IC No	145800
Handphone No	0178737719
Email Address	adi@gmail.com
Staff ID	111
Password	123455

The Address field is a large text area containing 'no 13, ump, 43300, selangor'. A note indicates 'Password should have 6-digits'. At the bottom of the form are 'Update' and 'Save' buttons. To the right of the form is a placeholder for a staff member's photo, showing a generic man icon. A 'CRMS' dialog box is open in the foreground, displaying the message 'Data updated' and an 'OK' button.

Figure 2.6.1(11) shows: Staff profile

If staff have moving new house or change contact number, this form able user to updating new info to make sure profile is always up to date. When click to save button, automatically all the record change is store in database.

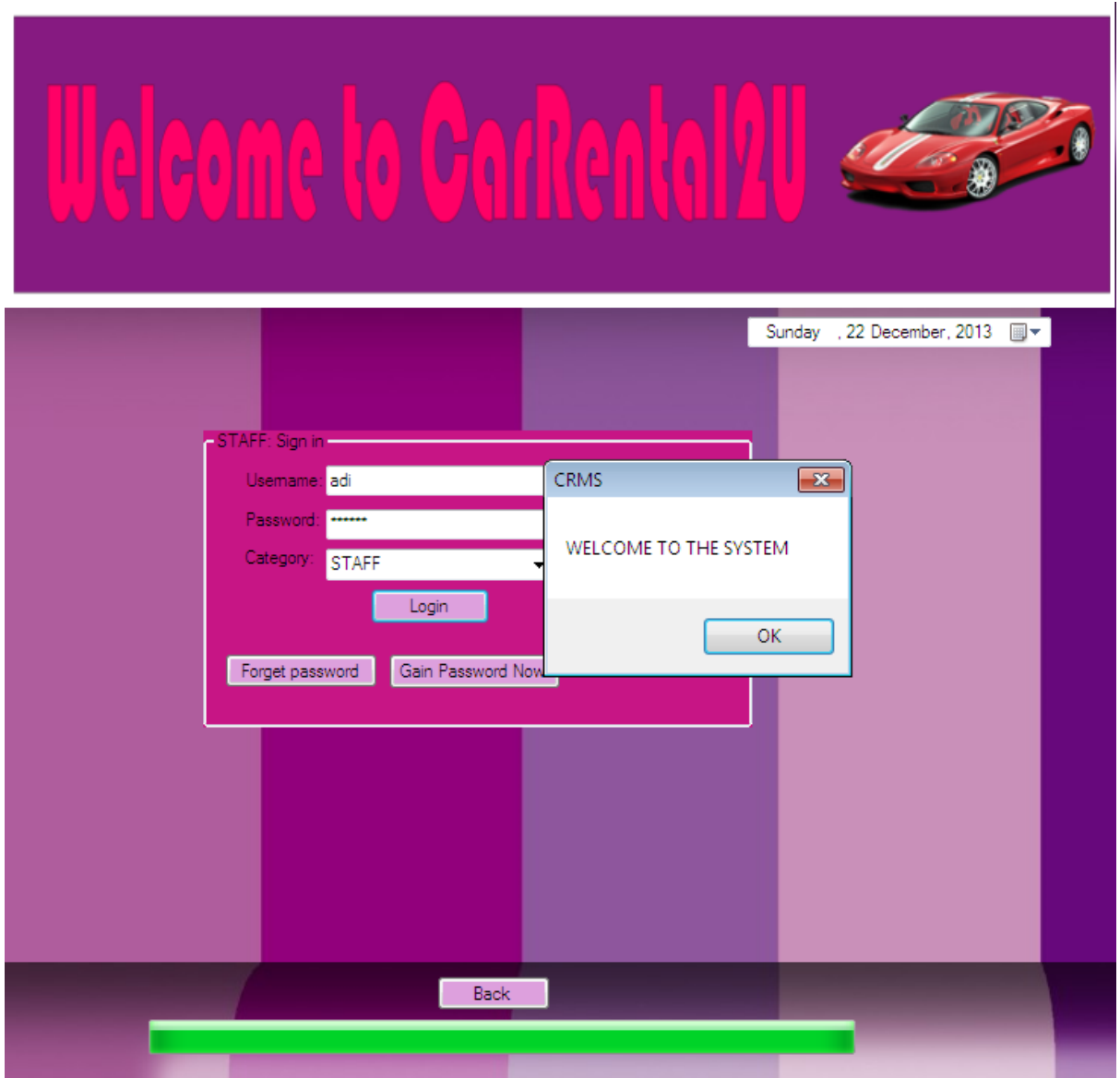


Figure 2.6.1(12) shows: Staff login form

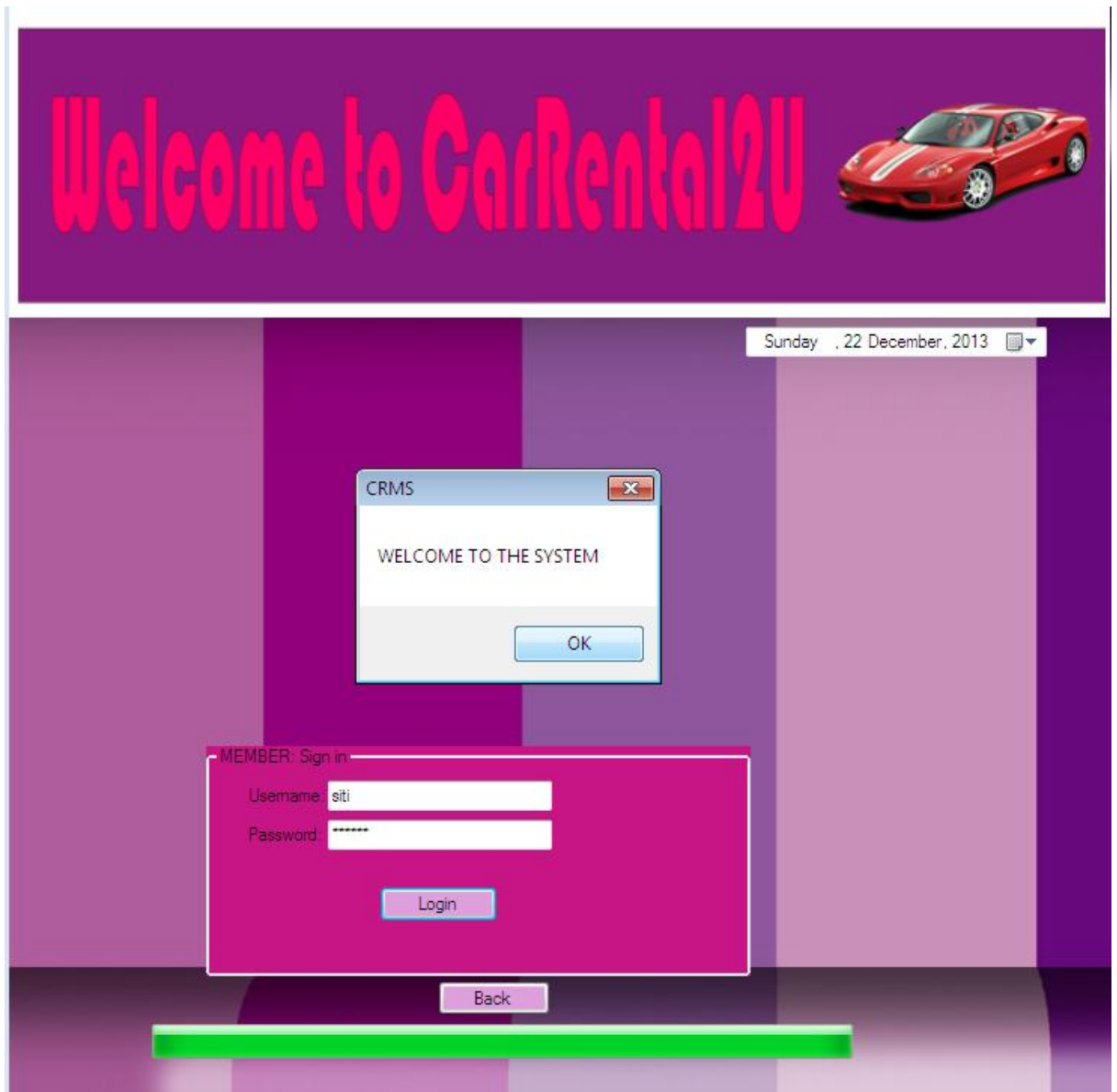


Figure 2.6.1(13) shows: Member login form



Figure 2.6.1(14) shows: Member menu form

Member's menu consists of rented form and profile form. If user want to rent a car, they need to click to rented form while changing profile record by clicking profile button.



Car Types

Menu

Promotions

Testimonials

Contact Us

Policy

Terms of Use

Log Out

Welcome to CarRental2U



CRMS

Car being rented

OK

Basic Information

Member name:

siti

IC No:

145800

Handphone no:

0178737719

Email Address:

siti@gmail.com

Search

Close



Car Rented Detail

Rent by:

siti

Car ID:

FZ122

Start Date:

23/12/2013


End Date:

25/12/13

Clear

Confirm

Monday , 23 December, 2013



CRMS

Your rental car Process is complete

OK

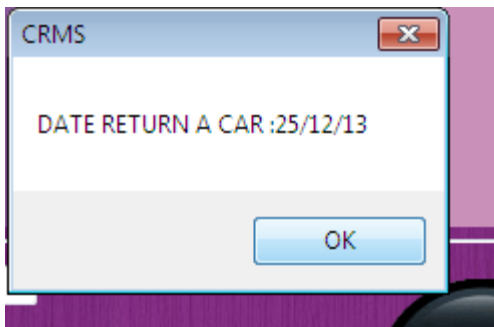


Figure 2.6.1(15) shows: Rental form

User need to entering name first for identification before proceed fill in date and car ID to rent a car. If rental process is successfully, it will proceed with receipt form after that.

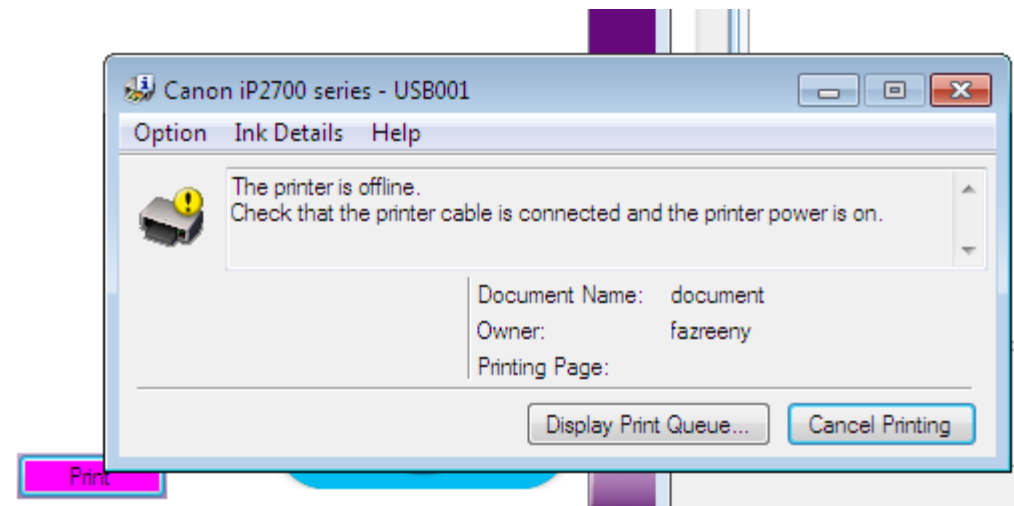


Figure 2.6.1(16) shows: Receipt form

User will click to print to proceed complete rental process. This is important to give information and reminder date a car should return back.

2.6.2 Conclusion

In this chapter, this system is successfully built as planning before. This is important to test a system to make sure it will run smoothly and can generate out interface as a feedback.

3.0 Conclusion

CRMs developed a system that facilitates the management of rental cars and admin staff in conducting their businesses. Besides, it also makes it easy to rent a vehicle for basic information customers are recorded when registering as a member. With the advent of such systems, it will facilitate staff and admin check vehicle status record, personal record keeping staff and customers as well as send SMS to the user to return rental car. For overall, develop system has to meet the overall objectives stated earlier. In conclusion, CRMSS are able to overcome the problems faced by staff and customers while using the manual system in the car rental business

3.1 Constraint

Due to the time frame for completing the project is relatively limited and operating system incompatible with visual basic 2008 is a major constraint for this project. However, managed to solved and the resulting system as planned.

3.2 Advantages

Along the development of Car Rental Management System with SMS Alert, there are few advantages of the system:

- a) Reduces human error
- b) Well organized information and database
- c) All the information is safe because only authorized people can access through this system.

3.3 Suggestion for future work

To further improve the quality and functionality CRMSS, improvements can be made in the future as well as diversify the function of a CRMSS applications as web-based system.

References:

1. Tudu, D. T. & Ramchandra Soren, R. (2011). *RFID BASED STUDENT DATABASE MANAGEMENT SYSTEM*. Department of Electronics and communication Engineering National Institute of Technology, Rourkela.
2. Yang, L., Cao, J. N., Zhu, W. P. , & Tang, S. J. (2012) A hybrid method for achieving high accuracy and efficiency in object tracking using passive RFID. *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on* , vol., no., pp.109-115.
3. Yujun Li , Hao Zheng , Zhiqiang Liu & Tengfei Yang. (2012). Design and Implementation of a Library Management System Based on the Web Service. Department of Computer Technology and Applications Qinghai University Xining, Qinghai, China.
4. A.T. Bin Abdullah., Ismail, I.B. Ibrahim, A.B., Hakim Bin Noor, M.Z.(2011). Library shelf management system using RFID technology. *System Engineering and Technology (ICSET), 2011 IEEE International Conference on*, vol., no., pp.215,218.

APPENDIX A

The Gant Chatt of the completion of my PSM 2.

Task/chapter	week1	week 2	week 3	week 4	week 5	week 6	week 7	week 8	week 9	week 10	week 11	week 12	week 13	week 14
2.4 Implementations														
2.4.1 Introduction to system interfaces														
2.4.2 Modem Configuration Management setup														
2.4.3 System Interfaces Using Visual Basic .Net														
2.5 Input and Output Design														
2.5.1 Input Design														
2.5.2 Output Design														
2.5.3 Database Design														
2.5.4 System Interface conclusion														
2.6 Testing														
2.6.1 Test data and descriptions														
2.6.2 Testing Conclusion														
3.0 Conclusion														
3.1 System weakness and Strengths														
3.1.1 Strength of the system														
3.1.2 Limitations of the system														
3.2 Suggestions for future work														
3.3 Significant contributions of the research														
3.4 Summary														

APPENDIX C

SYSTEM MANUAL & CODINGS

FORM LOGIN:

```
Public Class frmLogin
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim ds1, dsMlogin As New DataSet
    Dim da, daMlogin As OleDb.OleDbDataAdapter
    Dim sql, sqlMlogin As String

    Private Sub NavigateRecords()

        txtUserName.Text = ds1.Tables("UserName").Rows(inc).Item(0)
        mtxtPwd.Text = ds1.Tables("UserName").Rows(inc).Item(7)

    End Sub

    Private Sub btnGainPwd_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnGainPwd.Click
        If SMS.ConnectToGSM = True Then
            frmGainPwd.ShowDialog()
            '[VB - frmModem] frmSMS.Show(vbModal)
        Else
            MsgBox("GSM Modem is NOT connected with network!", vbCritical)
        End If
    End Sub

    Private Sub grpStaffMenu_Enter(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles grpStaffMenu.Enter

    End Sub

    Private Sub grpModem_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles grpModem.Enter

    End Sub

    Private Sub grpLoginChoose_Enter(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles grpLoginChoose.Enter

    End Sub

    Private Sub grpMember_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles grpMember.Enter

    End Sub
```

```

    Private Sub frmLogin_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblUserName' table. You can move, or remove it, as
needed.

Me.TblUserNameTableAdapter.Fill(Me.CarRentalMan1DataSet.tblUserName)
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblMember' table. You can move, or remove it, as
needed.
        Me.TblMemberTableAdapter.Fill(Me.CarRentalMan1DataSet.tblMember)

        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

        btnLogin.Enabled = True
        ProgressBar1.Hide()
        btnGainPwd.Visible = False

    End Sub

    Private Sub TblMemberBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        Me.Validate()
        Me.TblMemberBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

    End Sub

    Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
        ProgressBar1.Increment(100)
        If ProgressBar1.Value = ProgressBar1.Maximum Then
            Timer1.Stop()
            MsgBox("WELCOME TO THE SYSTEM")
            frmMenu.Show()

            Me.Hide()
            ProgressBar1.Hide()

        End If
    End Sub

    Private Sub btnLogin_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLogin.Click
        con.ConnectionString = dbProvider & dbSource
        con.Open()

        sql = "SELECT * FROM tblUserName"
        da = New OleDb.OleDbDataAdapter(sql, con)
        da.Fill(ds1, "UserName")

        con.Close()
        MaxRows = ds1.Tables("UserName").Rows.Count
        inc = -1

        da = New OleDb.OleDbDataAdapter(sql, con)

```

```

Dim StaffName As String
Dim StaffPwd As String
Dim PwdFound As Boolean = False
Dim StaffFound As Boolean = False
Dim AdminPwd As String

StaffName = txtUserName.Text
StaffPwd = mtxtPwd.Text

For i As Integer = 0 To MaxRows - 1
    If (CStr(ds1.Tables("UserName").Rows(i)("StaffName").ToString)
= StaffName And _
CStr(ds1.Tables("UserName").Rows(i)("StaffPwd").ToString) =
StaffPwd) And cobCategory.Text = "ADMIN" Then
        AdminPwd = InputBox("ADMIN Additional Password : ")
        If AdminPwd = "654321" Then
            Timer1.Start()

            Label3.Show()
            frmMenu.grpStaffMenu.Show()
            frmMenu.grpMemberMenu.Show()

            StaffFound = True
            PwdFound = True

            inc = i
            da.Dispose()

            'txtUserName.Text = ""
            ' mtxtPwd.Text = ""
        End If

    ElseIf
CStr(ds1.Tables("UserName").Rows(i)("StaffName").ToString) = StaffName And
-
CStr(ds1.Tables("UserName").Rows(i)("StaffPwd").ToString) =
StaffPwd And cobCategory.Text = "STAFF" Then

        Timer1.Start()
        ProgressBar1.Show()

        frmMenu.grpStaffMenu.Show()
        frmMenu.grpMemberMenu.Hide()

        frmMenu.btnList.Enabled = True
        frmMenu.btnManageCar.Enabled = False
        frmCarList.btnManage.Enabled = False
        StaffFound = True
        PwdFound = True

        inc = i
        da.Dispose()

        ' txtUserName.Text = ""

```

```

        ' mtxtPwd.Text = ""
    End If
Next

    If (Not StaffFound And Not PwdFound And cobCategory.Text <>
"ADMIN") Then
        MsgBox("Your are not an administrator", 0, "Not in the table")
    ElseIf (Not StaffFound And Not PwdFound And cobCategory.Text =
"ADMIN") Then
        MsgBox("CHECK YOUR SELECTION", 0, "Not in the table")
    ElseIf (Not StaffFound And Not PwdFound And cobCategory.Text <>
"STAFF") Then
        MsgBox("Your are not a member", 0, "Not in the table")
    ElseIf (Not StaffFound And Not PwdFound And cobCategory.Text =
"STAFF") Then
        MsgBox("CHECK YOUR SELECTION", 0, "Not in the table")
    End If
End Sub

Private Sub btnMLogin_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnMLogin.Click

    con.ConnectionString = dbProvider & dbSource
    con.Open()

    sqlMlogin = "SELECT * FROM tblMember"
    daMlogin = New OleDb.OleDbDataAdapter(sqlMlogin, con)
    daMlogin.Fill(dsMlogin, "MemberLogin")

    con.Close()
    MaxRows = dsMlogin.Tables("MemberLogin").Rows.Count
    inc = -1

    daMlogin = New OleDb.OleDbDataAdapter(sqlMlogin, con)

    Dim MemberName As String
    Dim MemberPwd As String
    Dim MemberPwdFound As Boolean = False
    Dim MemberFound As Boolean = False

    MemberName = txtMemName.Text
    MemberPwd = mtxtMemPwd.Text

    For i As Integer = 0 To MaxRows - 1
        If
(CStr(dsMlogin.Tables("MemberLogin").Rows(i) ("MemberName").ToString) =
MemberName And _

CStr(dsMlogin.Tables("MemberLogin").Rows(i) ("MemberPwd").ToString) =
MemberPwd) Then

            Timer1.Start()
            ProgressBar1.Show()

            frmMenu.grpStaffMenu.Hide()
            frmMenu.grpMemberMenu.Show()

            MemberFound = True

```



```

        MemberPwdFound = True

        inc = i
        daMlogin.Dispose()

        ' txtMemName.Text = ""
        ' mtxtMemPwd.Text = ""
    End If

Next

If (Not MemberFound And Not MemberPwdFound) Then
    MsgBox("CHECK YOUR NAME AND PASSWORD !", 0, "Not in the table")

End If
End Sub

Private Sub btnGet_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnConnect.Click
    If txtPortNo.Text = "" Then
        MsgBox("YOU FORGET TO INSERT THE PORT NUMBER OF GSM MODEM !!!")
    Else
        ModuleSMS_Setting.Initmodem()
    End If
End Sub

Private Sub btnForget_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnForget.Click
    btnGainPwd.Visible = True
End Sub

Private Sub lblJoin_LinkClicked(ByVal sender As System.Object, ByVal e
As System.Windows.Forms.LinkLabelLinkClickedEventArgs) Handles
lblJoin.LinkClicked
    Me.Hide()
    frmREGISTER_USER.Show()
End Sub

Private Sub btnLogMem_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLogMem.Click
    grpLoginChoose.Hide()
    grpMember.Show()
    grpStaffMenu.Hide()
    grpModem.Hide()

End Sub

Private Sub btnLogStaff_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLogStaff.Click
    grpLoginChoose.Hide()
    grpMember.Hide()
    grpStaffMenu.Hide()
    grpModem.Show()
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBack.Click
    grpLoginChoose.Show()
    grpMember.Hide()

```

```

        grpStaffMenu.Hide()
        grpModem.Hide()
    End Sub

    Private Sub txtPortNo_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtPortNo.TextChanged

    End Sub

    Private Sub ProgressBar1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ProgressBar1.Click

    End Sub
End Class

```

FORM AGREEMENT:

```

Public Class frmAgreement

    Private Sub PictureBox3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles PictureBox3.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub btnCarTypes_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCarTypes.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub btnCarReserve_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCarReserve.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub btnPromotions_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPromotions.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub btnTestimonials_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnTestimonials.Click
        Me.Hide()
        frmTestimonial.Show()

    End Sub

    Private Sub btnContact_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnContact.Click
        Me.Hide()
        frmContact.Show()

    End Sub

```

```

    Private Sub btnPolicy_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnPolicy.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub btnTerms_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnTerms.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub frmAgreement_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    End Sub

    Private Sub btnDisagree_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnDisagree.Click
        Me.Hide()
        frmLogin.Show()

    End Sub

    Private Sub btnAgree_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAgree.Click
        Me.Hide()
        frmREGISTER_USER.Show()

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()

    End Sub
End Class

```

FORM BORROWER CAR:

```
Public Class frmBorrowCar
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim dsMember, dsBorrowUpdate, dsborrowBook As New DataSet
    Dim daMember, daBorrowUpdate, daborrowBook As OleDb.OleDbDataAdapter
    Dim sql As String
    Dim OleDbCon1 As System.Data.OleDb.OleDbConnection

    Private Sub NavigateRecords()

        txtBorrower.Text =
dsborrowBook.Tables("BorrowCar").Rows(inc).Item(0)
        txtCarID.Text = dsborrowBook.Tables("BorrowCar").Rows(inc).Item(1)

    End Sub
    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()

    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()
    End Sub
End Class
```

```

End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Me.Hide()
    frmPolicy.Show()
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    Me.Hide()
    frmTerm.Show()
End Sub

Private Sub TblBorrowCarBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
    Me.Validate()
    Me.TblBorrowCarBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

End Sub

Private Sub frmBorrowCar_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    'TODO: This line of code loads data into the
    'CarRentalMan1DataSet.tblMember' table. You can move, or remove it, as
    needed.
    Me.TblMemberTableAdapter.Fill(Me.CarRentalMan1DataSet.tblMember)
    'TODO: This line of code loads data into the
    'CarRentalMan1DataSet.tblBorrowCar' table. You can move, or remove it, as
    needed.
    Me.TblBorrowCarTableAdapter.Fill(Me.CarRentalMan1DataSet.tblBorrowCar)

    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    grpBorrowCar.Hide()

    btnCarStatus.Visible = False

    btnPrintBorrow.Visible = False

End Sub

Private Sub btnSearchMember_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles btnSearchMember.Click
    grpBorrowCar.Show()

    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    con.ConnectionString = dbProvider & dbSource
    con.Open()
    sql = "SELECT * FROM tblMember"
    daMember = New OleDb.OleDbDataAdapter(sql, con)
    daMember.Fill(dsMember, "Member")

```

```

con.Close()
MaxRows = dsMember.Tables("Member").Rows.Count
daMember.Dispose()

Dim BorrowerName As String
Dim BorrowerNameFound As Boolean = False

BorrowerName = txtMemberName.Text

For i As Integer = 0 To MaxRows - 1
    If
CStr(dsMember.Tables("Member").Rows(i)("MemberName").ToString) =
BorrowerName Then

        BorrowerNameFound = True
        inc = i

        txtMemberName.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(1))
        lblICNO.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(3))
        lblContactNo.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(4))
        lblEmail.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(5))
        txtBorrower.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(1))
        grpBorrowCar.Show()

    End If
Next
If (Not BorrowerNameFound) Then
    MsgBox("You are NOT a member of this ", 0, "Not in the table")
End If
End Sub

Private Sub btnBorrow_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBorrow.Click
    Dim ReturnDate As String
    ReturnDate = InputBox("Insert the date return a car : ")
    txtReturnDate.Text = ReturnDate

    lblBorrowDate.Text = dtpBorrow.Value.Date

    con.ConnectionString = dbProvider & dbSource
    con.Open()

    sql = "SELECT * FROM tblBorrowCar"
    daborrowBook = New OleDb.OleDbDataAdapter(sql, con)
    daborrowBook.Fill(dsborrowBook, "BorrowCar")

    con.Close()
    MaxRows = dsborrowBook.Tables("BorrowCar").Rows.Count

    If inc <> -1 Then

        Dim cb As New OleDb.OleDbCommandBuilder(daborrowBook)

```

```

Dim dsNewRowBook As DataRow

dsNewRowBook = dsborrowBook.Tables("BorrowCar").NewRow()

If inc <> -1 Then
    Dim BorrowerTtile As String
    Dim BorrowerFound As Boolean = False

    BorrowerTtile = txtBorrower.Text
    For i As Integer = 0 To MaxRows - 1
        If
CStr(dsborrowBook.Tables("BorrowCar").Rows(i)("BorrowerName").ToString) =
BorrowerTtile Then
            BorrowerFound = True
            inc = i
            MsgBox("Please return the car that you rent
before", 0, "Return car before")
            grpBorrowCar.Hide()

        End If
    Next
    If (Not BorrowerFound) Then
        btnCarStatus.Visible = True
        btnCarStatus.PerformClick()

        dsNewRowBook.Item(0) = txtBorrower.Text
        dsNewRowBook.Item(1) = txtCarID.Text

        dsNewRowBook.Item(2) = dtpBorrow.Value.Date
        dsNewRowBook.Item(3) = txtReturnDate.Text
        dsborrowBook.Tables("BorrowCar").Rows.Add(dsNewRowBook)

        MaxRows = MaxRows + 1
        inc = 0
        dsborrowBook.Update(dsborrowBook, "BorrowCar")
        MsgBox("Your rental car Process is complete ")
        MsgBox("DATE RETURN A CAR :" & txtReturnDate.Text)

        btnPrintBorrow.Visible = True
        btnPrintBorrow.PerformClick()
    End If

    Else : MsgBox("No Data Have Been Added !!!")
    End If

End If

txtBorrower.Clear()
txtCarID.Clear()

txtReturnDate.Clear()
txtMemberName.Clear()
lblBorrowDate.Text = ""
lblICNO.Text = ""
LblContactNo.Text = ""
lblEmail.Text = ""

```

```

End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClear.Click
    txtMemberName.Text = ""
    lblICNO.Text = ""
    lblContactNo.Text = ""
    lblEmail.Text = ""
    txtCarID.Text = ""

    txtBorrower.Text = ""
End Sub

Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClose.Click
    Me.Hide()
    frmLogin.Close()

End Sub

Private Sub btnPrintBorrow_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnPrintBorrow.Click
    frmReceipt.Show()

    frmReceipt.ListBorrow.Items.Add("Date: " & dtpBorrow.Value)
    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("Borrower Name: " &
txtBorrower.Text)
    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("List of book(s) : ")
    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("Car ID: " & txtCarID.Text)

    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("Date of rent:" &
lblBorrowDate.Text)
    frmReceipt.ListBorrow.Items.Add("Date of return : " &
txtReturnDate.Text)
    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("Please kindly return the car on
time")
    frmReceipt.ListBorrow.Items.Add("Thank You, have a nice day.")
End Sub

Private Sub btnReturnCar_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)
    Me.Hide()
    frmReturnCar.ShowDialog()
End Sub

Private Sub btnCarStatus_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnCarStatus.Click
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    con.ConnectionString = dbProvider & dbSource
    con.Open()
    sql = "SELECT * FROM tblCarDatabase"
    daBorrowUpdate = New OleDb.OleDbDataAdapter(sql, con)
    daBorrowUpdate.Fill(dsBorrowUpdate, "CarDatabase")

```



```

con.Close()
MaxRows = dsBorrowUpdate.Tables("CarDatabase").Rows.Count

Dim book1Found As Boolean = False

For i As Integer = 0 To MaxRows - 1
    If
CStr(dsBorrowUpdate.Tables("CarDatabase").Rows(i)("CarID").ToString) =
txtCarID.Text Then
        book1Found = True
        inc = i

        Dim cb As New OleDb.OleDbCommandBuilder(daBorrowUpdate)
        If inc <> -1 Then
            dsBorrowUpdate.Tables("CarDatabase").Rows(inc).Item(5)
= "NOT AVAILABLE"
            daBorrowUpdate.Update(dsBorrowUpdate, "CarDatabase")
            MsgBox("Car being rented")
        End If
    End If

Next

If (Not book1Found) Then
    MsgBox("Cannot find the information that you request", 0, "Not
in the table")
End If

End Sub

Private Sub GroupBox2_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles GroupBox2.Enter

End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
    Me.Hide()
    frmLogin.Show()

    frmLogin.grpLoginChoose.Show()
End Sub
End Class

```

FORM BORROW CAR:

```
Public Class frmBorrowCar
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim dsMember, dsBorrowUpdate, dsborrowBook As New DataSet
    Dim daMember, daBorrowUpdate, daborrowBook As OleDb.OleDbDataAdapter
    Dim sql As String
    Dim OleDBCon1 As System.Data.OleDb.OleDbConnection

    Private Sub NavigateRecords()

        txtBorrower.Text =
dsborrowBook.Tables("BorrowCar").Rows(inc).Item(0)
        txtCarID.Text = dsborrowBook.Tables("BorrowCar").Rows(inc).Item(1)

    End Sub
    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()

    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()
    End Sub
End Class
```

```

End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Me.Hide()
    frmPolicy.Show()
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    Me.Hide()
    frmTerm.Show()
End Sub

Private Sub TblBorrowCarBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
    Me.Validate()
    Me.TblBorrowCarBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

End Sub

Private Sub frmBorrowCar_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    'TODO: This line of code loads data into the
    'CarRentalMan1DataSet.tblMember' table. You can move, or remove it, as
    needed.
    Me.TblMemberTableAdapter.Fill(Me.CarRentalMan1DataSet.tblMember)
    'TODO: This line of code loads data into the
    'CarRentalMan1DataSet.tblBorrowCar' table. You can move, or remove it, as
    needed.
    Me.TblBorrowCarTableAdapter.Fill(Me.CarRentalMan1DataSet.tblBorrowCar)

    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    grpBorrowCar.Hide()

    btnCarStatus.Visible = False

    btnPrintBorrow.Visible = False

End Sub

Private Sub btnSearchMember_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles btnSearchMember.Click
    grpBorrowCar.Show()

    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    con.ConnectionString = dbProvider & dbSource
    con.Open()
    sql = "SELECT * FROM tblMember"
    daMember = New OleDb.OleDbDataAdapter(sql, con)
    daMember.Fill(dsMember, "Member")

```

```

con.Close()
MaxRows = dsMember.Tables("Member").Rows.Count
daMember.Dispose()

Dim BorrowerName As String
Dim BorrowerNameFound As Boolean = False

BorrowerName = txtMemberName.Text

For i As Integer = 0 To MaxRows - 1
    If
CStr(dsMember.Tables("Member").Rows(i)("MemberName").ToString) =
BorrowerName Then

        BorrowerNameFound = True
        inc = i

        txtMemberName.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(1))
        lblICNO.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(3))
        lblContactNo.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(4))
        lblEmail.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(5))
        txtBorrower.Text =
CStr(dsMember.Tables("Member").Rows(inc).Item(1))
        grpBorrowCar.Show()

    End If
Next
If (Not BorrowerNameFound) Then
    MsgBox("You are NOT a member of this ", 0, "Not in the table")
End If
End Sub

Private Sub btnBorrow_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBorrow.Click
    Dim ReturnDate As String
    ReturnDate = InputBox("Insert the date return a car : ")
    txtReturnDate.Text = ReturnDate

    lblBorrowDate.Text = dtpBorrow.Value.Date

    con.ConnectionString = dbProvider & dbSource
    con.Open()

    sql = "SELECT * FROM tblBorrowCar"
    daborrowBook = New OleDb.OleDbDataAdapter(sql, con)
    daborrowBook.Fill(dsborrowBook, "BorrowCar")

    con.Close()
    MaxRows = dsborrowBook.Tables("BorrowCar").Rows.Count

    If inc <> -1 Then

        Dim cb As New OleDb.OleDbCommandBuilder(daborrowBook)

```

```

Dim dsNewRowBook As DataRow

dsNewRowBook = dsborrowBook.Tables("BorrowCar").NewRow()

If inc <> -1 Then
    Dim BorrowerTtile As String
    Dim BorrowerFound As Boolean = False

    BorrowerTtile = txtBorrower.Text
    For i As Integer = 0 To MaxRows - 1
        If
CStr(dsborrowBook.Tables("BorrowCar").Rows(i)("BorrowerName").ToString) =
BorrowerTtile Then
            BorrowerFound = True
            inc = i
            MsgBox("Please return the car that you rent
before", 0, "Return car before")
            grpBorrowCar.Hide()

        End If
    Next
    If (Not BorrowerFound) Then
        btnCarStatus.Visible = True
        btnCarStatus.PerformClick()

        dsNewRowBook.Item(0) = txtBorrower.Text
        dsNewRowBook.Item(1) = txtCarID.Text

        dsNewRowBook.Item(2) = dtpBorrow.Value.Date
        dsNewRowBook.Item(3) = txtReturnDate.Text
        dsborrowBook.Tables("BorrowCar").Rows.Add(dsNewRowBook)

        MaxRows = MaxRows + 1
        inc = 0
        dsborrowBook.Update(dsborrowBook, "BorrowCar")
        MsgBox("Your rental car Process is complete ")
        MsgBox("DATE RETURN A CAR :" & txtReturnDate.Text)

        btnPrintBorrow.Visible = True
        btnPrintBorrow.PerformClick()
    End If

    Else : MsgBox("No Data Have Been Added !!!")
    End If

End If

txtBorrower.Clear()
txtCarID.Clear()

txtReturnDate.Clear()
txtMemberName.Clear()
lblBorrowDate.Text = ""
lblICNO.Text = ""
LblContactNo.Text = ""
lblEmail.Text = ""

```

```

End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClear.Click
    txtMemberName.Text = ""
    lblICNO.Text = ""
    lblContactNo.Text = ""
    lblEmail.Text = ""
    txtCarID.Text = ""

    txtBorrower.Text = ""
End Sub

Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClose.Click
    Me.Hide()
    frmLogin.Close()

End Sub

Private Sub btnPrintBorrow_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnPrintBorrow.Click
    frmReceipt.Show()

    frmReceipt.ListBorrow.Items.Add("Date: " & dtpBorrow.Value)
    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("Borrower Name: " &
txtBorrower.Text)
    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("List of book(s) : ")
    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("Car ID: " & txtCarID.Text)

    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("Date of rent:" &
lblBorrowDate.Text)
    frmReceipt.ListBorrow.Items.Add("Date of return : " &
txtReturnDate.Text)
    frmReceipt.ListBorrow.Items.Add("")
    frmReceipt.ListBorrow.Items.Add("Please kindly return the car on
time")
    frmReceipt.ListBorrow.Items.Add("Thank You, have a nice day.")
End Sub

Private Sub btnReturnCar_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)
    Me.Hide()
    frmReturnCar.ShowDialog()
End Sub

Private Sub btnCarStatus_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnCarStatus.Click
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    con.ConnectionString = dbProvider & dbSource
    con.Open()
    sql = "SELECT * FROM tblCarDatabase"
    daBorrowUpdate = New OleDb.OleDbDataAdapter(sql, con)
    daBorrowUpdate.Fill(dsBorrowUpdate, "CarDatabase")

```

```

con.Close()
MaxRows = dsBorrowUpdate.Tables("CarDatabase").Rows.Count

Dim book1Found As Boolean = False

For i As Integer = 0 To MaxRows - 1
    If
CStr(dsBorrowUpdate.Tables("CarDatabase").Rows(i)("CarID").ToString) =
txtCarID.Text Then
        book1Found = True
        inc = i

        Dim cb As New OleDb.OleDbCommandBuilder(daBorrowUpdate)
        If inc <> -1 Then
            dsBorrowUpdate.Tables("CarDatabase").Rows(inc).Item(5)
= "NOT AVAILABLE"
            daBorrowUpdate.Update(dsBorrowUpdate, "CarDatabase")
            MsgBox("Car being rented")
        End If
    End If

Next

If (Not book1Found) Then
    MsgBox("Cannot find the information that you request", 0, "Not
in the table")
End If

End Sub

Private Sub GroupBox2_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles GroupBox2.Enter

End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
    Me.Hide()
    frmLogin.Show()

    frmLogin.grpLoginChoose.Show()
End Sub
End Class

```

FORM CAR LIST:

```
Public Class frmCarList
    Dim OleDBCon1 As System.Data.OleDb.OleDbConnection
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim daBook, daUser, daBorrow, daBorrowBook As OleDb.OleDbDataAdapter
    Dim DA1, DA2, DA3, DA4 As System.Data.OleDb.OleDbDataAdapter
    Dim DS1, DS2, DS3, DS4 As New DataSet()
    Dim sql1, sql2, sql3, sql4 As String
    Private Sub PictureBox1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox1.Click

        End Sub

    Private Sub PictureBox3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox3.Click

        End Sub

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

    End Sub
```



```

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub frmCarList_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblUserName' table. You can move, or remove it, as
needed.

Me.TblUserNameTableAdapter.Fill(Me.CarRentalMan1DataSet.tblUserName)
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblMember' table. You can move, or remove it, as
needed.
        Me.TblMemberTableAdapter.Fill(Me.CarRentalMan1DataSet.tblMember)
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblCarDatabase' table. You can move, or remove it, as
needed.

Me.TblCarDatabaseTableAdapter.Fill(Me.CarRentalMan1DataSet.tblCarDatabase)
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblBorrowCar' table. You can move, or remove it, as
needed.

Me.TblBorrowCarTableAdapter.Fill(Me.CarRentalMan1DataSet.tblBorrowCar)
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    End Sub

    Private Sub btnLoadCarList_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnLoadCarList.Click
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

        Dim SearchTitle As String
        Dim SearchFound As Boolean = False

        SearchTitle = InputBox("Enter the keyword to search for:")

        If SearchTitle = "cardatabase" Then

            SearchFound = True
            sql1 = "SELECT * FROM tblCarDatabase"
            OleDbCon1 = New System.Data.OleDb.OleDbConnection(dbProvider &
dbSource)
            DA1 = New System.Data.OleDb.OleDbDataAdapter(sql1, OleDbCon1)
            OleDbCon1.Open()
            DA1.Fill(DS1, "CarList")
            dgv1.DataSource = DS1.Tables(" CarList")

```

```

OleDbCon1.Close()

ElseIf SearchTitle = "username" Then

    SearchFound = True
    sql2 = "SELECT *FROM tblUserName"
    OleDbCon1 = New System.Data.OleDb.OleDbConnection(dbProvider &
dbSource)
    DA2 = New System.Data.OleDb.OleDbDataAdapter(sql2, OleDbCon1)
    OleDbCon1.Open()
    DA2.Fill(DS2, "UserList")
    dgv1.DataSource = DS2.Tables("UserList")
    OleDbCon1.Close()

ElseIf SearchTitle = "member" Then
    SearchFound = True
    sql3 = "SELECT * FROM tblMember"
    OleDbCon1 = New System.Data.OleDb.OleDbConnection(dbProvider &
dbSource)
    DA3 = New System.Data.OleDb.OleDbDataAdapter(sql3, OleDbCon1)
    OleDbCon1.Open()
    DA3.Fill(DS3, "MemberList")
    dgv1.DataSource = DS3.Tables("MemberList")
    OleDbCon1.Close()

ElseIf SearchTitle = "borrowcar" Then
    SearchFound = True
    sql4 = "SELECT * FROM tblBorrowCar"
    OleDbCon1 = New System.Data.OleDb.OleDbConnection(dbProvider &
dbSource)
    DA4 = New System.Data.OleDb.OleDbDataAdapter(sql4, OleDbCon1)
    OleDbCon1.Open()
    DA4.Fill(DS4, "BorrowCarList")
    dgv1.DataSource = DS4.Tables("BorrowCarList")
    OleDbCon1.Close()

Else : MsgBox("Cannot find the information that you request", 0,
"Not in the table")

End If

End Sub

Private Sub btnCar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCar.Click
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    con.ConnectionString = dbProvider & dbSource

    sql1 = ("SELECT * FROM tblCarDatabase")
    OleDbCon1 = New System.Data.OleDb.OleDbConnection(dbProvider &
dbSource)

    DA1 = New System.Data.OleDb.OleDbDataAdapter(sql1, OleDbCon1)

```

```

        If Not con.State = ConnectionState.Open Then
            con.Open()

        End If

        daBook = New OleDb.OleDbDataAdapter("SELECT * FROM tblCarDatabase
ORDER BY CarID", con)
        Dim dt1 As New DataTable
        daBook.Fill(dt1)
        Me.dgv1.DataSource = dt1

        con.Close()
    End Sub

    Private Sub btnUsername_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnUsername.Click
        con.ConnectionString = dbProvider & dbSource

        sql2 = ("SELECT * FROM tblUserName")
        OleDbCon1 = New System.Data.OleDb.OleDbConnection(dbProvider &
dbSource)

        DA2 = New System.Data.OleDb.OleDbDataAdapter(sql2, OleDbCon1)

        If Not con.State = ConnectionState.Open Then
            con.Open()

        End If

        daUser = New OleDb.OleDbDataAdapter("SELECT * FROM tblUserName
ORDER BY StaffName", con)
        Dim dt2 As New DataTable
        daUser.Fill(dt2)

        Me.dgv1.DataSource = dt2

        con.Close()
    End Sub

    Private Sub btnBorrower_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBorrower.Click
        con.ConnectionString = dbProvider & dbSource

        sql3 = ("SELECT * FROM tblMember")
        OleDbCon1 = New System.Data.OleDb.OleDbConnection(dbProvider &
dbSource)

        DA3 = New System.Data.OleDb.OleDbDataAdapter(sql3, OleDbCon1)

        If Not con.State = ConnectionState.Open Then
            con.Open()

        End If

        daBorrow = New OleDb.OleDbDataAdapter("SELECT * FROM tblMember
ORDER BY MemberName", con)

        Dim dt3 As New DataTable
        daBorrow.Fill(dt3)

```

```

        Me.dgv1.DataSource = dt3

        con.Close()
    End Sub

    Private Sub btnBorrowCar_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnBorrowCar.Click
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

        con.ConnectionString = dbProvider & dbSource

        sql4 = ("SELECT * FROM tblBorrowCar")
        OleDbCon1 = New System.Data.OleDb.OleDbConnection(dbProvider &
dbSource)

        DA4 = New System.Data.OleDb.OleDbDataAdapter(sql4, OleDbCon1)
        If Not con.State = ConnectionState.Open Then
            con.Open()

        End If

        daBorrowBook = New OleDb.OleDbDataAdapter("SELECT * FROM
tblBorrowCar ORDER BY BorrowerName", con)
        Dim dt4 As New DataTable
        daBorrowBook.Fill(dt4)
        Me.dgv1.DataSource = dt4

        con.Close()
    End Sub

    Private Sub btnPrintData_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)

    End Sub

    Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()
    End Sub

    Private Sub btnManage_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnManage.Click
        frmCarManage.Show()
        Me.Hide()

    End Sub
End Class

```

FORM CAR MANAGE

```
Public Class frmCarManage
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim dsb As New DataSet
    Dim dab As OleDb.OleDbDataAdapter
    Dim sql As String
    Private Sub NavigateRecords()

        txtCarID.Text = dsb.Tables("CarDatabase").Rows(inc).Item(0)
        cobCarTypes.Text = dsb.Tables("CarDatabase").Rows(inc).Item(1)
        cobTransmission.Text = dsb.Tables("CarDatabase").Rows(inc).Item(2)
        txtColor.Text = dsb.Tables("CarDatabase").Rows(inc).Item(3)
        txtPlatNo.Text = dsb.Tables("CarDatabase").Rows(inc).Item(4)
        cobStatus.Text = dsb.Tables("CarDatabase").Rows(inc).Item(5)

    End Sub
    Private Sub txtPlatNo_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtPlatNo.TextChanged

    End Sub

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button5.Click
```

```

        Me.Hide()
        frmContact.Show()

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()
    End Sub

    Private Sub frmCarManage_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the
        'CarRentalMan1DataSet.tblCarDatabase' table. You can move, or remove it, as
        needed.

        Me.TblCarDatabaseTableAdapter.Fill(Me.CarRentalMan1DataSet.tblCarDatabase)
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

        con.ConnectionString = dbProvider & dbSource
        con.Open()

        ' Because we want to SELECT all (*) the records from the table
        called tblBookDatabase,
        'we pass this string to the string variable we have called sql:
        sql = "SELECT * FROM tblCarDatabase"

        ' Data Adaptor is doing: Acting as a go-between for the Connection
        Object and the Data Set
        dab = New OleDb.OleDbDataAdapter(sql, con)

        'The Data Adapter can Fill a DataSet with records from a Table.
        dab.Fill(dsb, "CarDatabase")
        'The DataSet (ds) filled with the records selected from the table
        called tblBookDatabase.

        con.Close()

        'You can get how many rows are in the DataSet with this:
        MaxRows = dsb.Tables("CarDatabase").Rows.Count
        inc = -1
    End Sub

    Private Sub TblCarDatabaseBindingNavigatorSaveItem_Click(ByVal sender
As System.Object, ByVal e As System.EventArgs)
        Me.Validate()
        Me.TblCarDatabaseBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

    End Sub

```

```

Private Sub btnFirst_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnFirst.Click
    If inc <> 0 Then
        inc = 0
        NavigateRecords()

    End If
End Sub

Private Sub btnPrevious_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnPrevious.Click
    If inc > 0 Then
        inc = inc - 1
        NavigateRecords()

    ElseIf inc = -1 Then
        MsgBox("No Records Yet")
    ElseIf inc = 0 Then
        MsgBox("First Record")

    End If
End Sub

Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNext.Click
    If inc <> MaxRows - 1 Then
        inc = inc + 1
        NavigateRecords()

    Else
        MsgBox("No More Rows")

    End If
End Sub

Private Sub btnLast_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLast.Click
    If inc <> MaxRows - 1 Then
        inc = MaxRows - 1
        NavigateRecords()

    End If
End Sub

Private Sub btnSearch_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSearch.Click
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    sql = "SELECT * FROM tblCarDatabase"
    dab = New OleDb.OleDbDataAdapter(sql, con)
    dab.Fill(dsb, "CarDatabase")
    dab.Dispose()

    Dim CarID As String
    Dim carFound As Boolean = False

    CarID = InputBox("Enter the car types to search for.")

```

```

        For i As Integer = 0 To MaxRows - 1
            If CStr(dsb.Tables("CarDatabase").Rows(i)("CarID").ToString) =
CarID Then

                carFound = True

                inc = i

                'The Tables property needs something in between round
brackets.
                ' Quite bizarrely, this is NOT the name of your database
table!
                'It 's that identifier you used with the Data Adapter Fill.

                txtCarID.Text =
CStr(dsb.Tables("CarDatabase").Rows(inc).Item(0))
                cobCarTypes.Text =
CStr(dsb.Tables("CarDatabase").Rows(inc).Item(1))
                cobTransmission.Text =
CStr(dsb.Tables("CarDatabase").Rows(inc).Item(2))
                txtColor.Text =
CStr(dsb.Tables("CarDatabase").Rows(inc).Item(3))
                txtPlatNo.Text =
CStr(dsb.Tables("CarDatabase").Rows(inc).Item(4))
                cobStatus.Text =
CStr(dsb.Tables("CarDatabase").Rows(inc).Item(5))
            End If
        Next
        If (Not carFound) Then
            MsgBox("Cannot find the information that you request", 0, "Not
in the table")
        End If
    End Sub

    Private Sub btnAdd_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAdd.Click
        btnSave.Enabled = True
        btnAdd.Enabled = False
        btnUpdate.Enabled = False
        btnDelete.Enabled = False

        txtCarID.Clear()
        cobCarTypes.Text = ""
        cobTransmission.Text = ""
        txtColor.Clear()
        txtPlatNo.Clear()
        cobStatus.Text = ""
    End Sub

    Private Sub btnSave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSave.Click
        con.ConnectionString = dbProvider & dbSource
        con.Open()

        sql = "SELECT * FROM tblCarDatabase"
        dab = New OleDb.OleDbDataAdapter(sql, con)
        dab.Fill(dsb, "CarDatabase")
        MaxRows = dsb.Tables("CarDatabase").Rows.Count

        con.Close()

```



```

If inc <> -1 Then

    Dim cb As New OleDb.OleDbCommandBuilder(dab)
    Dim dsNewRow As DataRow

    dsNewRow = dsb.Tables("CarDatabase").NewRow()

    dsNewRow.Item(0) = txtCarID.Text
    dsNewRow.Item(1) = cobCarTypes.Text
    dsNewRow.Item(2) = cobTransmission.Text
    dsNewRow.Item(3) = txtColor.Text
    dsNewRow.Item(4) = txtPlatNo.Text
    dsNewRow.Item(5) = cobStatus.Text

    dsb.Tables("CarDatabase").Rows.Add(dsNewRow)

    dab.Update(dsb, "CarDatabase")
    NavigateRecords()

    MaxRows = MaxRows + 1
    inc = 0

    MsgBox("New Record added to the CarDatabase")

    btnSave.Enabled = False
    btnAdd.Enabled = True
    btnUpdate.Enabled = True
    btnDelete.Enabled = True

Else : MsgBox("No Data Have Been Added")
End If

End Sub

Private Sub btnUpdate_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnUpdate.Click
    Dim cb As New OleDb.OleDbCommandBuilder(dab)

    If inc <> -1 Then

        dsb.Tables("CarDatabase").Rows(inc).Item(0) = txtCarID.Text
        dsb.Tables("CarDatabase").Rows(inc).Item(1) = cobCarTypes.Text
        dsb.Tables("CarDatabase").Rows(inc).Item(2) =
cobTransmission.Text
        dsb.Tables("CarDatabase").Rows(inc).Item(3) = txtColor.Text
        dsb.Tables("CarDatabase").Rows(inc).Item(4) = txtPlatNo.Text
        dsb.Tables("CarDatabase").Rows(inc).Item(5) = cobStatus.Text
        dab.Update(dsb, "CarDatabase")

        MsgBox("Data updated")

        NavigateRecords()

    End If
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClear.Click
    btnSave.Enabled = False
    btnAdd.Enabled = True

```

```

        btnUpdate.Enabled = True
        btnDelete.Enabled = True

        inc = 0
        NavigateRecords()
    End Sub

    Private Sub btnDelete_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnDelete.Click
        btnSave.Enabled = False
        btnAdd.Enabled = True
        btnUpdate.Enabled = True
        btnDelete.Enabled = True

        inc = 0
        NavigateRecords()
    End Sub

    Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
        Me.Close()
        frmLogin.Close()
    End Sub

    Private Sub btnCarList_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCarList.Click
        Me.Hide()
        frmCarList.Show()
    End Sub

    Private Sub btnCarReturn_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnCarReturn.Click
        Me.Hide()
        frmReturnCar.Show()
    End Sub

    Private Sub GroupBox2_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles GroupBox2.Enter

    End Sub
End Class

```

FORM CAR TYPEI:

```
Public Class frmCarTypes

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button7.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()
    End Sub

    Private Sub GroupBox2_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles GroupBox2.Enter

    End Sub
End Class
```


FORM CONTACT:

```
Public Class frmContact

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()
    End Sub
```

```

    Private Sub frmContact_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    End Sub
End Class

```

FORM CUSTOMER PROMOTIONS:

```
Public Class frmCustPromo
```

```

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

```

```

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

```

```

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

```

```

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click

    End Sub

```

```

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

```

```

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

    End Sub

```

```

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

```

```

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmTerm.Show()

```

```

End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
    Me.Hide()
    frmLogin.Show()

    frmLogin.grpLoginChoose.Show()
End Sub

Private Sub frmCustPromo_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

End Sub
End Class

```

FORM GAIN PASSWORD:

```

Public Class frmGainPwd
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim ds1 As New DataSet
    Dim daG As OleDb.OleDbDataAdapter
    Dim sql As String
    Private Sub NavigateRecords()

        txtStaffName.Text = ds1.Tables("UserName").Rows(inc).Item(1)
        txtMsg.Text = ds1.Tables("UserName").Rows(inc).Item(2)
        mtxtIC.Text = ds1.Tables("UserName").Rows(inc).Item(3)
        txtTo.Text = ds1.Tables("UserName").Rows(inc).Item(4)

    End Sub
    Private Sub TblUserNameBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        Me.Validate()
        Me.TblUserNameBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

    End Sub

    Private Sub frmGainPwd_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the
        'CarRentalMan1DataSet.tblUserName' table. You can move, or remove it, as
        needed.

        Me.TblUserNameTableAdapter.Fill(Me.CarRentalMan1DataSet.tblUserName)
        grpRegainPWD.Hide()

        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

        con.ConnectionString = dbProvider & dbSource
        con.Open()

```

```

        sql = "SELECT * FROM tblUserName"
        daG = New OleDb.OleDbDataAdapter(sql, con)
        daG.Fill(ds1, "UserName")

        con.Close()
        MaxRows = ds1.Tables("UserName").Rows.Count
        inc = -1
    End Sub

    Private Sub btnGET_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnGET.Click
        daG = New OleDb.OleDbDataAdapter(sql, con)

        Dim StaffName As String
        Dim GainPwd As String
        Dim NameFound As Boolean = False
        Dim GainPwdFound As Boolean = False

        StaffName = txtStaffName.Text
        GainPwd = mtxtIC.Text

        For i As Integer = 0 To MaxRows - 1
            If (CStr(ds1.Tables("UserName").Rows(i)("StaffName").ToString)
= StaffName And _
CStr(ds1.Tables("UserName").Rows(i)("Staff_IC_NO").ToString) = GainPwd)
Then

                NameFound = True
                GainPwdFound = True
                inc = i
                daG.Dispose()
                MsgBox("YOUR CAN GAIN YOUR PASSWORD NOW !!!")
                txtStaffName.Text = ""
                mtxtIC.Text = ""

                txtMsg.Text =
CStr(ds1.Tables("UserName").Rows(inc).Item(7)) & " , is your password. Now,
you can log in. Thank you ; " &
CStr(ds1.Tables("UserName").Rows(inc).Item(2))
                txtTo.Text = CStr(ds1.Tables("UserName").Rows(inc).Item(3))
                txtFrom.Text = "0193930372"
                '-----

                '[VB - frmSMS] Private Sub cmdSend_Click()

                SMS.DeliveryReportOn()

                If SMS.SendSMS(Me.txtTo.Text, Me.txtMsg.Text) = True Then
                    '[VB - frmSMS] If SMS.SendSMS(Me.txtTo, Me.txtMsg) =
True Then

                        MsgBox("Message sent!", vbInformation)
                    Else
                        MsgBox("Message NOT sent!", vbInformation)
                        grpRegainPWD.Show()
                    End If
                End Sub

```



```

'-----
End If
Next

If (Not NameFound And Not GainPwdFound) Then
    MsgBox("Cannot find the information that you request", 0, "Not
in the table")
End If

End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    Me.Hide()
    frmLogin.Show()

    frmLogin.grpLoginChoose.Show()
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Me.Hide()
    frmLogin.Show()
End Sub
End Class

```

FORM MENU:

```

Public Class frmMenu
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim dsMemberPro, dsStaffPro As New DataSet
    Dim daMemberPro, daStaffPro As OleDb.OleDbDataAdapter
    Dim sqlMemberPro, sqlStaffPro As String
    Dim OleDbCon1 As System.Data.OleDb.OleDbConnection
    Private Sub PictureBox4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox4.Click

    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

```

```

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub frmMenu_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblMember' table. You can move, or remove it, as
needed.
        Me.TblMemberTableAdapter.Fill(Me.CarRentalMan1DataSet.tblMember)
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblUserName' table. You can move, or remove it, as
needed.

Me.TblUserNameTableAdapter.Fill(Me.CarRentalMan1DataSet.tblUserName)
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    End Sub

    Private Sub TblUserNameBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
        Me.Validate()
        Me.TblUserNameBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

    End Sub

    Private Sub btnStaffProfile_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles btnStaffProfile.Click

        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

        con.ConnectionString = dbProvider & dbSource
        con.Open()

```

```

sqlStaffPro = "SELECT * FROM tblUserName"
daStaffPro = New OleDb.OleDbDataAdapter(sqlStaffPro, con)
daStaffPro.Fill(dsStaffPro, "UserName")

con.Close()
MaxRows = dsStaffPro.Tables("UserName").Rows.Count
daStaffPro.Dispose()

Dim StaffName As String
Dim StaffNameFound As Boolean = False

StaffName = frmLogin.txtUserName.Text

For i As Integer = 0 To MaxRows - 1
    If
CStr(dsStaffPro.Tables("UserName").Rows(i)("StaffName").ToString) =
StaffName Then

        StaffNameFound = True
        inc = i

        frmProfile.txtStaffID.Text =
CStr(dsStaffPro.Tables("UserName").Rows(inc).Item(6))
        frmProfile.TxtStaffName.Text =
CStr(dsStaffPro.Tables("UserName").Rows(inc).Item(0))
        frmProfile.mtxtStaffPwd.Text =
CStr(dsStaffPro.Tables("UserName").Rows(inc).Item(7))
        frmProfile.mtxtStaffIC.Text =
(dsStaffPro.Tables("UserName").Rows(inc).Item(2))
        frmProfile.mtxtStaffPhone.Text =
CStr(dsStaffPro.Tables("UserName").Rows(inc).Item(3))
        frmProfile.TxtStaffEmail.Text =
CStr(dsStaffPro.Tables("UserName").Rows(inc).Item(4))
        frmProfile.txtStaffGender.Text =
CStr(dsStaffPro.Tables("UserName").Rows(inc).Item(1))
        frmProfile.rtxtStaffAdd.Text =
CStr(dsStaffPro.Tables("UserName").Rows(inc).Item(5))

        Me.Hide()
        frmProfile.Show()
        frmProfile.grpStaffProfile.Show()
        frmProfile.grpMemberProfile.Hide()
    End If
Next
If (Not StaffNameFound) Then
    MsgBox("You are NOT a staff of CarRental2U Company ", 0, "Not
in the table")
End If

End Sub

Private Sub btnMemberProfile_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles btnMemberProfile.Click

```

```

dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

con.ConnectionString = dbProvider & dbSource
con.Open()
sqlMemberPro = "SELECT * FROM tblMember"
daMemberPro = New OleDb.OleDbDataAdapter(sqlMemberPro, con)
daMemberPro.Fill(dsMemberPro, "Member")

con.Close()
MaxRows = dsMemberPro.Tables("Member").Rows.Count
daMemberPro.Dispose()

Dim MemberName As String
Dim MemberNameFound As Boolean = False

MemberName = frmLogin.txtMemName.Text

For i As Integer = 0 To MaxRows - 1
    If
CStr(dsMemberPro.Tables("Member").Rows(i)("MemberName").ToString) =
MemberName Then

        MemberNameFound = True
        inc = i

        frmProfile.txtMemID.Text =
CStr(dsMemberPro.Tables("Member").Rows(inc).Item(0))
        frmProfile.txtMemName.Text =
CStr(dsMemberPro.Tables("Member").Rows(inc).Item(1))
        frmProfile.mtxtMemIC.Text =
CStr(dsMemberPro.Tables("Member").Rows(inc).Item(3))
        frmProfile.mtxtMemContact.Text =
CStr(dsMemberPro.Tables("Member").Rows(inc).Item(4))
        frmProfile.txtMemEmail.Text =
CStr(dsMemberPro.Tables("Member").Rows(inc).Item(5))
        frmProfile.mtxtMemPwd.Text =
CStr(dsMemberPro.Tables("Member").Rows(inc).Item(7))
        frmProfile.txtMemGender.Text =
CStr(dsMemberPro.Tables("Member").Rows(inc).Item(2))
        frmProfile.rtxtMemAdd.Text =
CStr(dsMemberPro.Tables("Member").Rows(inc).Item(6))

        Me.Hide()
        frmProfile.Show()

        frmProfile.grpMemberProfile.Show()
        frmProfile.grpStaffProfile.Hide()
    End If
Next
If (Not MemberNameFound) Then
    MsgBox("Only register member will able to log in", 0, "Not in
the table")
End If

```

```

End Sub

Private Sub btnManageCar_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)
    Me.Hide()
    frmCarManage.Show()

End Sub

Private Sub btnReturnCar_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnReturnCar.Click
    Me.Hide()
    frmReturnCar.ShowDialog()

End Sub

Private Sub btnList_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnList.Click
    Me.Hide()
    frmCarList.ShowDialog()

End Sub

Private Sub btnBorrowCar_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnBorrowCar.Click
    Me.Hide()
    frmBorrowCar.ShowDialog()

End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
    Me.Hide()
    frmLogin.Show()

    frmLogin.grpLoginChoose.Show()

End Sub

Private Sub btnCarInfo_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)

End Sub

Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnManageCar.Click
    Me.Hide()
    frmCarManage.ShowDialog()

End Sub
End Class

```

FORM POLICY:

```
Public Class frmPolicy

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub PictureBox1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox1.Click

    End Sub
```

```

        Private Sub frmPolicy_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

        End Sub
End Class

```

FORM PROFILE:

```

Public Class frmProfile
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim dsMemberProUp, dsStaffProUp As New DataSet
    Dim daMemberProUp, daStaffProUp As OleDb.OleDbDataAdapter
    Dim sqlMemberProUp, sqlStaffProUp As String
    Dim OleDbCon1 As System.Data.OleDb.OleDbConnection
    Private Sub grpStaffProfile_Enter(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles grpStaffProfile.Enter

        End Sub

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

```

```

End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Me.Hide()
    frmPolicy.Show()
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    Me.Hide()
    frmTerm.Show()
End Sub

Private Sub TblMemberBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
    Me.Validate()
    Me.TblMemberBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

End Sub

Private Sub frmProfile_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    'TODO: This line of code loads data into the
    'CarRentalMan1DataSet.tblUserName' table. You can move, or remove it, as
    needed.

Me.TblUserNameTableAdapter.Fill(Me.CarRentalMan1DataSet.tblUserName)
    'TODO: This line of code loads data into the
    'CarRentalMan1DataSet.tblMember' table. You can move, or remove it, as
    needed.
    Me.TblMemberTableAdapter.Fill(Me.CarRentalMan1DataSet.tblMember)
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"
    con.ConnectionString = dbProvider & dbSource
    con.Open()

    txtMemName.ReadOnly = True
    txtMemID.ReadOnly = True
    mtxtMemPwd.ReadOnly = True
    mtxtMemIC.ReadOnly = True
    mtxtMemContact.ReadOnly = True
    txtMemEmail.ReadOnly = True
    txtMemGender.ReadOnly = True

    TxtStaffName.ReadOnly = True
    txtStaffID.ReadOnly = True
    mtxtStaffPwd.ReadOnly = True
    mtxtStaffIC.ReadOnly = True
    mtxtStaffPhone.ReadOnly = True
    TxtStaffEmail.ReadOnly = True
    txtStaffGender.ReadOnly = True

End Sub

Private Sub btnUpMem_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnUpMem.Click

```



```

txtMemName.ReadOnly = False
txtMemID.ReadOnly = False
mtxtMemPwd.ReadOnly = False
mtxtMemIC.ReadOnly = False
mtxtMemContact.ReadOnly = False
txtMemEmail.ReadOnly = False
txtMemGender.ReadOnly = False
rtxtMemAdd.ReadOnly = False
End Sub

Private Sub btnSaveMem_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSaveMem.Click
    sqlMemberProUp = "SELECT * FROM tblMember"
    daMemberProUp = New OleDb.OleDbDataAdapter(sqlMemberProUp, con)
    daMemberProUp.Fill(dsMemberProUp, "Member")

    con.Close()
    MaxRows = dsMemberProUp.Tables("Member").Rows.Count
    inc = -1

    Dim UpdateMemFound As Boolean = False
    For i As Integer = 0 To MaxRows - 1
        If
CStr(dsMemberProUp.Tables("Member").Rows(i)("MemberName").ToString) =
txtMemName.Text Then
            UpdateMemFound = True
            inc = i

            Dim cb As New OleDb.OleDbCommandBuilder(daMemberProUp)

            If inc <> -1 Then

                dsMemberProUp.Tables("Member").Rows(inc).Item(0) =
txtMemID.Text
                dsMemberProUp.Tables("Member").Rows(inc).Item(1) =
txtMemName.Text
                dsMemberProUp.Tables("Member").Rows(inc).Item(3) =
mtxtMemIC.Text
                dsMemberProUp.Tables("Member").Rows(inc).Item(4) =
mtxtMemContact.Text
                dsMemberProUp.Tables("Member").Rows(inc).Item(5) =
txtMemEmail.Text
                dsMemberProUp.Tables("Member").Rows(inc).Item(7) =
mtxtMemPwd.Text
                dsMemberProUp.Tables("Member").Rows(inc).Item(2) =
txtMemGender.Text
                dsMemberProUp.Tables("Member").Rows(inc).Item(6) =
rtxtMemAdd.Text

                daMemberProUp.Update(dsMemberProUp, "Member")

                MsgBox("Data updated")
            End If
        End If
    Next
    If (Not UpdateMemFound) Then
        MsgBox("Update is not available")
    End If
End Sub

```

```

        txtMemID.Text = dsMemberProUp.Tables("Member").Rows(inc).Item(0)
        txtMemName.Text = dsMemberProUp.Tables("Member").Rows(inc).Item(1)
        mtxtMemIC.Text = dsMemberProUp.Tables("Member").Rows(inc).Item(3)
        mtxtMemContact.Text =
dsMemberProUp.Tables("Member").Rows(inc).Item(4)
        txtMemEmail.Text = dsMemberProUp.Tables("Member").Rows(inc).Item(5)
        mtxtMemPwd.Text = dsMemberProUp.Tables("Member").Rows(inc).Item(7)
        txtMemGender.Text =
dsMemberProUp.Tables("Member").Rows(inc).Item(2)
        rtxtMemAdd.Text = dsMemberProUp.Tables("Member").Rows(inc).Item(6)

```

End Sub

```

Private Sub btnUpStaff_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnUpStaff.Click
    TxtStaffName.ReadOnly = False
    txtStaffID.ReadOnly = False
    mtxtStaffPwd.ReadOnly = False
    mtxtStaffIC.ReadOnly = False
    mtxtStaffPhone.ReadOnly = False
    TxtStaffEmail.ReadOnly = False
    txtStaffGender.ReadOnly = False
    rtxtStaffAdd.ReadOnly = False

```

End Sub

```

Private Sub btnSaveStaff_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnSaveStaff.Click
    sqlStaffProUp = "SELECT * FROM tblUserName"
    daStaffProUp = New OleDb.OleDbDataAdapter(sqlStaffProUp, con)
    daStaffProUp.Fill(dsStaffProUp, "UserName")

```

```

    con.Close()
    MaxRows = dsStaffProUp.Tables("UserName").Rows.Count

```

```

    Dim UpdateStaffFound As Boolean = False
    For i As Integer = 0 To MaxRows - 1
        If
CStr(dsStaffProUp.Tables("UserName").Rows(i)("StaffName").ToString) =
TxtStaffName.Text Then
            UpdateStaffFound = True
            inc = i

```

```

        Dim cb As New OleDb.OleDbCommandBuilder(daStaffProUp)
        If inc <> -1 Then

            dsStaffProUp.Tables("UserName").Rows(inc).Item(6) =
txtStaffID.Text
            dsStaffProUp.Tables("UserName").Rows(inc).Item(0) =
TxtStaffName.Text
            dsStaffProUp.Tables("UserName").Rows(inc).Item(7) =
mtxtStaffPwd.Text
            dsStaffProUp.Tables("UserName").Rows(inc).Item(2) =
mtxtStaffIC.Text
            dsStaffProUp.Tables("UserName").Rows(inc).Item(3) =
mtxtStaffPhone.Text
            dsStaffProUp.Tables("UserName").Rows(inc).Item(4) =
TxtStaffEmail.Text

```

```

        dsStaffProUp.Tables("UserName").Rows(inc).Item(1) =
txtStaffGender.Text
        dsStaffProUp.Tables("UserName").Rows(inc).Item(5) =
rtxtStaffAdd.Text

        daStaffProUp.Update(dsStaffProUp, "UserName")

        MsgBox("Data updated")

    End If
End If
Next
If (Not UpdateStaffFound) Then
    MsgBox("Update is not available")
End If

txtStaffID.Text = dsStaffProUp.Tables("UserName").Rows(inc).Item(6)
TxtStaffName.Text =
dsStaffProUp.Tables("UserName").Rows(inc).Item(0)
mtxtStaffPwd.Text =
dsStaffProUp.Tables("UserName").Rows(inc).Item(7)
mtxtStaffIC.Text =
dsStaffProUp.Tables("UserName").Rows(inc).Item(2)
mtxtStaffPhone.Text =
dsStaffProUp.Tables("UserName").Rows(inc).Item(3)
TxtStaffEmail.Text =
dsStaffProUp.Tables("UserName").Rows(inc).Item(4)
txtStaffGender.Text =
dsStaffProUp.Tables("UserName").Rows(inc).Item(1)
rtxtStaffAdd.Text =
dsStaffProUp.Tables("UserName").Rows(inc).Item(5)

End Sub

Private Sub grpMemberProfile_Enter(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles grpMemberProfile.Enter

End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
    Me.Hide()
    frmLogin.Show()

    frmLogin.grpLoginChoose.Show()
End Sub

Private Sub txtMemGender_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtMemGender.TextChanged

End Sub
End Class

```

FORM RECEIPT:

```
Public Class frmReceipt

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub frmReceipt_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    End Sub

    Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExit.Click
```

```

        Me.Close()
        frmLogin.Show()
    End Sub

    Private Sub btnBack_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBack.Click
        Me.Hide()
        frmBorrowCar.Show()
        ListBorrow.ClearSelected()
    End Sub

    Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()
    End Sub

    Private Sub btnprint_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnprint.Click
        PrintForm1.Print()
    End Sub
End Class

```

FORM REGISTER:

```

Public Class frmREGISTER_USER
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim dsMember, dsStaff As New DataSet
    Dim daMember, daStaff As OleDb.OleDbDataAdapter
    Dim sqlMember, sqlStaff As String
    Dim OleDbCon1 As System.Data.OleDb.OleDbConnection
    Private Sub grpRegister_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles grpRegister.Enter

    End Sub

    Private Sub frmREGISTER_USER_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the
        'CarRentalMan1DataSet.tblMember' table. You can move, or remove it, as
        needed.
        Me.TblMemberTableAdapter.Fill(Me.CarRentalMan1DataSet.tblMember)
        'TODO: This line of code loads data into the
        'CarRentalMan1DataSet.tblUserName' table. You can move, or remove it, as
        needed.
        Me.TblUserNameTableAdapter.Fill(Me.CarRentalMan1DataSet.tblUserName)

        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

        txtUserID.ReadOnly = True
    End Sub
End Class

```

```

End Sub

Private Sub TblUserNameBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
    Me.Validate()
    Me.TblUserNameBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClear.Click
    TxtName.Clear()
    MaskedIC.Clear()
    MaskedPhone.Clear()
    TxtEmail.Clear()
    TxtGender.Clear()
    txtUserID.Clear()
    TxtAddress.Clear()
End Sub

Private Sub btnRegisterMember_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnRegisterMember.Click
    Dim MemberPwd As String
    MemberPwd = InputBox("Create your own password:")

    con.ConnectionString = dbProvider & dbSource
    con.Open()

    sqlMember = "SELECT * FROM tblMember"
    daMember = New OleDb.OleDbDataAdapter(sqlMember, con)
    daMember.Fill(dsMember, "Member")

    con.Close()
    MaxRows = dsMember.Tables("Member").Rows.Count

    If inc <> -1 Then

        Dim cb As New OleDb.OleDbCommandBuilder(daMember)
        Dim dsNewRowMember As DataRow

        dsNewRowMember = dsMember.Tables("Member").NewRow()

        If inc <> -1 Then
            Dim memberTitle As String
            Dim memberFound As Boolean = False
            memberTitle = TxtName.Text

            For i As Integer = 0 To MaxRows - 1
                If
CStr(dsMember.Tables("Member").Rows(i)("MemberName").ToString) =
memberTitle Then
                    memberFound = False
                    inc = i
                    MsgBox("Please Change you name because the name had
been used")

                    memberTitle = InputBox("Insert the new name : ")
                    dsNewRowMember.Item(0) = txtUserID.Text

```

```

        dsNewRowMember.Item(1) = TxtName.Text
        dsNewRowMember.Item(2) = TxtGender.Text
        dsNewRowMember.Item(3) = MaskedIC.Text
        dsNewRowMember.Item(4) = MaskedPhone.Text
        dsNewRowMember.Item(5) = TxtEmail.Text
        dsNewRowMember.Item(6) = TxtAddress.Text
        dsNewRowMember.Item(7) = MemberPwd

    End If
Next

If (Not memberFound) Then
    memberFound = True
    dsNewRowMember.Item(0) = txtUserID.Text
    dsNewRowMember.Item(1) = memberTitle
    dsNewRowMember.Item(2) = TxtGender.Text
    dsNewRowMember.Item(3) = MaskedIC.Text
    dsNewRowMember.Item(4) = MaskedPhone.Text
    dsNewRowMember.Item(5) = TxtEmail.Text
    dsNewRowMember.Item(6) = TxtAddress.Text
    dsNewRowMember.Item(7) = MemberPwd
End If

dsMember.Tables("Member").Rows.Add(dsNewRowMember)

daMember.Update(dsMember, "Member")
inc = 0

MsgBox("Your Registration Process is succeed !, You may
LOGIN NOW")

Else : MsgBox("No Data Have Been Added !!!")
End If
End If
End Sub

Private Sub BtnRegisterStaff_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles BtnRegisterStaff.Click
    Dim AdminPwdTitle As String
    Dim YourPwd As String
    Dim AdminPwdFound As Boolean = False

    AdminPwdTitle = InputBox("Enter the Admin Password :")
    YourPwd = InputBox("Create your own Password :")

    If AdminPwdTitle = "123456" Then

        AdminPwdFound = True
        con.ConnectionString = dbProvider & dbSource
        con.Open()

        sqlStaff = "SELECT * FROM tblUserName"
        daStaff = New OleDb.OleDbDataAdapter(sqlStaff, con)
        daStaff.Fill(dsStaff, "UserName")

        con.Close()
        MaxRows = dsStaff.Tables("UserName").Rows.Count

        If inc <> -1 Then

            Dim cb As New OleDb.OleDbCommandBuilder(daStaff)

```

```

Dim dsNewRowStaff As DataRow

dsNewRowStaff = dsStaff.Tables("UserName").NewRow()

If inc <> -1 Then

    Dim BookTitle As String
    Dim bookFound As Boolean = False

    BookTitle = TxtName.Text

    For i As Integer = 0 To MaxRows - 1
        If
CStr(dsStaff.Tables("Username").Rows(i)("StaffName").ToString) = BookTitle
Then

            bookFound = False
            inc = i
            MsgBox("Please Change your name because the
name had been used")

            BookTitle = InputBox("Insert the new name :")

            dsNewRowStaff.Item(0) = BookTitle
            dsNewRowStaff.Item(1) = TxtGender.Text
            dsNewRowStaff.Item(2) = MaskedIC.Text
            dsNewRowStaff.Item(3) = MaskedPhone.Text
            dsNewRowStaff.Item(4) = TxtEmail.Text
            dsNewRowStaff.Item(5) = TxtAddress.Text
            dsNewRowStaff.Item(6) = txtUserID.Text
            dsNewRowStaff.Item(7) = YourPwd

        End If
    Next
    If (Not bookFound) Then
        bookFound = True
        dsNewRowStaff.Item(6) = txtUserID.Text
        dsNewRowStaff.Item(0) = BookTitle
        dsNewRowStaff.Item(7) = YourPwd
        dsNewRowStaff.Item(2) = MaskedIC.Text
        dsNewRowStaff.Item(3) = MaskedPhone.Text
        dsNewRowStaff.Item(4) = TxtEmail.Text
        dsNewRowStaff.Item(1) = TxtGender.Text
        dsNewRowStaff.Item(5) = TxtAddress.Text
    End If

    dsStaff.Tables("UserName").Rows.Add(dsNewRowStaff)

    daStaff.Update(dsStaff, "UserName")

    inc = 0
    MsgBox("System administrator will assign a USER ID for
you later. Please wait")
    MsgBox("Your Registration Process is succeed , You may
LOGIN NOW")

Else : MsgBox("No Data Have Been Added !!!")

```



```

        End If
    End If
Else
    MsgBox("Your admin password is incorrect")

    End If
End Sub

Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClose.Click
    Me.Hide()
    frmLogin.Show()

    frmLogin.grpLoginChoose.Show()
End Sub
End Class

```

FORM RETURN CAR:

```

Public Class frmReturnCar
    Dim inc As Integer
    Dim MaxRows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim dbProvider As String
    Dim dbSource As String
    Dim dsReturn, dsReminder, dsBorrowUpdate As New DataSet
    Dim daReturn, daReminder, daBorrowUpdate As OleDb.OleDbDataAdapter
    Dim sqlReturn, sqlReminder, Sql As String
    Private Sub NavigateRecords()

        txtBorrowerName.Text =
dsReturn.Tables("BorrowCar").Rows(inc).Item(0)
        txtCarId.Text = dsReturn.Tables("BorrowCar").Rows(inc).Item(1)
        txtBorrow.Text = dsReturn.Tables("BorrowCar").Rows(inc).Item(2)
        txtReturn.Text = dsReturn.Tables("BorrowCar").Rows(inc).Item(3)

    End Sub

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

```

```

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub frmReturnCar_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblCarDatabase' table. You can move, or remove it, as
needed.

Me.TblCarDatabaseTableAdapter.Fill(Me.CarRentalMan1DataSet.tblCarDatabase)
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblBorrowCar' table. You can move, or remove it, as
needed.

Me.TblBorrowCarTableAdapter.Fill(Me.CarRentalMan1DataSet.tblBorrowCar)
        'TODO: This line of code loads data into the
'CarRentalMan1DataSet.tblMember' table. You can move, or remove it, as
needed.
        Me.TblMemberTableAdapter.Fill(Me.CarRentalMan1DataSet.tblMember)

        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
        dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

        con.ConnectionString = dbProvider & dbSource
        con.Open()

        sqlReturn = "SELECT * FROM tblBorrowCar"
        daReturn = New OleDb.OleDbDataAdapter(sqlReturn, con)
        daReturn.Fill(dsReturn, "BorrowCar")

        con.Close()

```

```

MaxRows = dsReturn.Tables("BorrowCar").Rows.Count
inc = -1
grpSendReminder.Hide()
btnCarStatus.Visible = False

End Sub

Private Sub TblMemberBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
    Me.Validate()
    Me.TblMemberBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.CarRentalMan1DataSet)

End Sub

Private Sub grpReturn_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles grpReturn.Enter

End Sub

Private Sub btnRetrunBack_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnRetrunBack.Click
    sqlReturn = "SELECT * FROM tblBorrowCar"
    daReturn = New OleDb.OleDbDataAdapter(sqlReturn, con)
    daReturn.Fill(dsReturn, "BorrowCar")

    con.Close()
    MaxRows = dsReturn.Tables("BorrowCar").Rows.Count

    Dim cb As New OleDb.OleDbCommandBuilder(daReturn)
    Dim testmsg As Integer

    Dim UpdateMemFound As Boolean = False
    For i As Integer = 0 To MaxRows - 1
        If
CStr(dsReturn.Tables("BorrowCar").Rows(i)("BorrowerName").ToString) =
txtBorrowerName.Text Then
            UpdateMemFound = True
            inc = i

            testmsg = MsgBox("RETURN BACK A CAR ?", 1, "Delete Option")
            If testmsg = 1 Then
                btnCarStatus.Visible = True
                btnCarStatus.PerformClick()

                MsgBox("Please Use our service again. Thank you")
                inc = (i)
                dsReturn.Tables("BorrowCar").Rows(inc).Delete()

                MaxRows = MaxRows - 1
                inc = 0

                daReturn.Update(dsReturn, "BorrowCar")

            Else
                MessageBox.Show("You might check your data again")
            End If
        End If
    Next

```

```

        End If
    ElseIf
CStr(dsReturn.Tables("BorrowCar").Rows(i)("BorrowerName").ToString) =
txtBorrowerName.Text Then
        UpdateMemFound = True
        MsgBox("There is no more data or records")
    End If

Next
If (Not UpdateMemFound) Then
    MsgBox("Update is not available")
End If
End Sub

Private Sub btnGenerate_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnGenerate.Click
    If dtpReturn.Value.Date = txtReturn.Text Then
        txtReminder.Text = "You return a car on time. Thank you for
your cooperation."
    ElseIf dtpReturn.Value.Date > txtReturn.Text Then
        txtReminder.Text = "You are late. Please return a car as soon
as possible. Thank you."
    ElseIf dtpReturn.Value.Date < txtReturn.Text Then
        txtReminder.Text = "Reminder: Please return a car on time.Have
a nice day."
    End If
End Sub

Private Sub btnRemind_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnRemind.Click
    If SMS.ConnectToGSM = True Then
        grpSendReminder.Show()
        '[VB - frmModem] frmSMS.Show(vbModal)
    Else
        MsgBox("GSM Modem is NOT connected with network!", vbCritical)
    End If
End Sub

Private Sub btnCarStatus_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnCarStatus.Click
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

    con.ConnectionString = dbProvider & dbSource
    con.Open()
    Sql = "SELECT * FROM tblCarDatabase"
    daBorrowUpdate = New OleDb.OleDbDataAdapter(Sql, con)
    daBorrowUpdate.Fill(dsBorrowUpdate, "CarDatabase")

    con.Close()
    MaxRows = dsBorrowUpdate.Tables("CarDatabase").Rows.Count

    Dim book1Found As Boolean = False

    For i As Integer = 0 To MaxRows - 1

```

```

        If
CStr(dsBorrowUpdate.Tables("CarDatabase").Rows(i)("CarID").ToString) =
txtCarId.Text Then
            book1Found = True
            inc = i

            Dim cb As New OleDb.OleDbCommandBuilder(daBorrowUpdate)
            If inc <> -1 Then

                dsBorrowUpdate.Tables("CarDatabase").Rows(inc).Item(5)
= "AVAILABLE"
                daBorrowUpdate.Update(dsBorrowUpdate, "CarDatabase")
                MsgBox("Car being returned")
            End If
        End If

    Next
    If (Not book1Found) Then
        MsgBox("No Record for a car is rent ", 0, "Not in the table")
    End If

```

```
End Sub
```

```

Private Sub btnSend_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSend.Click
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;"
    dbSource = "Data Source=C:\Users\fazreeny\Documents\Visual Studio
2008\Backup Files\crms\carRentalMan1.accdb"

```

```

    con.ConnectionString = dbProvider & dbSource
    con.Open()

```

```

    sqlReminder = "SELECT * FROM tblMember"
    daReminder = New OleDb.OleDbDataAdapter(sqlReminder, con)
    daReminder.Fill(dsReminder, "Member")

```

```

    con.Close()
    MaxRows = dsReminder.Tables("Member").Rows.Count
    inc = -1

```

```

    Dim BorrowerName As String
    Dim NameFound As Boolean = False

```

```
BorrowerName = txtBorrowerName.Text
```

```

    For i As Integer = 0 To MaxRows - 1
        If
CStr(dsReminder.Tables("Member").Rows(i)("MemberName").ToString) =
BorrowerName Then

            NameFound = True
            inc = i

```

```

daReminder.Dispose()

txtBorrowerNum.Text =
CStr(dsReminder.Tables("Member").Rows(inc).Item(4))
txtAdminNum.Text = "0193930372"

'-----

''[VB - frmSMS] Private Sub cmdSend_Click()

SMS.DeliveryReportOn()

If SMS.SendSMS(Me.txtBorrowerNum.Text, Me.txtReminder.Text)
= True Then
    '[VB - frmSMS] If SMS.SendSMS(Me.txtTo, Me.txtMsg) =
True Then

        MsgBox("Message sent!", vbInformation)
    Else
        MsgBox("Message NOT sent!", vbInformation)

    End If

'-----
End If
Next

If (Not NameFound) Then
    MsgBox("Cannot find the information that you request", 0, "Not
in the table")
End If
End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
    Me.Hide()
    frmLogin.Show()

    frmLogin.grpLoginChoose.Show()
End Sub
End Class

```

FORM TERM:

```
Public Class frmTerm

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmMenu.Show()

    End Sub

    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmTestimonial.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()
    End Sub

    Private Sub frmTerm_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    End Sub
End Class
```

FORM TESTIMONIALS:

```
Public Class frmTestimonial

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide()
        frmCarTypes.Show()
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.Hide()
        frmMenu.Show()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        frmCustPromo.Show()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Me.Hide()
        frmLogin.Show()

        frmLogin.grpLoginChoose.Show()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Me.Hide()
        frmContact.Show()
    End Sub

    Private Sub Button9_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button9.Click
        Me.Hide()
        frmPolicy.Show()
    End Sub

    Private Sub Button8_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button8.Click
        Me.Hide()
        frmTerm.Show()
    End Sub

    Private Sub TblMemberBindingNavigatorSaveItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
```



```

End Sub

Private Sub frmTestimonial_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load

End Sub

Private Sub RichTextBox5_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles RichTextBox5.TextChanged

End Sub
End Class

```

MODULE SMS:

```

Module ModuleSMS_Setting
    'Option Explicit On
    'Option Explicit

    Public SMS As New MobitekSMSAPI5.SMS
    'Public USSD As New MobitekSMSAPI5.USSD
    'Public PB As New MobitekSMSAPI5.Phonebook

    Public Sub Initmodem()

        Dim intModemStatus As Integer
        SMS.ConnectToGSM()

        'Initialised the modem at COM port 7 ie. to establish connection
        btw. PC and GSM modem
        'Please change the COM port number accordingly, please refer to the
        Installation Guide on how to find the COM Port number
        intModemStatus = SMS.ModemInit(frmLogin.txtPortNo.Text)

        Select Case intModemStatus

            Case 0
                MsgBox("GSM Modem Not Connected!")
                '[VB - Module1] frmModem.txtText = "GSM Modem Not
Connected!"
                frmLogin.grpStaffMenu.Show()
                frmLogin.txtPortNo.Clear()

                Exit Sub

            Case 1
                MsgBox("SUCCESSFULLY CONNECTED WITH MODEM!")
                '[VB - Module1] frmModem.txtText = "GSM Modem Connected!"
                frmLogin.grpStaffMenu.Show()
                frmLogin.grpModem.Hide()
                Exit Sub

            Case 2
                MsgBox("PIN Required!")

```

```

        '[VB - Module1] frmModem.txtText = "PIN Required!"
        frmLogin.Show()
    Exit Sub

    Case 3
        MsgBox("Incorrect PIN Entered! Warning after 3 tries of
incorrect PIN entered, your SIM card will be blocked by TELCO!")
        '[VB - Module1] frmModem.txtText = "Incorrect PIN entered!
Warning: after 3 tries of incorrect PIN entered, your SIM card will be
blocked by TELCO!"
    Exit Sub

    Case 4
        MsgBox("Your SIM card is blocked by TELCO!")
        '[VB - Module1] frmModem.txtText = "Your SIM card is
blocked by TELCO!"
    Exit Sub

    Case 5
        MsgBox("Your SIM card has problem!")
        '[VB - Module1] frmModem.txtText = "Your SIM card has
problem!"
    Exit Sub

    Case Else
        MsgBox("GSM Modem Not Connected!")
        '[VB - Module1] frmModem.txtText = "GSM Modem Not
Connected!"
    Exit Sub

End Select

'You can use this API function anywhere in your code to check the
modem's connection to GSM network
If SMS.ConnectToGSM = True Then

    MsgBox("GSM modem is connected to GSM Network.")
Else
    MsgBox("GSM modem is disconnected from the GSM Network. Please
re-initialised GSM modem.")

    'Close modem connection to PC

    SMS = Nothing

    Exit Sub
End If
End Sub
End Module

```