



NEAR EAST UNIVERSITY

FACULTY OF ECONOMICA AND
ADMINISTRATIVE SCIENCES

DEPARTMENT OF COMPUTER
INFORMATION SYSTEMS

2005/2006 SPRING TERM
CIS 400
(Graduation Project)

“RENT A CAR MANAGEMENT SYSTEM

Submitted to : Department of C.I.S.
Submitted by: Murat ÖZERIN
Student No. : 20001026

Lefkosa
2006



CONTENTS

I. ACKNOWLEDGEMENTS1
II. ABSTRACT2
III. INTRODUCTION4
IV. SYSTEM DESIGN	
• Data Flow Diagram10
• Database Design18
V. USER MANUAL	
• User Guidelines21
• User Flow Charts30
VI. REFERENCES46
APPENDIX I	
• Screen outputs47
APPENDIX II	
• Source Codes53

ACKNOWLEDGEMENTS

First of all I say thanks to my Almighty God who gave me abilities and helped me do anything that was possible by me.

I am then very much thankful to my advisor Dr. Mustafa Menekay who helped me from the core, and helped me when ever I was struck in programming problems.

I then say thanks to my parents who helped me and supported me till this very time, thank you very much mom and dad.

I will not forget my friends here who always motivated me when ever I got disappointed in any point while making this program

ABSTRACT

This software is a specific program only suitable for rent a car management system that makes business in keeping several cars and renting them to the public. The renting system is divided into 6 main processes in order for better user interface and friendly environment.

My solution to the problem I guess is quite understandable as I divided the system into main six processes, such as cars, members, reservations, rentals, reports and accounts. Each of the section I designed is easy to understand and learn fast. There are no menus used but command buttons were preferred instead so all is in front of the user for quick access.

I added user messages as well for the correspondence with the program so user may understand a bit the inside logic of the program how his process is handled.

I hope this software will help the rent a car management business manager keep close eye on his business accounts, his cars and performance, his customers activities using the IRS I designed for the purpose.

As there are hundreds of ways to solve any problem so if you find any suggestions about my program please do not hesitate telling me them.

INTRODUCTION

RENT A CAR MANAGEMENT SYSTEM

Brief explanation of the business:

“Rent a Car” is a kind of business in which a company holds several cars of different category and makes and give the cars to different people who are permitted by the law of the local government to drive a car for a specific period of time with a definite price called rent. Thus making profit through business.

All cars are insured with an insurance company so incase of any damage by a customer to the car or accident, the company (Rent A Car Company) does not pay the lost caused by that damage.

The company keeps records of all of its customers, cars, renting details, and its accounts. These records help them to maintain their performance and provide better service to their customers. It also helps the management in keeping the track of reservations, renting, or a customer’s profile may help in the reputation of that particular customer where to rent a car to him or not if he use to damage the cars.

Aim of the project:

My project is focused on converting a manual system of any “Rent a Car” company into a computer based system that shall keep records of the available cars, customers profiles, reservation details, renting details and other accounting information related to the business.

Like all other programs I shall try to focus on reducing the time consumed by the user in order to retrieve any kind of information out of the database than it would take doing it manually.

Besides this reservations and proper reservations play a key role in this kind of business. I have focused on this process as well in order to avoid multi reservations of a vehicle where the dates are conflicting.

Thus I can say that my this new database application program shall increase efficiency of the business providing accurate information based on the data entered by the user right in time, where for managers to choose the best decision before hand.

Existing System (Manual):

On the reception desk they have a big file that contains two basic parts.

1. Reservations

2. Renting

In reservation part they keep the information of the customer along with the car he is interested in, the dates and the duration of the vehicle that will stay with the customer.

In the other section receptionist writes down the name, surname, and the license number of the customer along with the car registration number, with the amount received.

Drawbacks of the Existing Manual System:

After my study I found following drawbacks in the existing manual system;

- There is no file of the vehicles that are available, but the receptionist look at the car in order to tell the customer its specifications or he/she remembers it by heart.
- It is very difficult for a new employee to tell the customer specification of a car as there is no detail data about the cars.
- Without such file or written information, it is very difficult rather impossible for new employee to convince which car is best suiting customer's need.
- There is no file about customer profile, their history of car hiring from the same company, but only in the reservation or renting book.
- Multiple data is entered in the registration book, meaning if the same customer comes to hire any car, all of his data is written again in the book, waste of time.
- Incase of lot of customers, two or more reservations dates may conflict each other thus causing customer dissatisfaction affecting badly the business in the local market.
- Same may happen for renting a car as well.
- Accounts are not kept accurately, who paid for which renting is not exactly known. In a file manually amount received is written by the receptionist.
- Employee takes lot of time in order to tell the business owner the detail activities of the day of the business and might not be correct.

New Computerized System:

As business grows, it becomes more and more difficult for the management to keep eye on all of the activities mentioned above manually. So the computer comes in the field to provide a better IRS having a bottle neck analysis, keep eye on all the records and maintaining them up to date, thus helping the management officers to take an appropriate action before hand.

After analyzing system of a local car renting company that was using a manual system of their record keeping of their customers, cars, renting activities as well as reservations in their only book, I came up with a solution described below.

Solution:

In car rent business to have a computer based system the application must posses the utilities for;

1. Date entry of all available cars
2. Edit/Update car specifications
3. Search a car and its availability
4. Customers and member registration with edit and update options
5. A proper reservation system that manages to schedule the dates so no conflict occurs
6. Hiring (renting) cars into the database and keeping track of all the cars rented when and to whom
7. Proper IRS (Information Reporting System) based on the data collected by the above mentioned processes
8. Computerized accounting system showing details of profit and loss as well as customer, members and business accounts

Keeping in mind the above eight (8) facilities I divided the whole system into six (6) main processes.

1. Car Registration (Registration of a car)
2. Members (Registration of a customer)
3. Reservation
4. Rentals
5. Accounts
6. Reports

Each of this process is explained as;

1. Car Registration:

This is our first main process in which we enter for the first time any car details to our system with all of its specifications and renting prices. In this process we have ability to enter a new car information, search for an existing car in the database, to modify or update the detail of a specific car in the database or may mark it up as a discontinue item incase it is to be no longer use in the business.

2. Members:

This process is used incase whenever a new customer arrives at the office for hiring a car, we call our customers as our members. Members of the business have some special benefits depending on how frequently does he/she hire car that a normal customer does not have i.e. a member may rent a car in credits, always availability of a car, etc. usually a member is not an individual person but is another company or any hotel business partner. This process is sub divided into three processes

- Add new member
- Edit/update member info
- Member Search

3. Reservations:

This process is very sensitive for the business. A proper reservation system makes the company popular among its customers otherwise conflicts in dates of reservation of the same car for two different customers may result into a very bad image in the market. I designed a computer program that will thoroughly check the reservation records in the database before a user is permitted to add another reservation for a specific period of time. In case of any conflict the user will be prompted and not permitted to make further process. There are two sub processes in Reservations

- New Reservation
- Edit Reservation

In new reservation we add a new record to the database confirming the reservation has been made by the system, while the other sub process allows us to edit the dates of reservation, update it to rent a car or simply cancel the reservation of the car. Here I would like to mention that we do not edit the dates of the reservation but cancel it and reserve it for new dates if required.

4. Rentals:

This process is hiring a car to a specific customer mentioned in the program who rents the car into the database. We add the information of the customer in here to our database that can be used in the future by the management officers. Accounts of business and its members are automatically updated through this process. It has also two sub parts in it, i.e.

- Rent a car to new customer
- Rent a car to member

5. Accounts:

Here we can check the accounts details of our business and our members. From here we can find out if we made profit or loss for a specific period of time.

6. Reports:

This section is dedicated to the managerial purpose only. It allows them to have a complete look on their business activities, reservations, rental details, cars and their

members as well as their accounts. Profit can be checked from here; some reports are mentioned as an example here below;

- All reservation in specific period of time
- All rentals in specific period of time
- Members' List
- Cars' List
- Available cars
- Profit reports etc.

6. Benefits of new system:

- It keeps all records of the available cars for hiring in the business
- It is very user friendly
- New employee can easily adopt this system thus being able to discuss any car with the customer easily
- Reservation dates confliction is completely avoided
- Accounts of the business are secured
- Status of each vehicle is recorded and can be found easily through the application where exactly and to whom exactly it is hired.
- History of the cars is available, enabling owner to decide for future what car is most liked by the customer
- History of the customer is available, thus enabling receptionist if to rent a car to him or not, as it may happen that same customer has a damage record more than once.
- History of the reservation is available
- History of hiring car is available
- Based on all the above reports it is very easy to determine any kind of mistake or problem in the business and take an appropriate action before time.

PROJECT BOUNDRIES

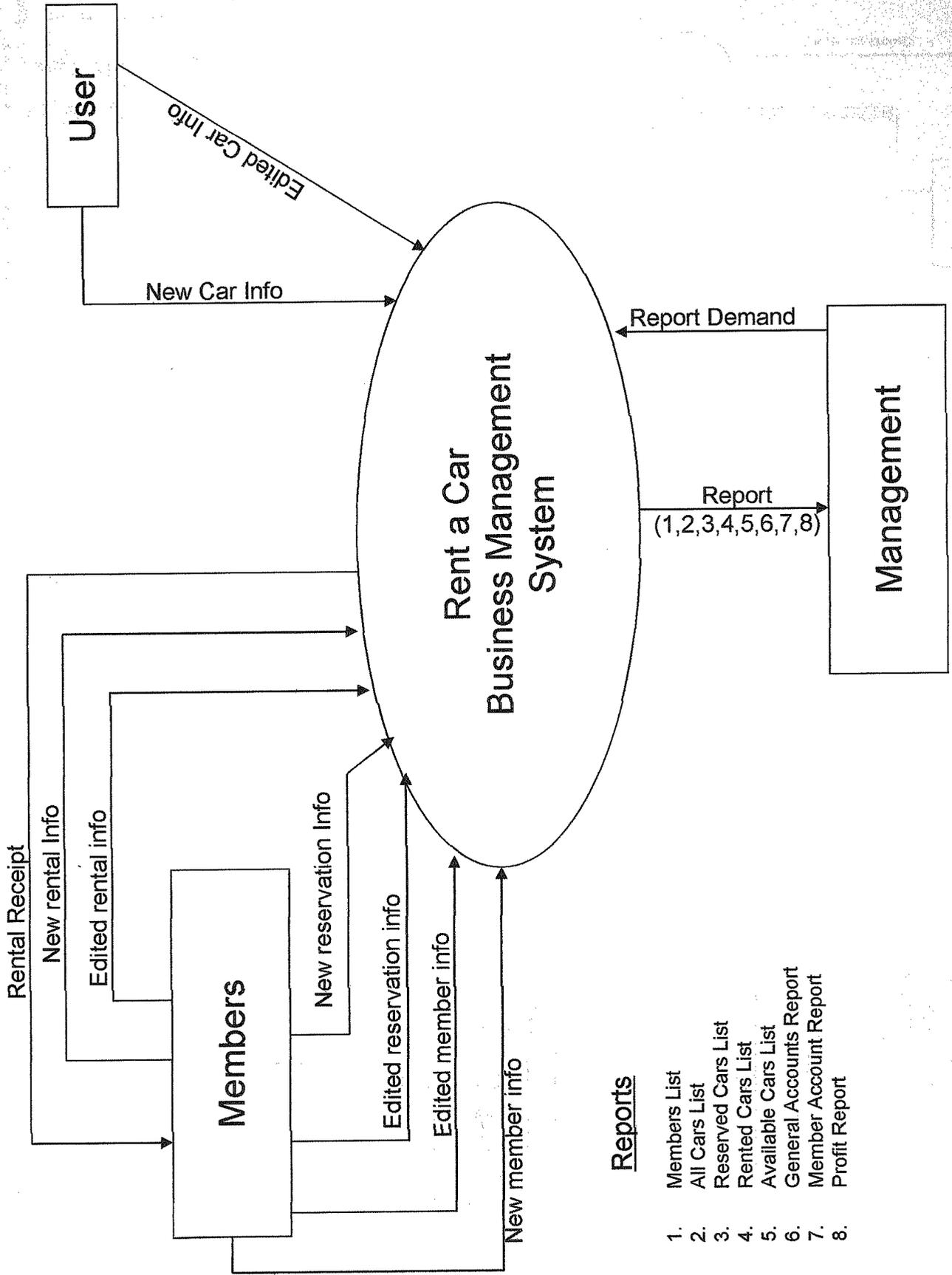
- 1) Most of the employees do not have computer knowledge.
- 2) Registering specifications of the cars might be a boring task for existing employees and can be time consuming.
- 3) Wrong entry to the system may cause in generating wrong reports.
- 4) Loss of computer, damage to hard disk etc, all data might be lost, so all reports are kept manually as well.
- 5) It is not easy to train employee and is time consuming as well.
- 6) Most of old fashion experienced employees do not want to accept computer based application as they have fear new young may replace them
- 7) Purchase of computers and making a network system in the company cost, and maintenance is expensive.

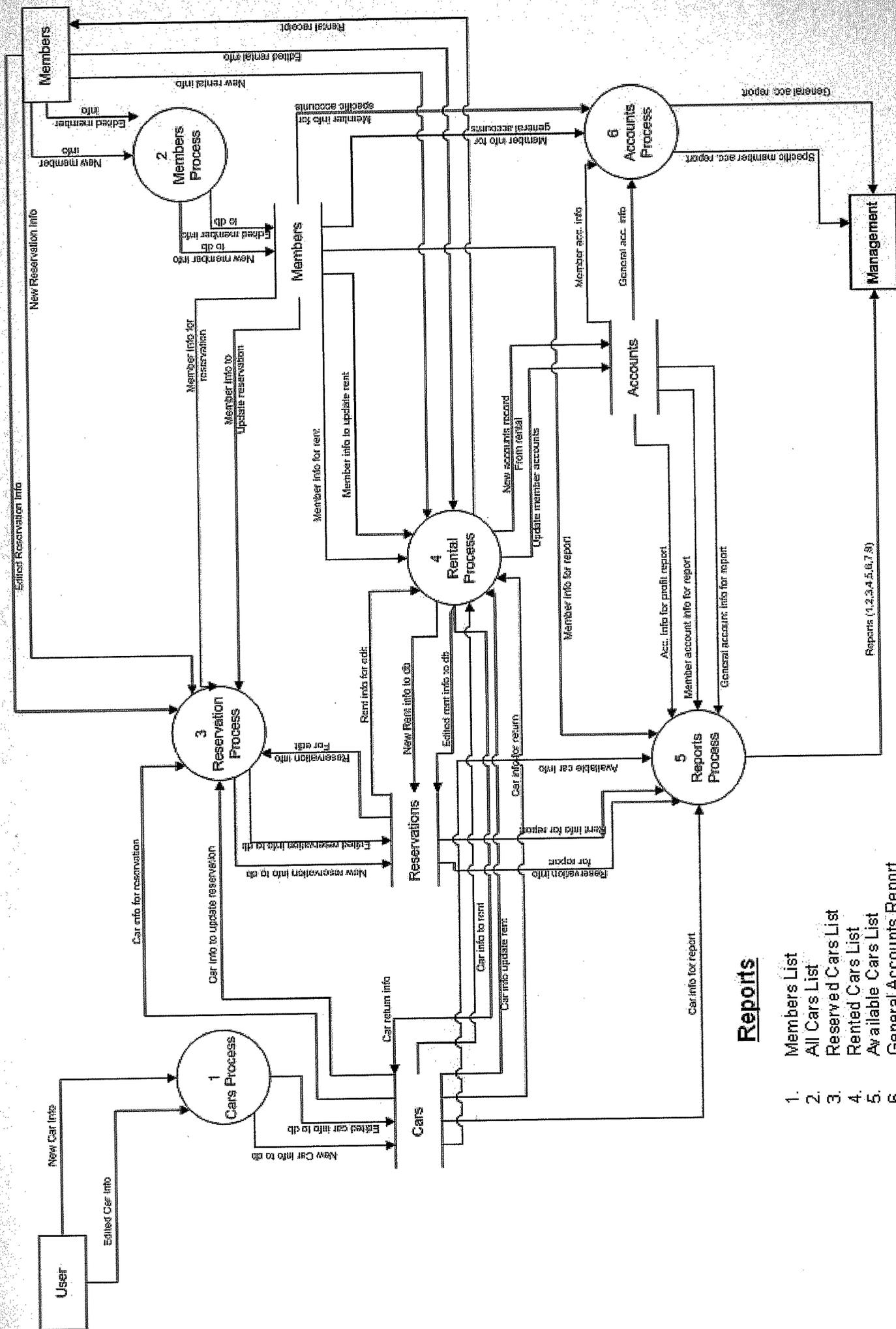
SYSTEM REQUIREMENTS

- Two computers for reception and one for the boss
- Computer minimum 1.5 GHz
- Hard Disk capacity 100 MB
- Operating system: Windows XP or Windows NT
- Installed MS Office 2002 on the system

I consider that my study shall bring positive solution to such any "Rent a Car" company. Besides that I shall be very glad for anyone's good suggestion to my solutions.

Context DFD



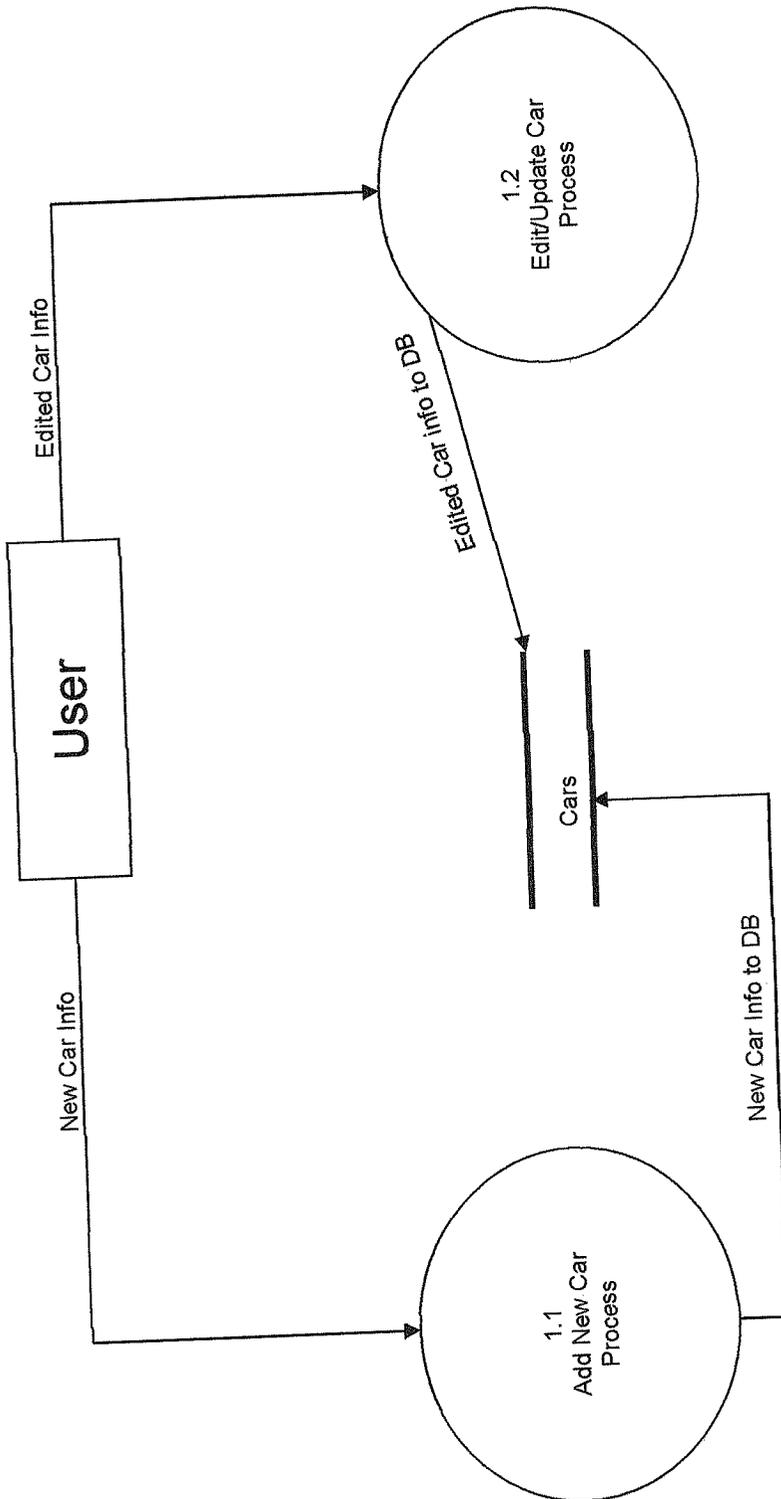


Reports

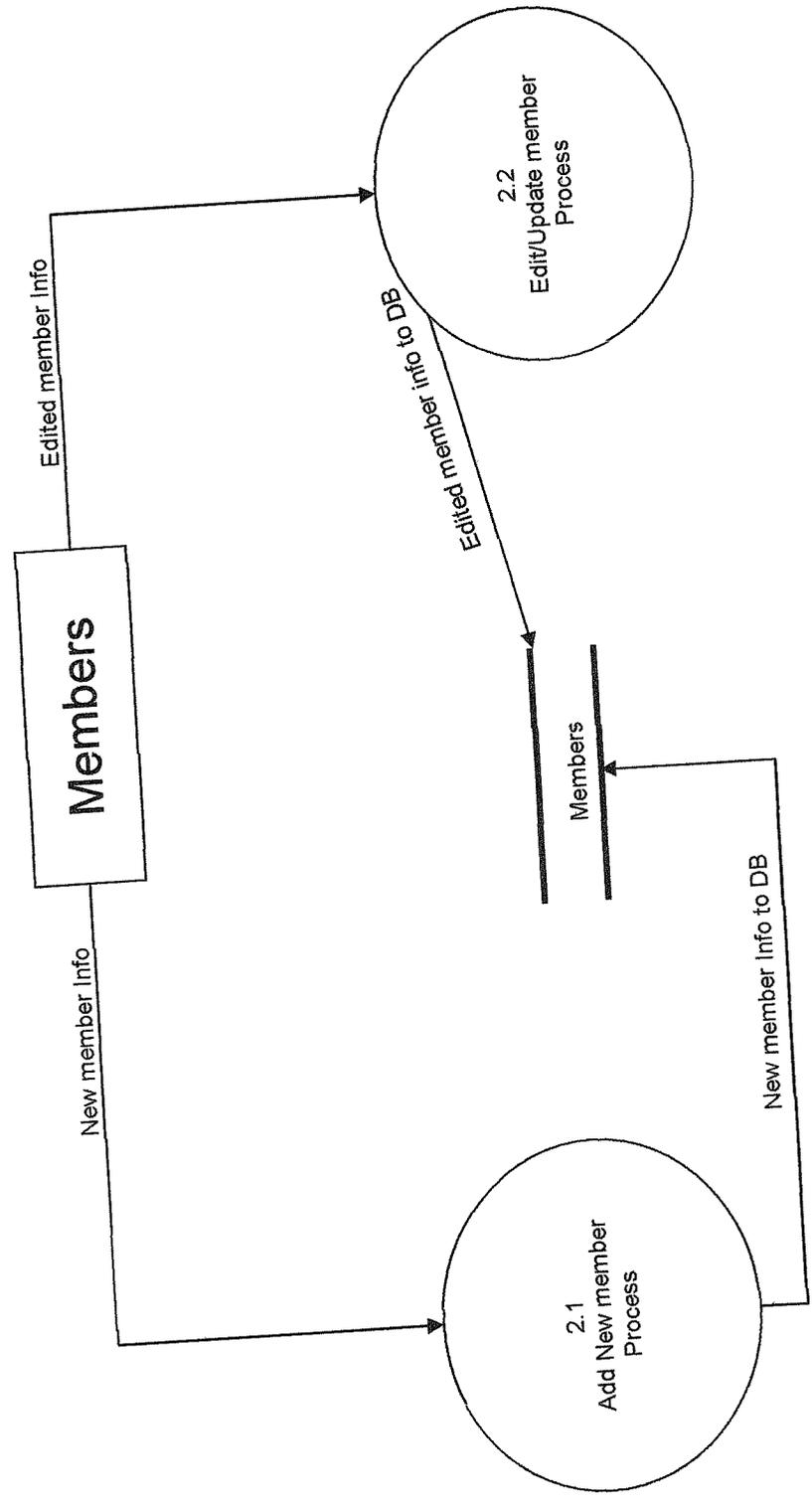
- 1. Members List
- 2. All Cars List
- 3. Reserved Cars List
- 4. Rented Cars List
- 5. Available Cars List
- 6. General Accounts Report
- 7. Member Account Report

Reports (1,2,3,4,5,6,7,8)

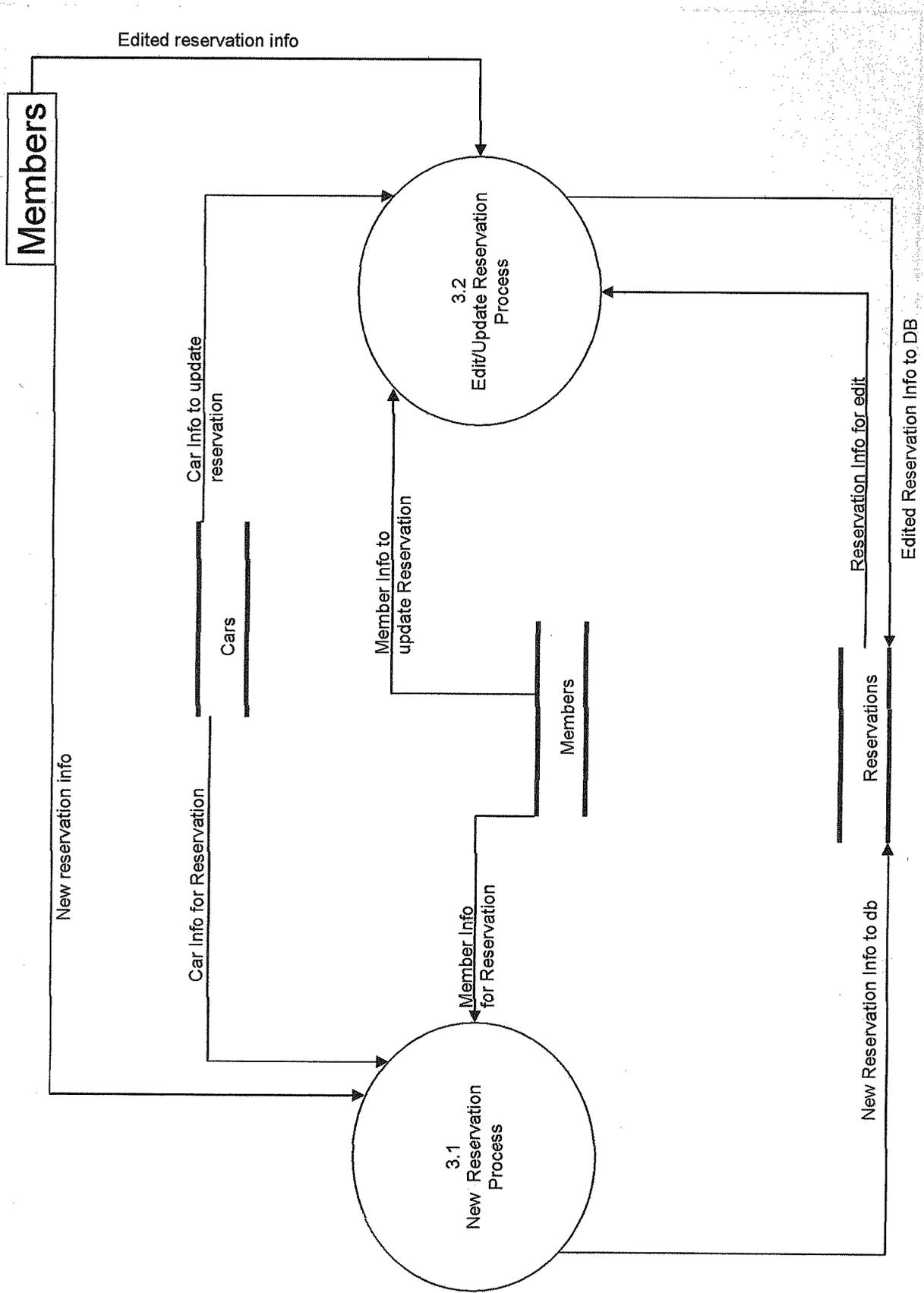
Detailed DFD for Car Process



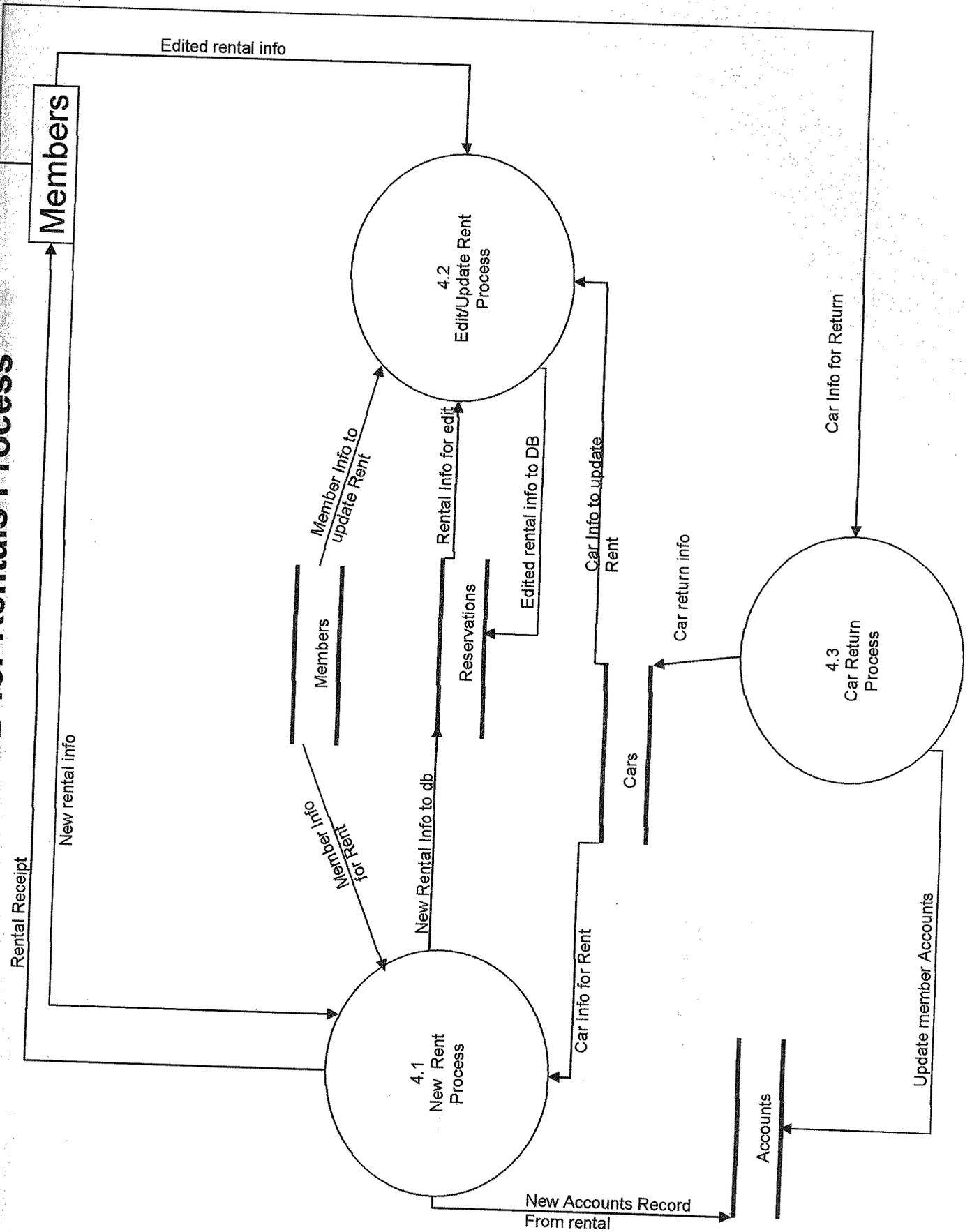
Detailed DFD for Member Process



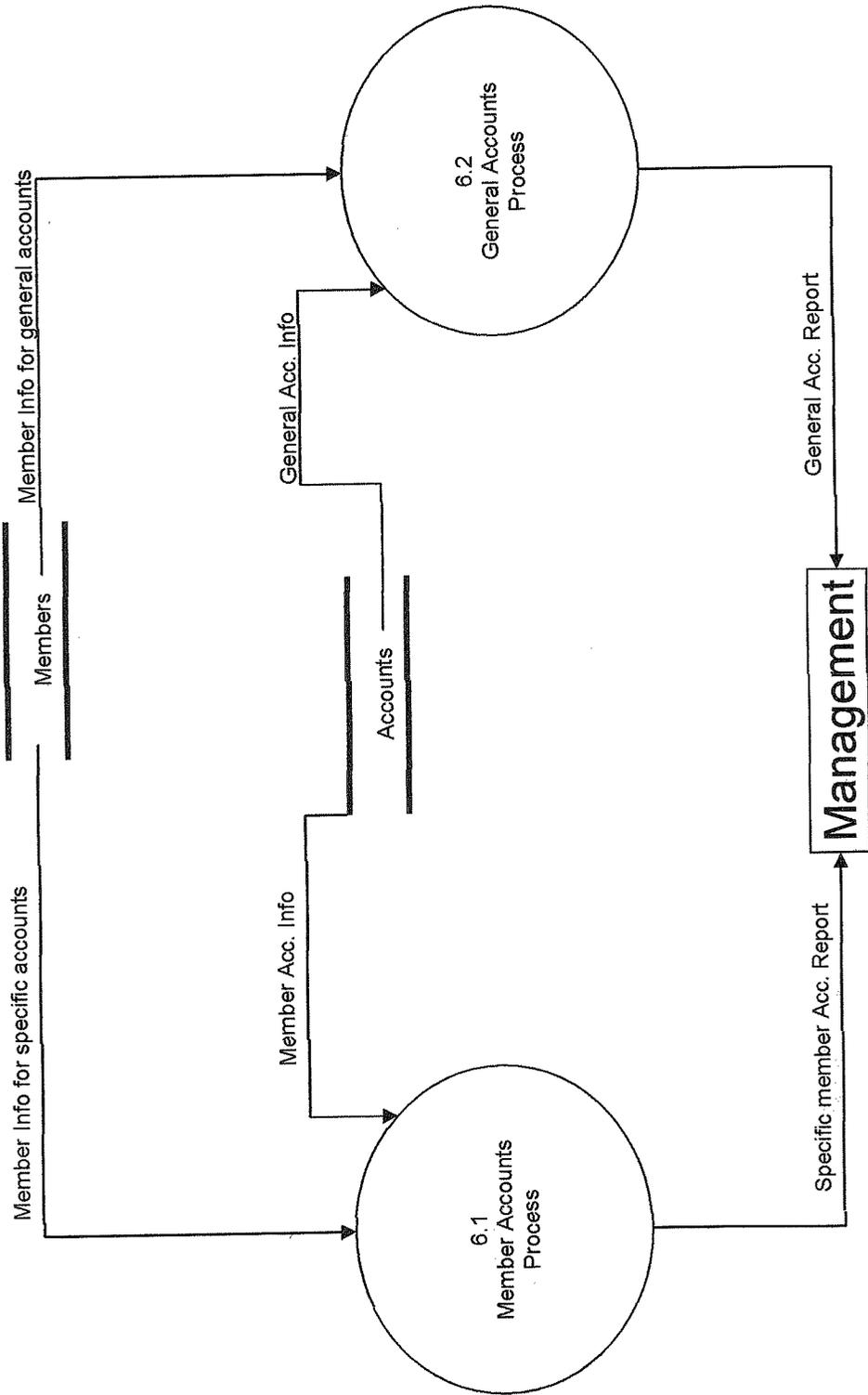
Detailed DFD for Reservation Process



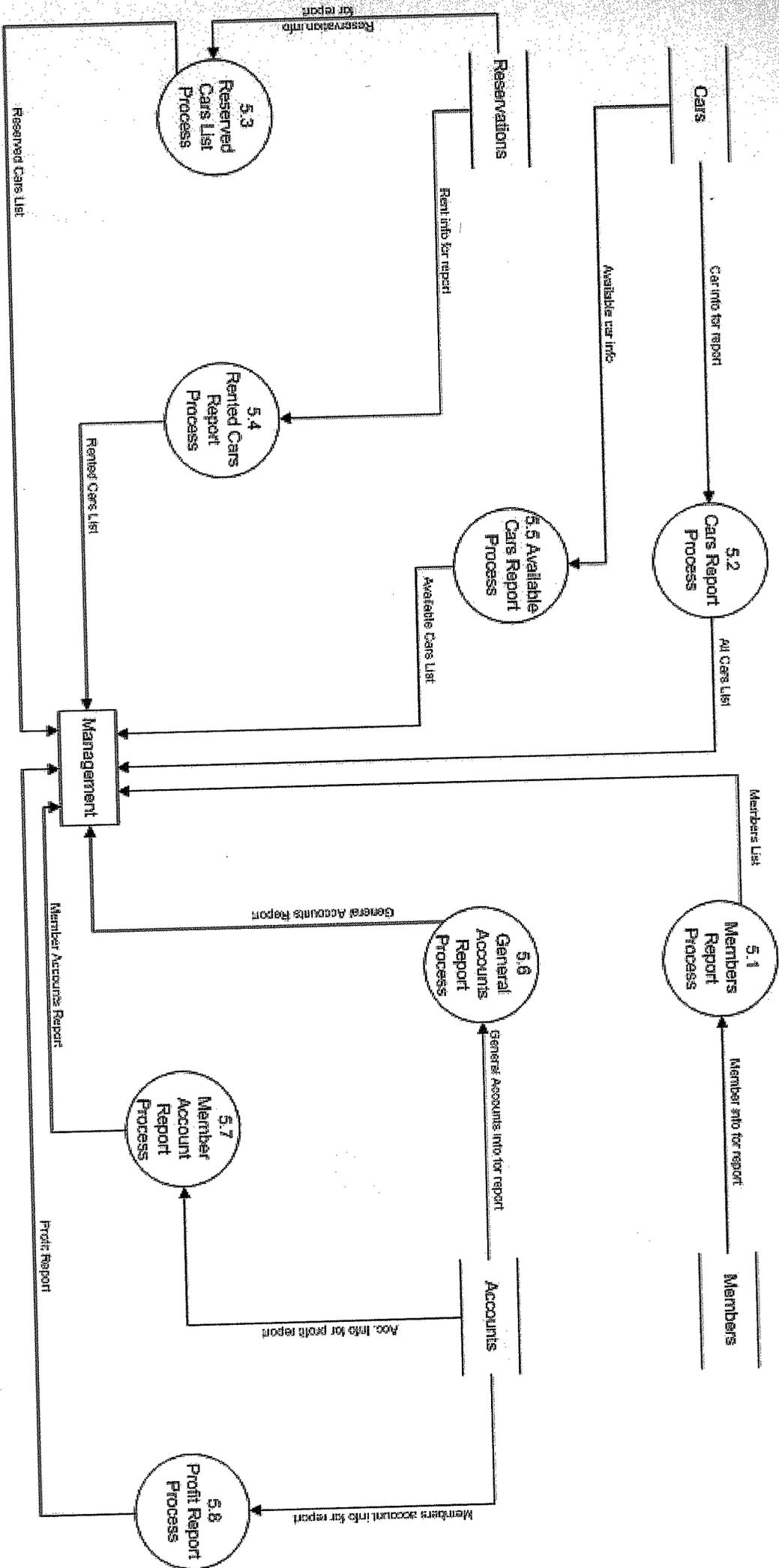
Detailed DFD for Rentals Process



Detailed DFD for Accounts Process



Detailed DFD for Report Process



System Design

Car Table

Field Name	Data Type
carreg	Text
made	Text
modal	Text
color	Text
extras	Memo
rent	Currency
rented	Yes/No
picture	Text
deactivate	Yes/No

Field Properties	
General	Lookup
Field Size	10
Format	
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	Yes (No Duplicate)
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Smart Tags	

Members Table

Field Name	Data Type
memid	AutoNumber
name	Text
sname	Text
fone	Text
licenseNo	Text
address	Text
NICNo	Text
email	Text

Field Properties	
General	Lookup
Field Size	Long Integer
New Values	Increment
Format	
Caption	
Indexed	Yes (No Dupli)
Smart Tags	

Reservation Table

reservations : Table

Field Name	Data Type
res_serial	AutoNumber
carreg	Text
memid	Number
date	Date/Time
fromdate	Date/Time
duration	Number
exp_return	Date/Time
canceled	Yes/No
rented	Yes/No
reserved	Yes/No
returned	Yes/No

Field Properties

General Lookup

Field Size	Long Integer
New Values	Increment
Format	
Caption	
Indexed	Yes (No Duplicates)
Smart Tags	

Accounts Table

accounts : Table

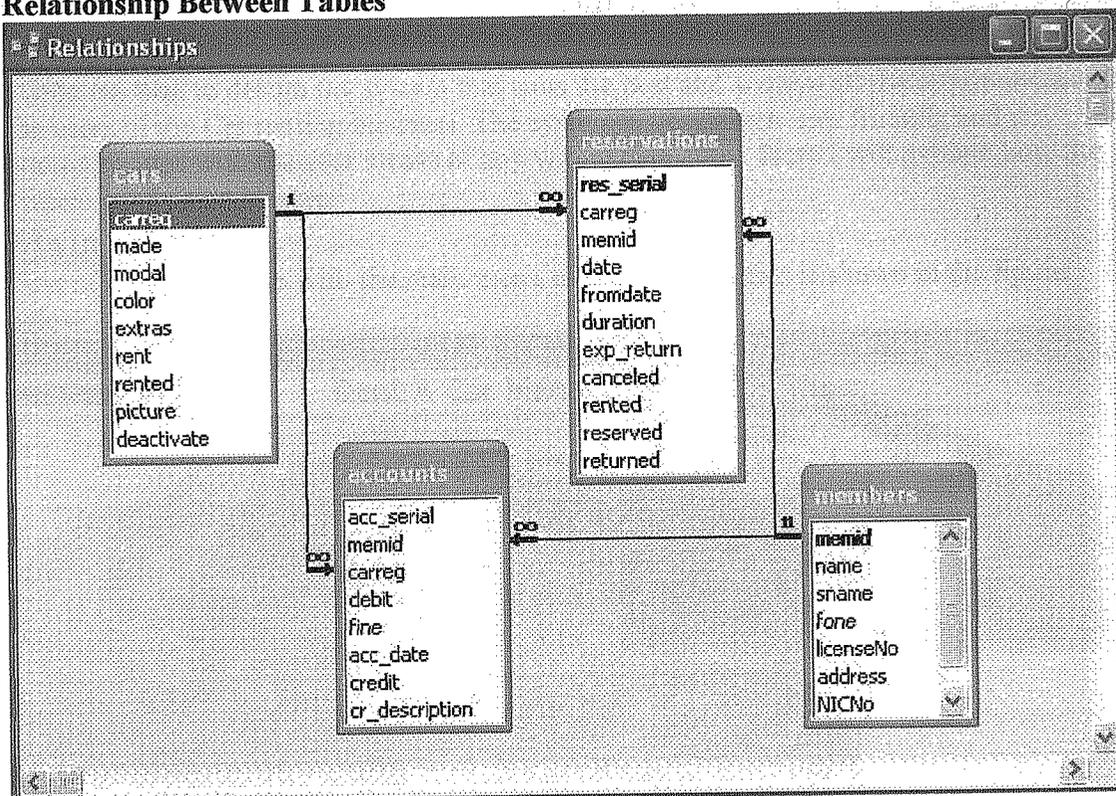
Field Name	Data Type
acc_serial	Number
memid	Number
carreg	Text
debit	Currency
fine	Currency
acc_date	Date/Time
credit	Currency
cr_description	Text

Field Properties

General Lookup

Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	
Default Value	0
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (Duplicates O
Smart Tags	

Relationship Between Tables



User Guidelines

I tried my best to design my system that suits the best and acquire all the requirements of the users of the Car renting business. The system is very much user friendly and it communicates with the user incase of any wrong data entry. Here I will explain how the system works step by step.

Main Menu

First of all I will begin with the main menu of the program that is shown in the figure below

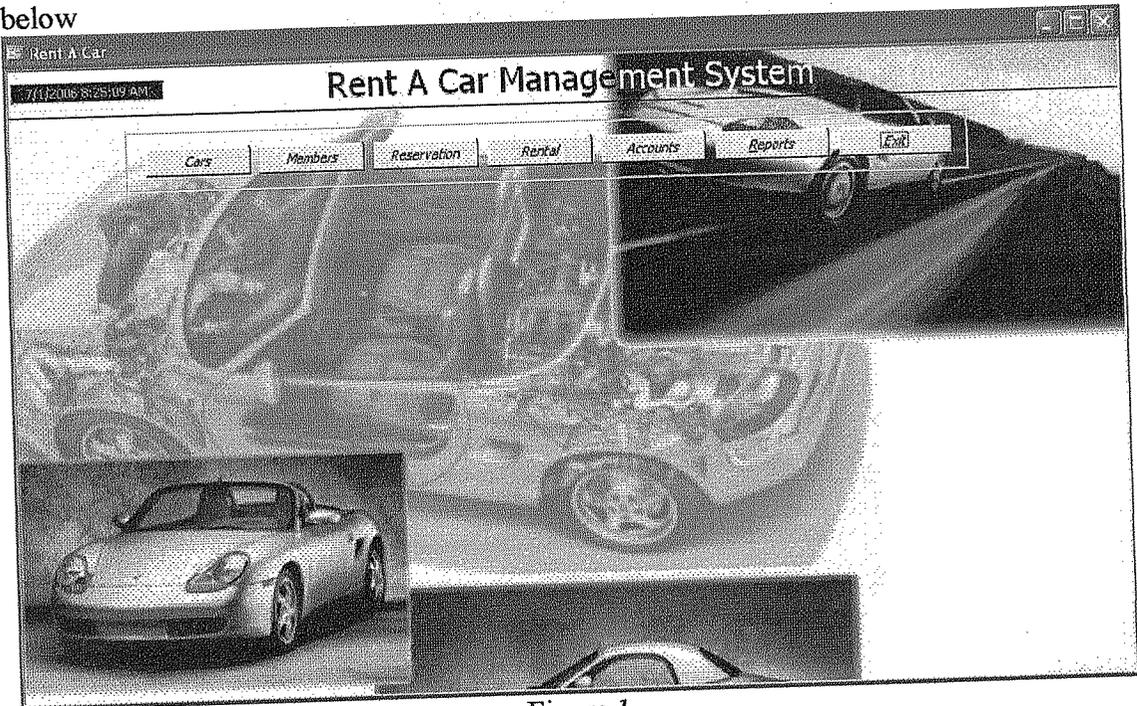


Figure 1

Figure 1 shows the main options of Rent a car program. It has 6 main processes with one extra exit option.

1. Cars
2. Members
3. Reservations
4. Rentals
5. Reports
6. Accounts
7. Exit (to log out of the system)

I will explain each of the above main processes one by one how to operate them.

Cars

This is the first main process of the system which contains two sub processes as shown in figure 2 below

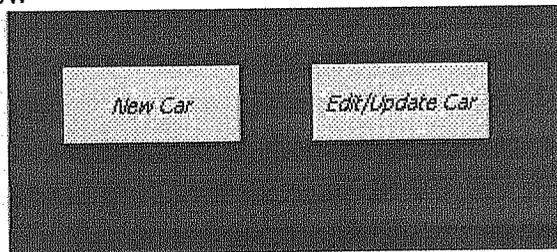


Figure 2

Each of the above process is explained how it works and what does it do one by one

New Car

The screenshot shows a window titled 'New Car Data Entry Form' with a sub-header 'New Car Registration'. On the left, there is a background image of a tropical beach with palm trees. Overlaid on this image is a form with the following fields: 'Car Registration No.', 'Made', 'Model', 'Color', 'Extras', and 'Rent Price'. Below these fields are three buttons: 'New', 'Save', and 'Back'. On the right side of the window, there is a large empty rectangular area. Below this area is a button labeled 'Add/Change picture'. A text box above the 'Add/Change picture' button contains the instruction: 'Click "Add/Change" to add or change picture'.

Figure 3

This is the very first process where we enter the data about the cars that we are going to rent for the first time. In the corresponding fields we enter the specific data such as car registration number, its model and made, color and rent price etc. We have then three buttons to choose the process what the program should do. To save the data click the

“Save” button and the information will be stored in the database for future. To enter another car data click “New” or click “Back” to go back to the previous menu.

Edit Car

Car Registration No.	4
Make	HONDA
Model	CIVIC
Color	Blue
Extras	CD,AC,TURBO
Rent Price	23
Deactivate	<input type="checkbox"/>

Buttons: *New*, *Save Changes*, *Find*, *Back*

Image placeholder: *Click to add/change picture*

Figure 4

On this form after choosing the edit car option from the sub menu, we enter the car registration number in its corresponding and then click “Find”, if the record with such registration number exists in the database its data will be displayed in rest of the fields otherwise the user will be prompted to enter the car registration number again with an error message.

Members

Buttons: *New Member*, *Edit/Update Member*

Figure 5

Like the cars sub menu this sub menu contains two sub processes as well. Each is described in detail below

New Member



Figure 6

In figure 6 we can see that this process is similar to the new car data entry form. The only difference here is that we enter a member's data or new customer's data instead of the car information. Click "Save" in order to save the record in the database, "New" to enter a new member's data and "Back" to go back to the previous sub menu.

Edit Member

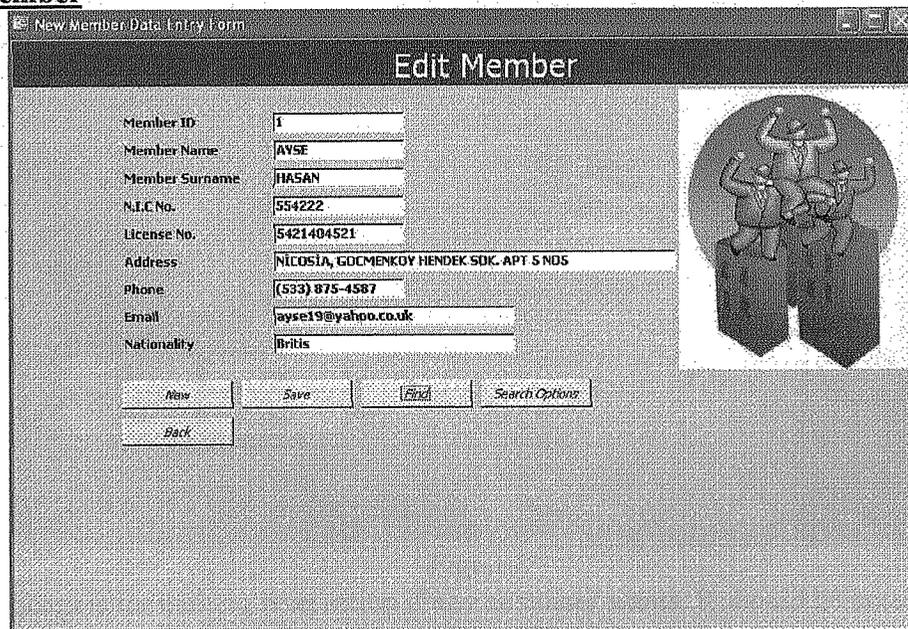


Figure 7

On this form we enter member id if we know and click "Find", the required information will be retrieved from the database and will be displayed in their relative text fields. If the member ID is not known there is an option to search for a member using his name or surname as query parameter. To do so click "Search Options" and a small form will pop

up enter the name or surname as the query parameter and click "Go" if any data exists in the database as a result of the query it will be displayed as a list, double click the required member ID and then click "Find" button to retrieve the record, make any changes as u desire and then click save again, to search for another member click "New" or "Back" to go back to the previous menu.

Reservations

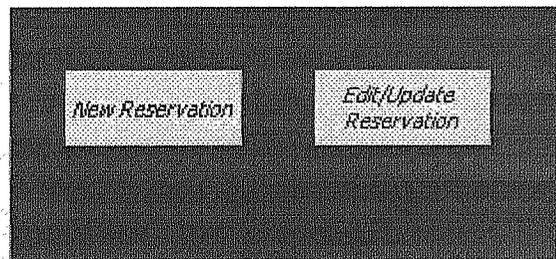


Figure 8

This process also consists of two sub processes. Each is explained below

New Reservation

Figure 9

This process is for making reservation if any customer wants. It should be noted here that the reservation can be made for registered user. If he is not registered, user must register him before he can make reservation through new member's process. In the member ID field enter the member ID who wants to make reservation, if the data is found in the database his information will be displayed, similarly the car registration number is given to the form process and then the reservation date and the duration of the renting, if there

comes out any conflict in the dates of the reservation made before user will be prompted for that and no reservation will be made after click the "Save" button. Once the reservation is made successfully a reservation ID will be shown on the screen, user must remember that ID in order to update the reservation which can be done easily by printing the reservation slip by clicking "Print" button. Use "Back" button to go to previous menu.

Edit Reservation

Figure 10

This is the process where we are allowed to update the reservation to rent the car or cancel it simply. A reservation ID is given to the form and if that is found active will be displayed with the information on the form. Choose appropriate option for specific process.

Rentals

This is similar to the reservation sub process containing a sub menu with three options as shown in figure 11

Figure 11

New Rent

This process is just similar to the new reservation on the form, the only difference between them is that here directly a car is rented to the customer (member) and his accounts are updated.

Member Information

Car Information

Rent Information

Sun	Mon	Tue	Wed	Thu	Fri	Sat
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Figure 12

Edit Rent

Here a user can edit his rental information similar to the edit reservation process as explained above

Member Information

Car Information

Reservation Information

Figure 13

Reports

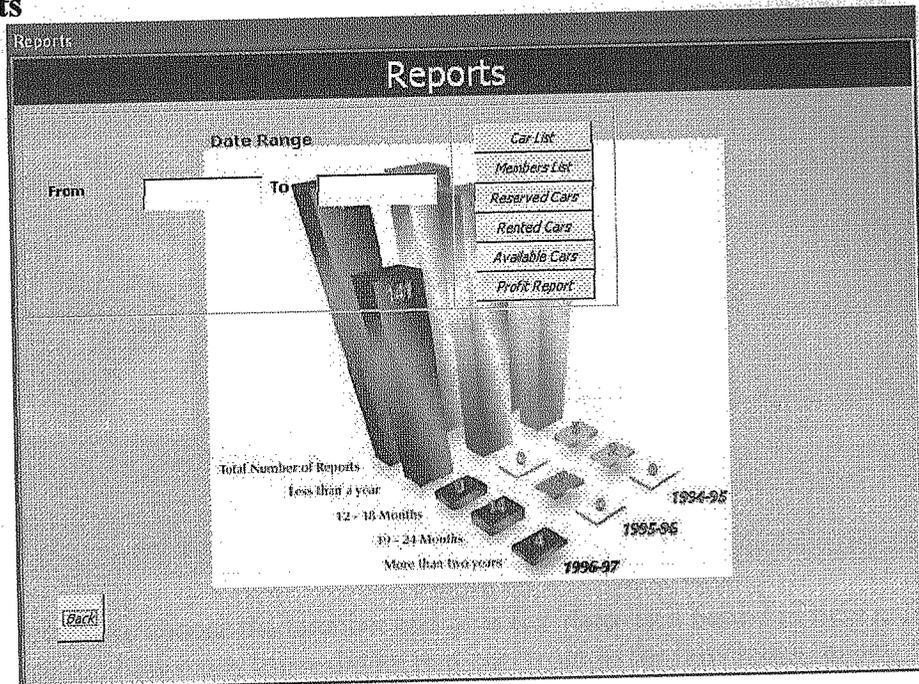


Figure 14

This is a process for managerial purpose. All reports are printed through this process except accounts process. Choose the appropriate button for the desired report, For profit reports or reservation or renting report or member between a period of time you must give a period of two dates.

Accounts

This process consists of two sub processes as shown in the figure

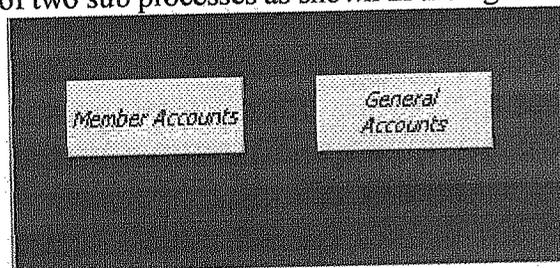
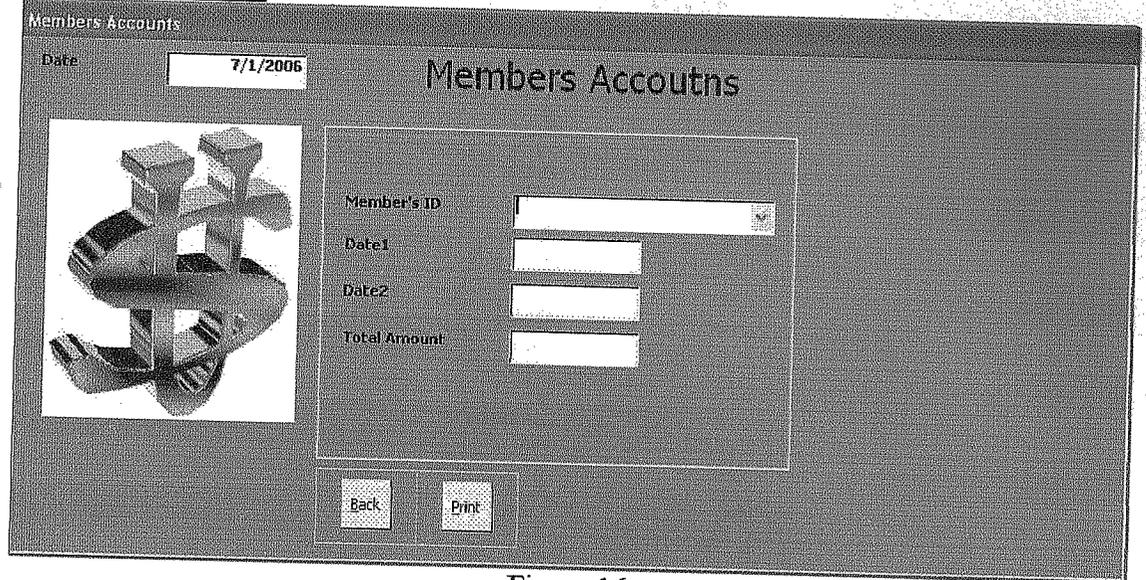


Figure 15

Member Accounts



Members Accounts

Date 7/1/2006

Members Accounts

Member's ID

Date1

Date2

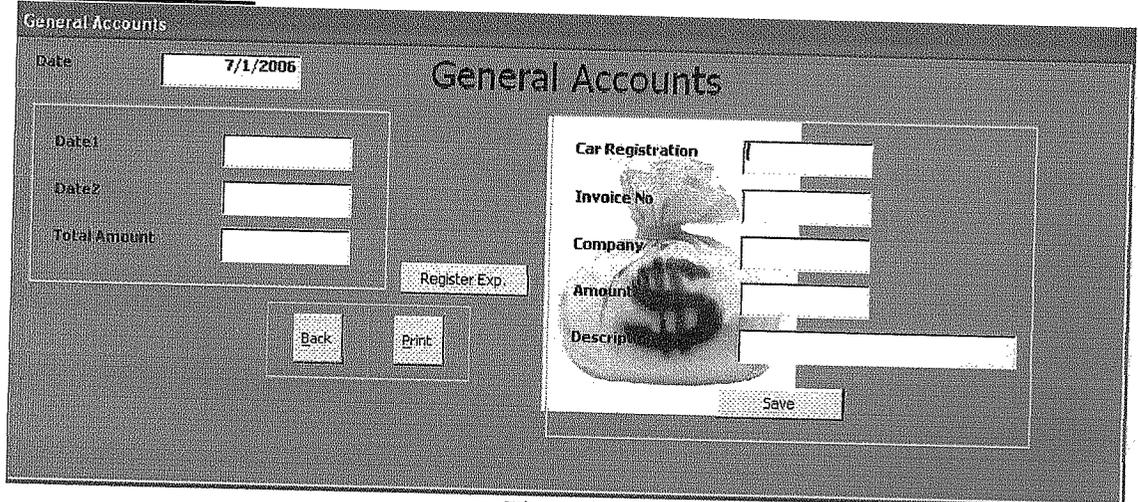
Total Amount

Back Print

Figure 16

In this process we choose the member and give two dates as a period of time and we get the information how many times did he rent cars from us for how much money. The report can be printed by clicking the "Print" button.

General Accounts



General Accounts

Date 7/1/2006

General Accounts

Date1

Date2

Total Amount

Car Registration

Invoice No

Company

Amount

Description

Register Exp.

Back Print

Save

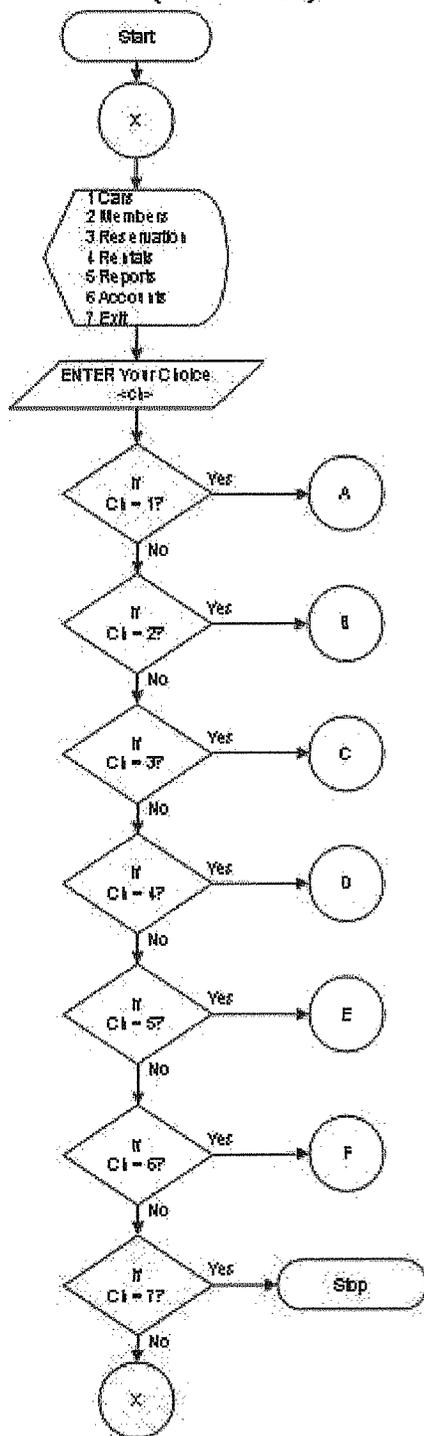
Figure 17

In this process we simply give two dates and get who ever rented car between those two dates with the amount of money they paid each time. Same report can be printed as well.

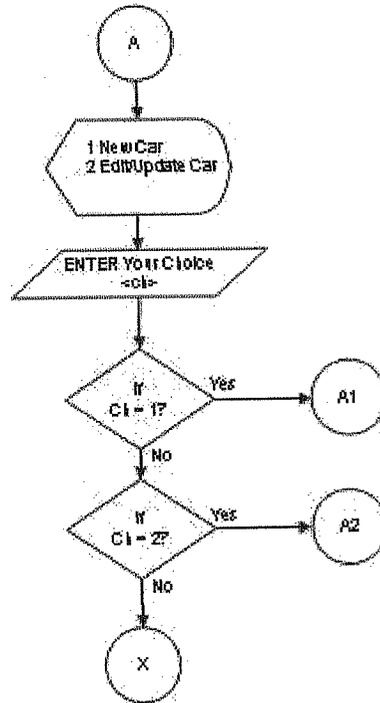
I hope this user manual will help all the users of my program.

User Flowcharts

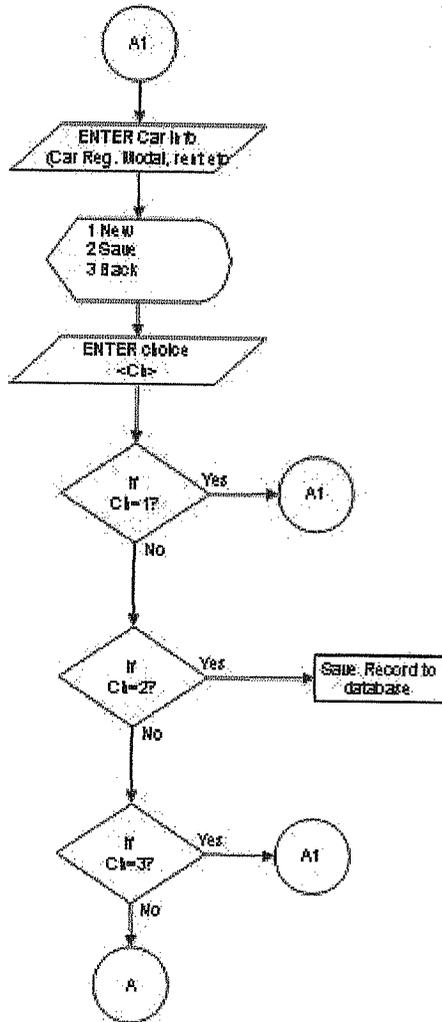
MAIN MENU (MAIN PAGE)



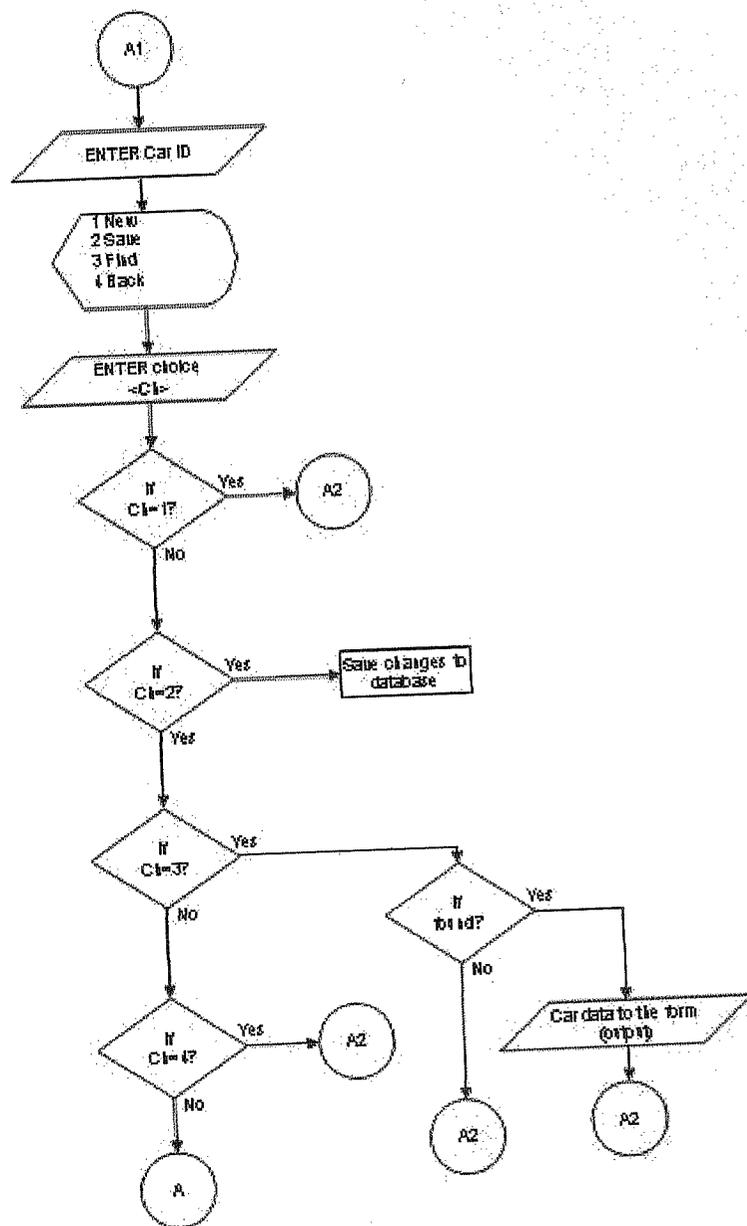
Cars Sub Menu



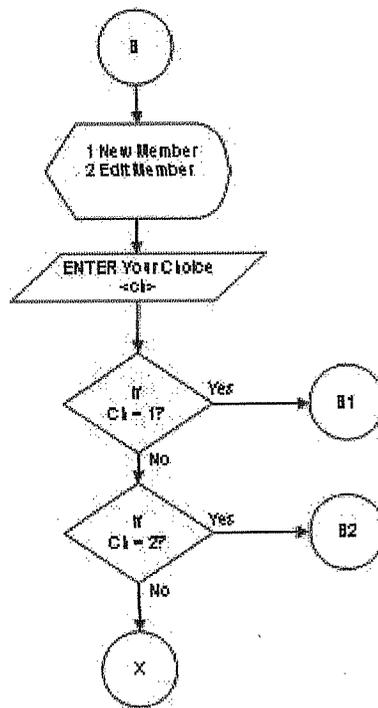
New Car Registration



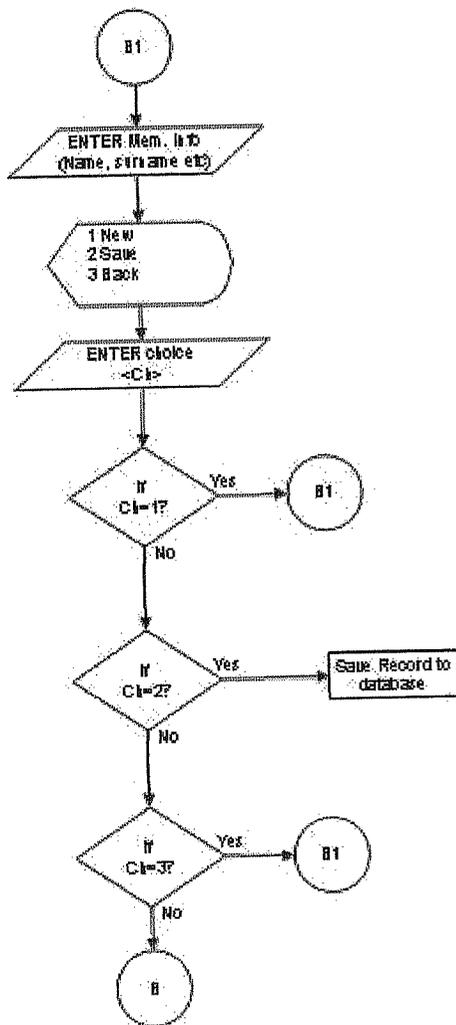
Edit Car Specifications



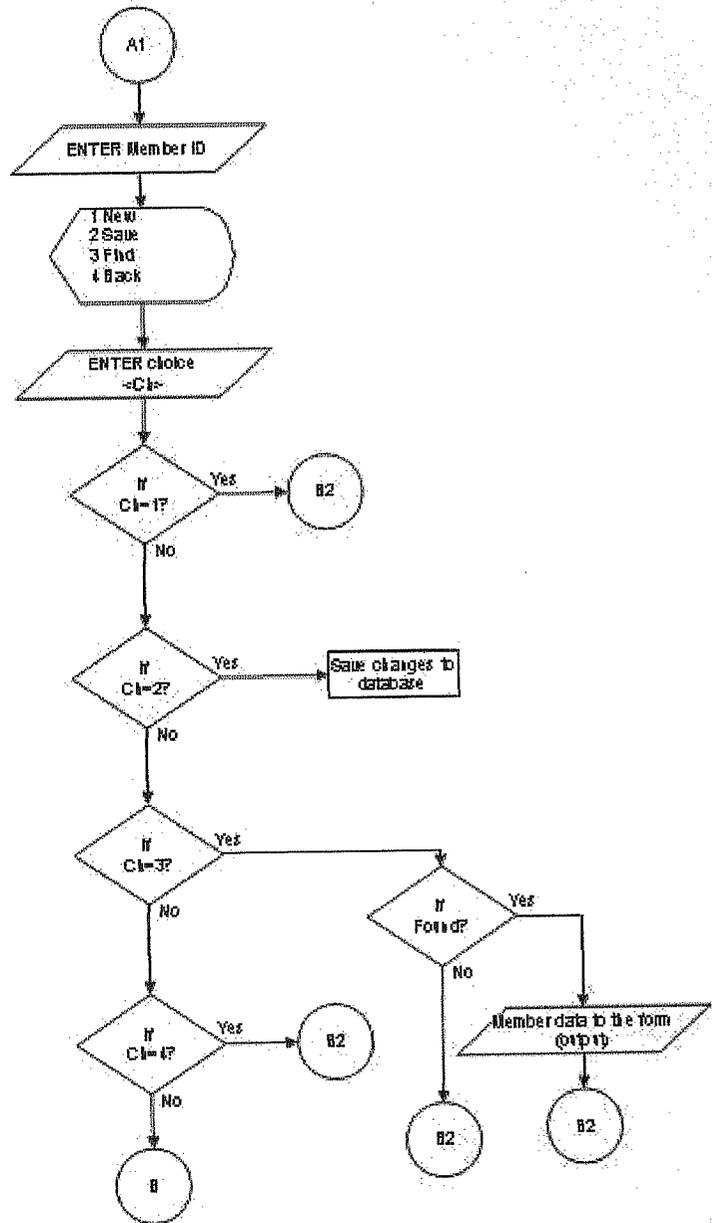
Members Sub Menu



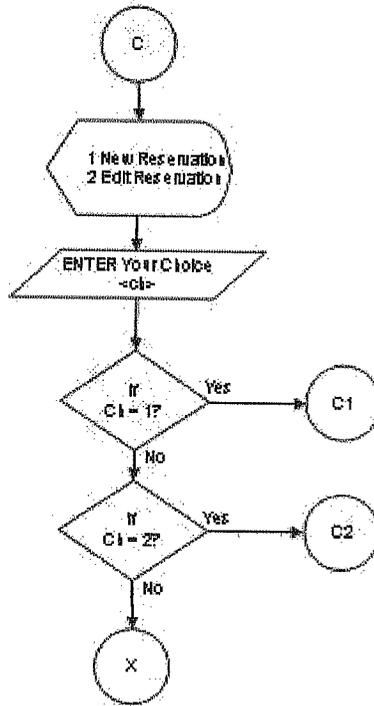
New Member Registration



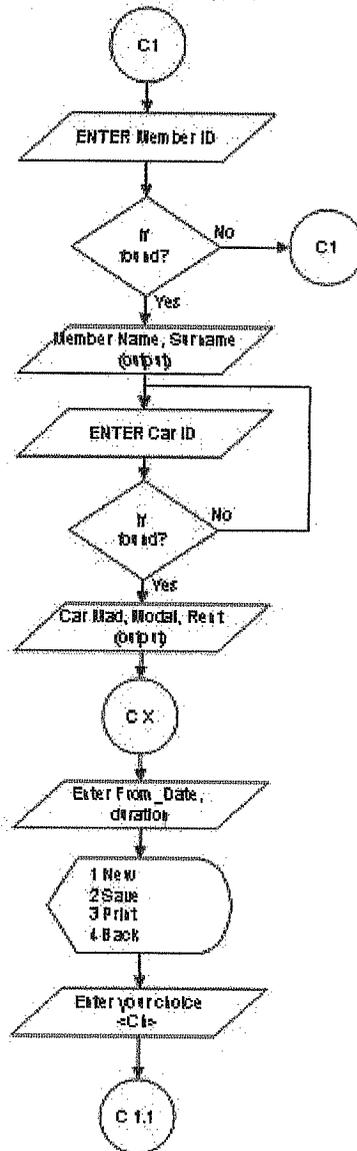
Edit Member Specifications



Reservation Sub Menu

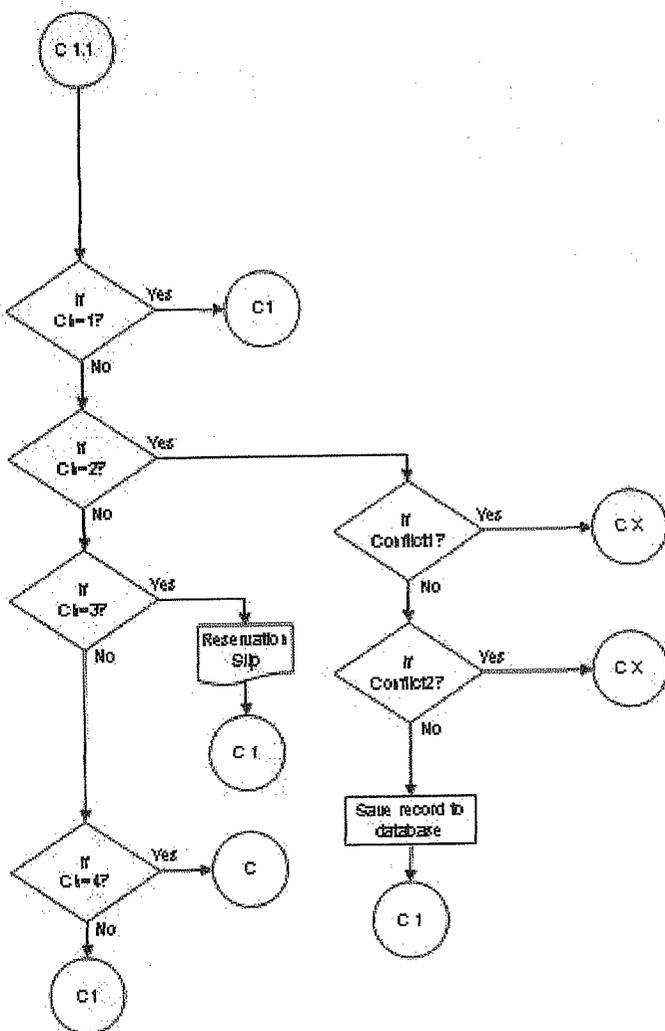


New Reservation

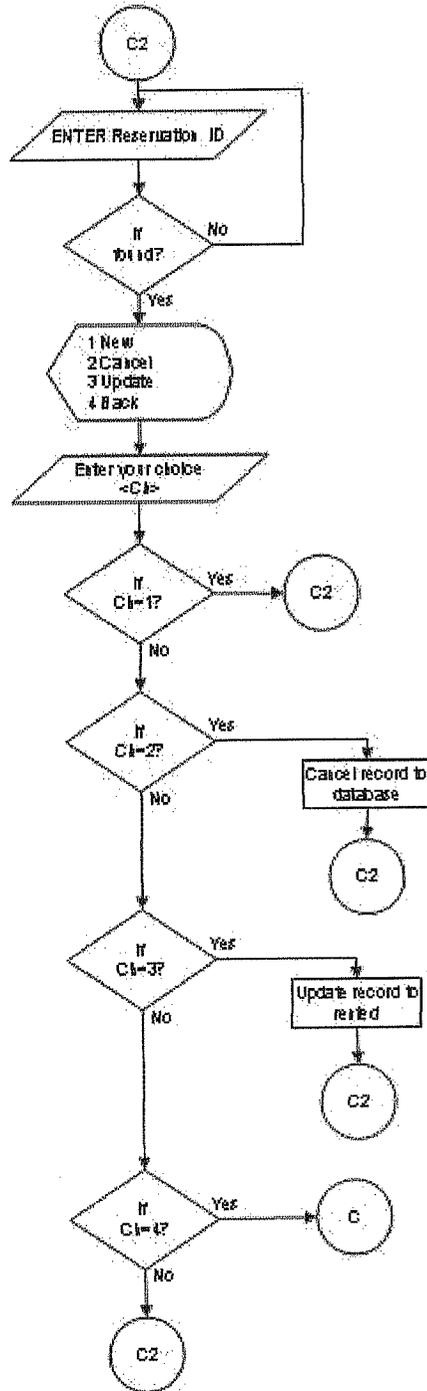


New Reservation continue

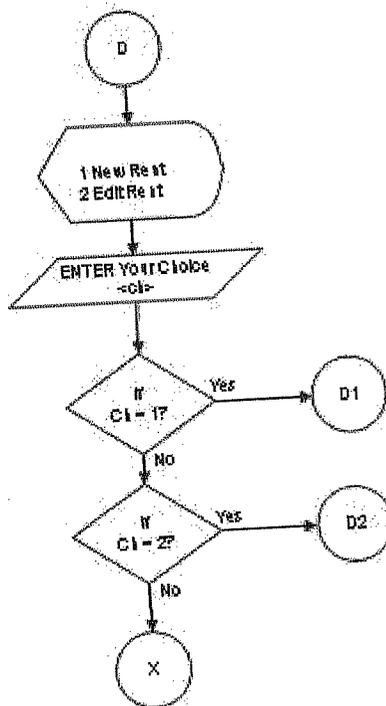
Note:
Conflict1
 If From_Date - Today's Date?
Conflict2
 If From_date or Retire date lies between any Active Reservation?



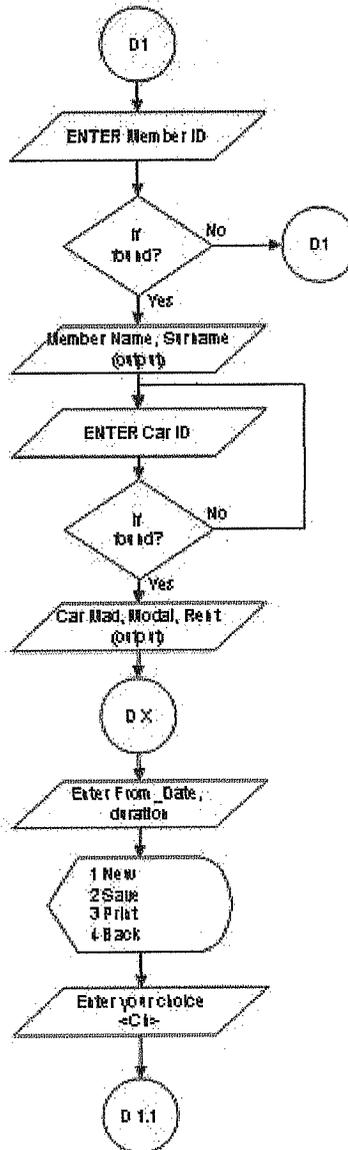
Edit Reservation



Rent Sub Menu

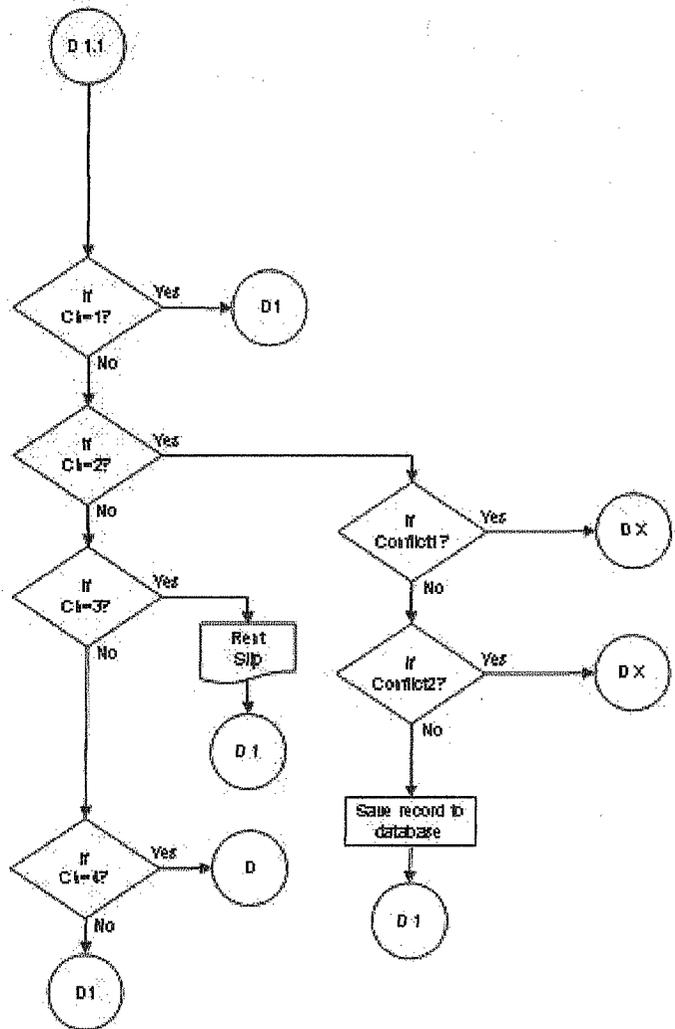


New Rent

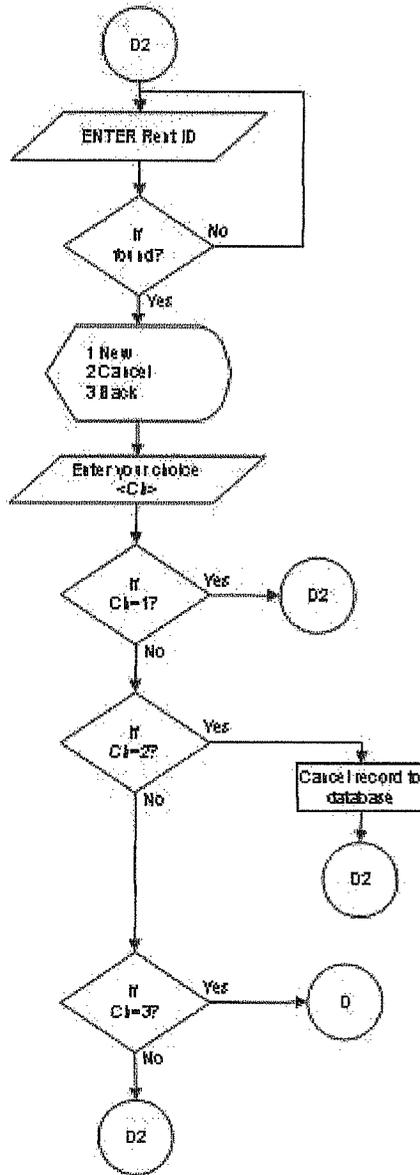


New Rent continue

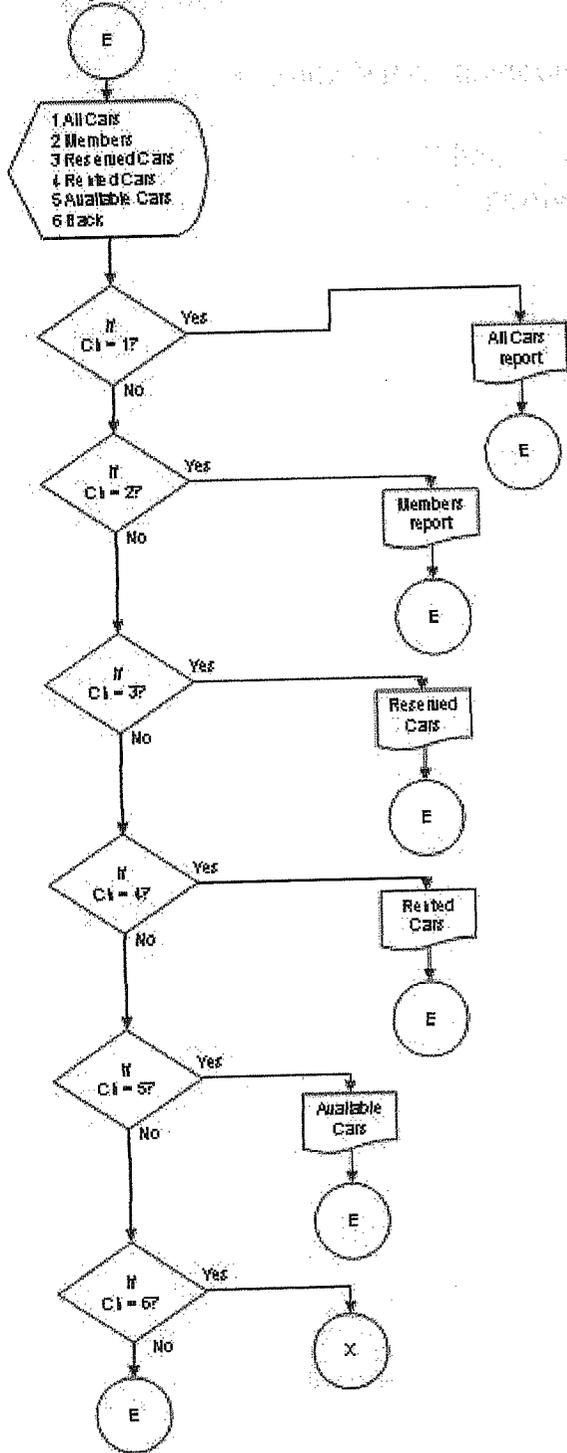
Note:
Conflict1
 If From_Date < Today's Date?
Conflict2
 If From_date or Retri date lies
 between any Active Reservation?



Edit Rent



Reports

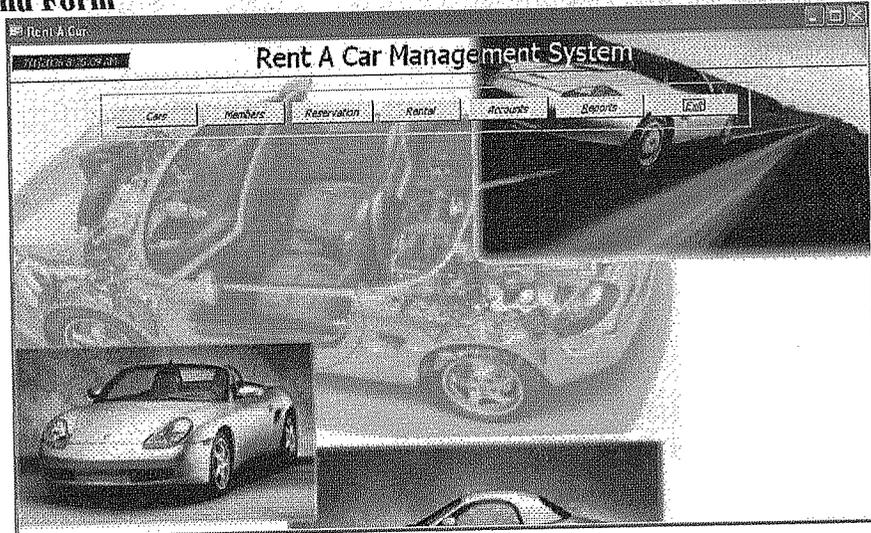


Reference

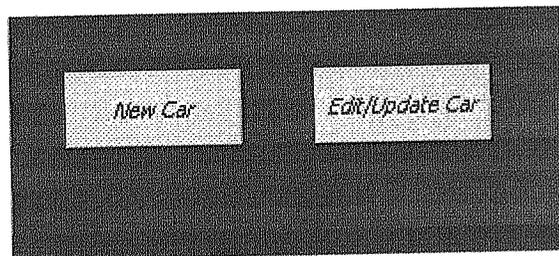
1. Jones Dana and Novalis Susan, "Access 2002 VBA Handbook", published by SYBEX, 2002.
2. Microsoft Office Professional 2002 for windows XP help.
3. Suraski, Zeev, "Basics of Visual Basic", edition No. 1, published by PEARSON, 2001.
4. www.access-programmers.com
5. <http://www.microsoft.com/mspress/books/5054.asp>
6. www.utteraccess.com

Screen Outputs

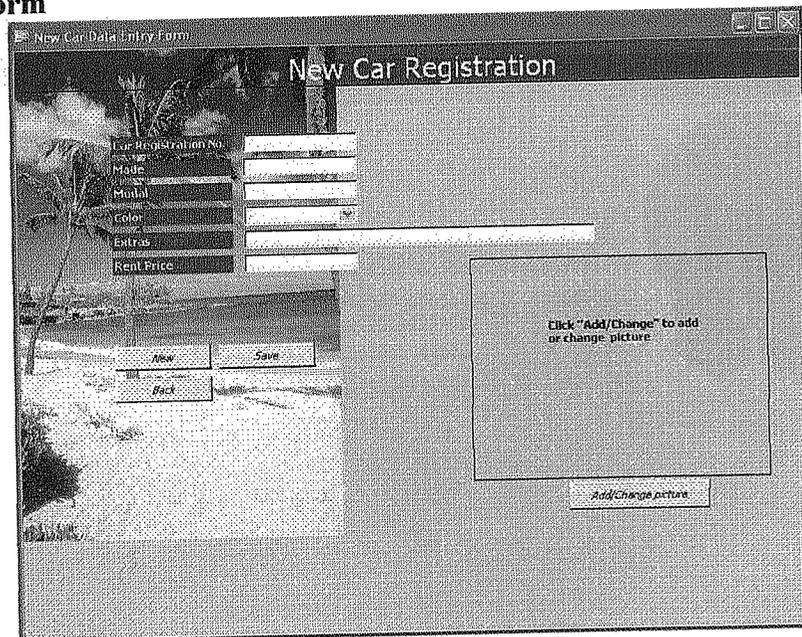
Main Menu Form



Car Submenu Form



New Car Form



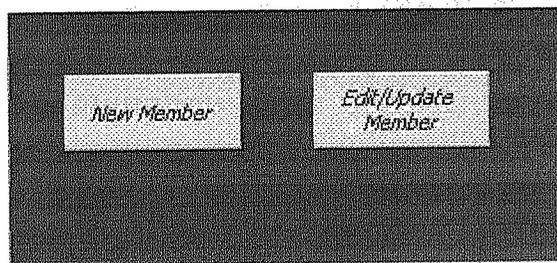
Edit Car Form

The screenshot shows a web browser window titled "Edit Car Data Form" with a sub-header "Edit Car Information". The form contains the following fields and values:

Car Registration No.	4
Make	HONDA
Model	CIVIC
Color	Blue
Extras	CD,AC,TURBO
Rent Price	23
Deactivate	<input type="checkbox"/>

Navigation buttons include "New", "Save Changes", "Find", and "Back". A photograph of a silver Honda Civic is displayed on the right, with a caption "Click to add/change picture".

Members Submenu



New Member Form

The screenshot shows a web browser window titled "New Member Data Entry Form" with a sub-header "New Member Registration". The form contains the following fields:

- Member Name
- Member Surname
- N.I.C. No.
- License No.
- Address
- Phone
- Email
- Nationality

Navigation buttons include "New", "Save", and "Back". An image of two muscular men flexing their muscles is shown on the right side of the form.

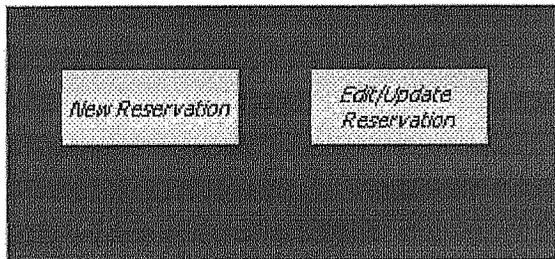
Edit Member Form

Edit Member

Member ID	1
Member Name	AYSE
Member Surname	HASAN
N.I.C. No.	554222
License No.	5421404521
Address	NICOSIA, GOCMENKOV HEDEK SOK. APT 5 N05
Phone	(533) 875-4587
Email	ayse19@yahoo.co.uk
Nationality	British

Buttons: New, Save, **Print**, Search Options, Back

Reservation Submenu



New Reservation Form

New Reservation

Member Information Member ID: 1 Member Name: AYSE Member Surname: HASAN N.I.C. No.: 554222 License No.: 5421404521		Car Information Car Registration No.: 4 Make: HONDA Model: CIVIC Color: BLUE Rent Price: 23	
Reservation Information Today: 7/1/2006 From Date: 7/1/2006 Duration: [Field] Expected Return: [Field] Generated Res. ID: [Field]			

Buttons: New, Save, Back, Print

Edit Reservation Form

Edit Reservation

Member Information

Reservation ID: 254
 Member ID: 1
 Member Name: AHMED
 Member Surname: HASAN
 N.I.C No.: 455462
 License No.: 45518

Car Information

Car Registration No.: 19
 Make: HONDA
 Model: CIVIC
 Color: GREEN
 Rent Price: 35

Reservation Information

Today: 7/1/2006
 From Date: 6/19/2006
 Duration: 3
 Expected Return: 6/20/2006

Buttons: New, Update, Back, Cancel

Rental Submenu

Buttons: New Rent, Edit/Update Rent, Return Car

New Rent Form

New Rent

Member Information

Member ID:
 Member Name:
 Member Surname:
 N.I.C No.:
 License No.:

Car Information

Car Registration No.:
 Make:
 Model:
 Color:

Reservation Information

Today: 7/1/2006

Calendar: Jul 2006

Sun	Mon	Tue	Wed	Thu	Fri	Sat
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Buttons: New, Back

Edit Rent Form

Reports Form

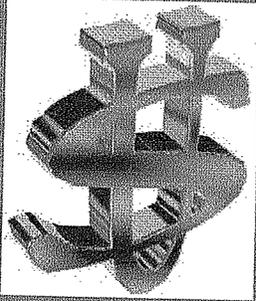
Accounts Submenu

Members' Accounts Form

Members Accounts

Date:

Members Accountns



Member's ID:

Date1:

Date2:

Total Amount:

General Accounts Form

General Accounts

Date:

General Accounts

Date1:

Date2:

Total Amount:

Car Registration

Invoice No:

Company:

Amount:

Description:

Source Codes of the program

PROGRAM CODES

MAIN MENU

Accounts Sub Menu

```
Private Sub cmdaccounts_Click()  
Me.sub_menu.Visible = True  
Me.sub_menu.SourceObject = "subaccmenu"  
End Sub
```

Cars Sub Menu

```
Private Sub cmdcars_Click()  
Me.sub_menu.Visible = True  
Me.sub_menu.SourceObject = "subcarsmenu"  
End Sub
```

Exit Button

```
Private Sub cmdExit_Click()  
If MsgBox("Do you want to log out", vbYesNo, Me.Caption) = vbYes Then  
    DoCmd.Close acForm, "Main_menu"  
    Quit  
End If  
End Sub
```

Members Sub Menu

```
Private Sub cmdmembers_Click()  
Me.sub_menu.Visible = True  
Me.sub_menu.SourceObject = "submembmenu"  
End Sub
```

Rentals Sub Menu

```
Private Sub cmdrentals_Click()  
Me.sub_menu.Visible = True  
Me.sub_menu.SourceObject = "subrentmenu"  
End Sub
```

To Open Reports Form

```
Private Sub cmdreports_Click()  
DoCmd.OpenForm "report"  
End Sub
```

Reservation Sub Menu

```
Private Sub cmdreservation_Click()  
Me.sub_menu.Visible = True  
Me.sub_menu.SourceObject = "subresmenu"  
End Sub
```

```

Sub Form Invisible Code
Private Sub Detail_Click()
Me.cmdcars.SetFocus
Me.sub_menu.Visible = False
End Sub

```

SUB FORM CARS

```

To Open New Car Data Entry Form
Private Sub Command0_Click()
DoCmd.OpenForm "NewCar"
End Sub

```

```

To Open Edit Car Data Form
Private Sub Command1_Click()
DoCmd.OpenForm "EditCar"
End Sub

```

NEW CAR FORM

```

Option Compare Database
Public db As DAO.Database
Public rs As DAO.Recordset
Public s As String

```

```

To Go Back To Main Menu
Private Sub cmdBack_Click()
DoCmd.Close acForm, "NewCar"
End Sub

```

```

To Initial All Form Fields
Private Sub cmdNew_Click()
Me.txtcarreg = "": Me.txtmade = ""
Me.txtcolor = "": Me.txtrent = ""
Me.txtextras = "": Me.txtmodal = ""
Me.txtcarreg.SetFocus
End Sub

```

```

Save Button
Private Sub cmdSave_Click()
Set db = CurrentDb()
s = "SELECT * FROM cars WHERE carreg=" & Me.txtcarreg & ""
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
rs.AddNew

```

```

rs.Fields("carreg").Value = Me.txtcarreg
rs.Fields("made").Value = Me.txtmade
rs.Fields("modal").Value = Me.txtmodal
rs.Fields("color").Value = Me.txtcolor
rs.Fields("extras").Value = Me.txtextras
rs.Fields("rent").Value = Me.txtrent
rs.Update
MsgBox ("New record of a car has been added to the database"), vbInformation,
Me.Caption
Else
MsgBox ("This car registration number already exists in the database"), vbCritical,
Me.Caption
End If
rs.Close: Set rs = Nothing
db.Close: Set db = Nothing
End Sub

```

EDIT FORM

```

Option Compare Database
Public db As DAO.Database
Public rs As DAO.Recordset
Public s As String

```

```

To Go Back to Main Menu
Private Sub cmdBack_Click()
DoCmd.Close acForm, "EditCar"
End Sub

```

```

Find Button
Private Sub cmdFind_Click()
Set db = CurrentDb()
s = "SELECT * FROM cars WHERE carreg='" & Me.txtcarreg & "'"
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
MsgBox ("This car registration number does not exists in the database"), vbCritical,
Me.Caption
Me.txtcarreg.SetFocus
Else
Me.txtmade = rs.Fields("made").Value
Me.txtmodal = rs.Fields("modal").Value
Me.txtcolor = rs.Fields("color").Value
Me.txtextras = rs.Fields("extras").Value
Me.txtrent = rs.Fields("rent").Value
End If
rs.Close: Set rs = Nothing

```

```
db.Close: Set db = Nothing
End Sub
```

```
Private Sub cmdNew_Click()
Me.txtcarreg = "": Me.txtmade = ""
Me.txtcolor = "": Me.txtrent = ""
Me.txtextras = "": Me.txtmodal = ""
Me.txtcarreg.SetFocus
End Sub
```

Save Changes Button

```
Private Sub cmdSave_Click()
Set db = CurrentDb()
s = "SELECT * FROM cars WHERE carreg=" & Me.txtcarreg & ""
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
MsgBox ("This car registration number does not exists in the database"), vbCritical,
Me.Caption
Me.txtcarreg.SetFocus
Else
rs.Edit
rs.Fields("made").Value = Me.txtmade
rs.Fields("modal").Value = Me.txtmodal
rs.Fields("color").Value = Me.txtcolor
rs.Fields("extras").Value = Me.txtextras
rs.Fields("rent").Value = Me.txtrent
rs.Update
MsgBox ("Record has been updated in the database"), vbInformation, Me.Caption
End If
rs.Close: Set rs = Nothing
db.Close: Set db = Nothing
End Sub
```

MEMBER SUB MENU FORM

To Open New Member Data Entry Form

```
Private Sub Command0_Click()
DoCmd.OpenForm "NewMember"
End Sub
```

To Open Edit Member Data Form

```
Private Sub Command1_Click()
DoCmd.OpenForm "EditMember"
End Sub
```

NEW MEMBER FORM

Option Compare Database

```
Public db As DAO.Database
Public rs As DAO.Recordset
Public s As String
```

To Go Back To Main Menu

```
Private Sub cmdBack_Click()
DoCmd.Close acForm, "NewMember"
End Sub
```

Form Field Initializing

```
Private Sub cmdNew_Click()
Me.txtname = "": Me.txtsname = ""
Me.txtadd = "": Me.txtfone = ""
Me.txtlicense = "": Me.txtNIC = ""
Me.txtname.SetFocus
End Sub
```

Save Button

```
Private Sub cmdSave_Click()
Set db = CurrentDb()
```

```
Set rs = db.OpenRecordset("members")
rs.AddNew
rs.Fields("name").Value = Me.txtname
rs.Fields("sname").Value = Me.txtsname
rs.Fields("address").Value = Me.txtadd
rs.Fields("fone").Value = Me.txtfone
rs.Fields("licenseNo").Value = Me.txtlicense
rs.Fields("NICNo").Value = Me.txtNIC
rs.Update
MsgBox ("New record of a member has been added to the database"), vbInformation,
Me.Caption
rs.Close: Set rs = Nothing
db.Close: Set db = Nothing
End Sub
```

EDIT MEMBER FORM

Option Compare Database

```
Public db As DAO.Database
Public rs As DAO.Recordset
Public s As String
```

To Go Back To Main Menu

```
Private Sub cmdBack_Click()  
DoCmd.Close acForm, "NewMember"  
End Sub
```

Form Fields Initializer

```
Private Sub cmdNew_Click()  
Me.txtname = "": Me.txtsname = ""  
Me.txtadd = "": Me.txtfone = ""  
Me.txtlicense = "": Me.txtNIC = ""  
Me.txtname.SetFocus  
End Sub
```

To Open Search Member Options

```
Private Sub cmdFind2_Click()  
DoCmd.OpenForm "submemsearch"  
End Sub
```

Query Member Search Result Form

```
Private Sub cmdSearch_Click()  
DoCmd.OpenForm "memsearch"  
DoCmd.Close acForm, "submemsearch"  
End Sub
```

Save Changes Button

```
Private Sub cmdSave_Click()  
Set db = CurrentDb()
```

```
Set rs = db.OpenRecordset("members")
```

```
rs.AddNew
```

```
rs.Fields("name").Value = Me.txtname
```

```
rs.Fields("sname").Value = Me.txtsname
```

```
rs.Fields("address").Value = Me.txtadd
```

```
rs.Fields("fone").Value = Me.txtfone
```

```
rs.Fields("licenseNo").Value = Me.txtlicense
```

```
rs.Fields("NICNo").Value = Me.txtNIC
```

```
rs.Update
```

```
MsgBox ("New record of a member has been added to the database"), vbInformation,
```

```
Me.Caption
```

```
rs.Close: Set rs = Nothing
```

```
db.Close: Set db = Nothing
```

```
End Sub
```

RSERVATION SUB MENU

New Reservation Button

```
Private Sub Command0_Click()  
DoCmd.OpenForm "NewReservation"  
End Sub
```

Edit Reservation Button

```
Private Sub Command1_Click()  
DoCmd.OpenForm "EditReservation"  
End Sub
```

NEW RESERVATION FORM

Option Compare Database

```
Public db As DAO.Database  
Public rs As DAO.Recordset  
Public s As String
```

To Go Back to Main Menu

```
Private Sub cmdBack_Click()  
DoCmd.Close acForm, "NewReservation"  
End Sub
```

Form Fields Initialize Code

```
Private Sub cmdNew_Click()  
Me.txtID = "": Me.txtname = "": Me.txtsname = ""  
Me.txtlicense = "": Me.txtNIC = ""  
Me.txtcarreg = "": Me.txtcolor = "": Me.txtmade = ""  
Me.txtrent = "": Me.txtmodal = ""  
Me.txtduration = "": Me.txtfrom = ""  
Me.txtID.SetFocus  
End Sub
```

Print Reservation Slip Button

```
Private Sub cmdPrint_Click()  
DoCmd.OpenReport "resSlip", acViewPreview  
End Sub
```

Save Reservation Data Button

```
Private Sub cmdSave_Click()  
Set db = CurrentDb()  
If Me.txtfrom < Me.txtdate Then  
    MsgBox "You cant make a reservation for past dates"  
Else
```

```
s = "SELECT * FROM rentals WHERE (((#" & Format(Me.txtfrom, "mm\dd\yyyy") &
"# >= fromdate) AND (#" & Format(Me.txtfrom, "mm\dd\yyyy") & "# < exp_return)
AND (carreg=" & Me.txtcarreg & ") AND (canceled=False) AND (reserved=True)
AND (rented=False))) OR (((#" & Format(Me.txtexp_return, "mm\dd\yyyy") & "# >=
fromdate) AND (#" & Format(Me.txtexp_return, "mm\dd\yyyy") & "# < exp_return)
AND (carreg=" & Me.txtcarreg & ") AND (canceled=False) AND (reserved=True)
AND (rented=False)))"
```

```
Set rs = db.OpenRecordset(s)
```

```
If rs.EOF And rs.BOF Then
```

```
rs.AddNew
```

```
rs.Fields("carreg").Value = Me.txtcarreg
```

```
rs.Fields("memid").Value = Me.txtID
```

```
rs.Fields("date").Value = Me.txtdate
```

```
rs.Fields("duration").Value = Me.txtduration
```

```
rs.Fields("exp_return").Value = Me.txtexp_return
```

```
rs.Fields("fromdate").Value = Me.txtfrom
```

```
Me.txtresID = rs.Fields("serial").Value
```

```
rs.Fields("reserved").Value = True
```

```
rs.Update
```

```
MsgBox "Reservation is successful"
```

```
Else
```

```
MsgBox "Reservation exists for the mentioned dates"
```

```
End If
```

```
rs.Close: Set rs = Nothing
```

```
db.Close: Set db = Nothing
```

```
End If
```

```
End Sub
```

Car registration Update Event

```
Private Sub txtcarreg_AfterUpdate()
```

```
Set db = CurrentDb()
```

```
s = "SELECT * FROM cars WHERE carreg=" & Me.txtcarreg & ""
```

```
Set rs = db.OpenRecordset(s)
```

```
If rs.EOF And rs.BOF Then
```

```
MsgBox ("This car registration number does not exists in the database"), vbCritical,
```

```
Me.Caption
```

```
Me.txtID.SetFocus
```

```
Me.txtcarreg = ""
```

```
Me.txtcarreg.SetFocus
```

```
Else
```

```
Me.txtmade = rs.Fields("made").Value
```

```
Me.txtmodal = rs.Fields("modal").Value
```

```
Me.txtcolor = rs.Fields("color").Value
```

```
Me.txtrent = rs.Fields("rent").Value
```

```
End If
```

```
rs.Close: Set rs = Nothing
```

```
db.Close: Set db = Nothing
End Sub
```

Member ID Update Event

```
Private Sub txtID_AfterUpdate()
Set db = CurrentDb()
s = "SELECT * FROM members WHERE memid=" & Me.txtID & ""
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
MsgBox ("No record found"), vbCritical, Me.Caption
Me.txtcarreg.SetFocus
Me.txtID = ""
Me.txtID.SetFocus
Else
Me.txtname = rs.Fields("name").Value
Me.txtsname = rs.Fields("sname").Value
Me.txtlicense = rs.Fields("licenseNo").Value
Me.txtNIC = rs.Fields("NICNo").Value
End If
rs.Close: Set rs = Nothing
db.Close: Set db = Nothing
End Sub
```

EDIT RESERVATION FORM

Option Compare Database

```
Public db As DAO.Database
Public rs As DAO.Recordset, rs1 As DAO.Recordset
Public s As String, s1 As String
```

Back to Main Menu Button

```
Private Sub cmdBack_Click()
DoCmd.Close acForm, "EditReservation"
End Sub
```

Reservation Cancel Button

```
Private Sub cmdcancel_Click()
Set db = CurrentDb()
s = "SELECT * FROM rentals WHERE serial=" & Me.txtserial & " AND
reserved=True"
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
MsgBox "No such active reservation found in the database", vbCritical, Me.Caption
Else
If MsgBox("Are you sure you want to cancel the selected reservation", vbYesNo) =
vbYes Then
```

```

rs.Edit
rs.Fields("canceled").Value = True
rs.Fields("reserved").Value = False
rs.Update
MsgBox "Reservation has been canceled", , Me.Caption
End If
End If
End Sub

```

Form Fields Initialize Button

```

Private Sub cmdNew_Click()
Me.txtID = "": Me.txtname = "": Me.txtsname = ""
Me.txtlicense = "": Me.txtNIC = ""

Me.txtcarreg = "": Me.txtcolor = "": Me.txtmade = ""
Me.txtrent = "": Me.txtmodal = ""

Me.txtduration = "": Me.txtfrom = "": Me.txtserial = ""
Me.txtserial.SetFocus
End Sub

```

Reservation Update to Rent Button

```

Private Sub cmdUpdate_Click()
Set db = CurrentDb()
s = "SELECT * FROM rentals WHERE serial=" & Me.txtserial & " AND
reserved=True"
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
MsgBox "No such reservation found in the database", vbCritical, Me.Caption
Else
If MsgBox("Are you sure you want to rent", vbYesNo) = vbYes Then
rs.Edit
rs.Fields("rented").Value = True
rs.Update
Set rs = db.OpenRecordset("accounts")
rs.AddNew
rs.Fields("serial").Value = Me.txtresID
rs.Fields("memid").Value = Me.txtID
rs.Fields("carreg").Value = Me.txtcarreg
rs.Fields("debit").Value = Me.txtrent * Me.txtduration
rs.Fields("acc_date").Value = Me.txtdate
rs.Update
MsgBox "Car is rented to the customer", , Me.Caption
End If
End If
End Sub

```

```

Private Sub txtcarreg_AfterUpdate()
Set db = CurrentDb()
s = "SELECT * FROM cars WHERE carreg=" & Me.txtcarreg & ""
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
MsgBox ("This car registration number does not exists in the database"), vbCritical,
Me.Caption
Me.txtID.SetFocus
Me.txtcarreg = ""
Me.txtcarreg.SetFocus
Else
Me.txtmade = rs.Fields("made").Value
Me.txtmodal = rs.Fields("modal").Value
Me.txtcolor = rs.Fields("color").Value
Me.txtrent = rs.Fields("rent").Value
End If
rs.Close: Set rs = Nothing
db.Close: Set db = Nothing
End Sub

```

Member ID After Update Event

```

Private Sub txtID_AfterUpdate()
Set db = CurrentDb()
s = "SELECT * FROM members WHERE memid=" & Me.txtID & ""
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
MsgBox ("No record found"), vbCritical, Me.Caption
Me.txtcarreg.SetFocus
Me.txtID = ""
Me.txtID.SetFocus
Else
Me.txtname = rs.Fields("name").Value
Me.txtsname = rs.Fields("sname").Value
Me.txtlicense = rs.Fields("licenseNo").Value
Me.txtNIC = rs.Fields("NICNo").Value
End If
rs.Close: Set rs = Nothing
db.Close: Set db = Nothing
End Sub

```

Reservation ID After Update Event

```

Private Sub txtserial_AfterUpdate()
Set db = CurrentDb()
s = "SELECT * FROM rentals WHERE serial=" & Me.txtserial & ""
Set rs = db.OpenRecordset(s)

```

```

If rs.EOF And rs.BOF Then
    MsgBox "No such reservation number found in the database"
Else
    Me.txtcarreg = rs.Fields("carreg").Value
    Me.txtID = rs.Fields("memid").Value
    Me.txtfrom = rs.Fields("fromdate").Value
    Me.txtduration = rs.Fields("duration").Value
    s1 = "SELECT * FROM members WHERE memid=" & Me.txtID & ""
    Set rs1 = db.OpenRecordset(s1)
    If rs1.EOF And rs1.BOF Then
        MsgBox "No member found"
    Else
        Me.txtname = rs1.Fields("name").Value
        Me.txtsname = rs1.Fields("sname").Value
        Me.txtlicense = rs1.Fields("licenseNo").Value
        Me.txtNIC = rs1.Fields("NICNo").Value
    End If
    s1 = "SELECT * FROM cars WHERE carreg=" & Me.txtcarreg & ""
    Set rs1 = db.OpenRecordset(s1)
    If rs1.EOF And rs1.BOF Then
        MsgBox "Nothing found"
    Else
        Me.txtcolor = rs1.Fields("color").Value
        Me.txtmade = rs1.Fields("made").Value
        Me.txtmodal = rs1.Fields("modal").Value
        Me.txtrent = rs1.Fields("rent").Value
    End If
End If
End Sub

```

RENT SUB MENU

Return Car Form Button

```

Private Sub cmdReturn_Click()
    DoCmd.OpenForm "ReturnCar"
End Sub

```

New Rent Form Button

```

Private Sub Command0_Click()
    DoCmd.OpenForm "NewRent"
End Sub

```

Edit Rent Form Button

```

Private Sub Command1_Click()
    DoCmd.OpenForm "EditRent"
End Sub

```

NEW RENT FORM

```
Option Compare Database
Public db As DAO.Database
Public rs As DAO.Recordset
Public s As String
```

Back Button

```
Private Sub cmdBack_Click()
DoCmd.Close acForm, "NewRent"
End Sub
```

Form Fields Initializer

```
Private Sub cmdNew_Click()
Me.txtID = "": Me.txtname = "": Me.txtsname = ""
Me.txtlicense = "": Me.txtNIC = ""
Me.txtcarreg = "": Me.txtcolor = "": Me.txtmade = ""
Me.txtrent = "": Me.txtmodal = ""
Me.txtduration = "": Me.txtfrom = ""
Me.txtID.SetFocus
End Sub
```

Print Rent Slip Button

```
Private Sub cmdPrint_Click()
DoCmd.OpenReport "rentSlip", acViewPreview
End Sub
```

Save Rent Data Button

```
Private Sub cmdSave_Click()
Set db = CurrentDb()
If Me.txtfrom < Me.txtdate Then
MsgBox "You cant rent a car in past dates"
Else
s = "SELECT * FROM cars WHERE carreg=" & Me.txtcarreg & " AND
rented=True"
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
s = "SELECT * FROM rentals WHERE (((#" & Format(Me.txtfrom,
"mm\dd\yyyy") & "# >= fromdate) AND (#" & Format(Me.txtfrom, "mm\dd\yyyy") &
"# < exp_return) AND (carreg=" & Me.txtcarreg & ") AND (canceled=False) AND
(reserved=True) AND (rented=false))) OR (((#" & Format(Me.txtexp_return,
"mm\dd\yyyy") & "# >= fromdate) AND (#" & Format(Me.txtexp_return,
"mm\dd\yyyy") & "# < exp_return) AND (carreg=" & Me.txtcarreg & ") AND
(canceled=False) AND (reserved=True) AND (rented=false))) OR (((#" &
Format(Me.txtexp_return, "mm\dd\yyyy") & "# >= fromdate) AND (#" &
Format(Me.txtexp_return, "mm\dd\yyyy") & "# < exp_return) AND (carreg=" &
```

```

Me.txtcarreg & "" AND (canceled=False) AND (reserved=True) AND (rented=True)))
OR (((#" & Format(Me.txtfrom, "mm\dd\yyyy") & "# >= fromdate) AND (#" &
Format(Me.txtfrom, "mm\dd\yyyy") & "# < exp_return) AND (carreg="" &
Me.txtcarreg & "" AND (canceled=False) AND (reserved=True) AND
(rented=TRUE)))"

```

```

Set rs = db.OpenRecordset(s)

```

```

If rs.EOF And rs.BOF Then

```

```

    rs.AddNew

```

```

    rs.Fields("carreg").Value = Me.txtcarreg

```

```

    rs.Fields("memid").Value = Me.txtID

```

```

    rs.Fields("date").Value = Me.txtdate

```

```

    rs.Fields("duration").Value = Me.txtduration

```

```

    rs.Fields("exp_return").Value = Me.txtexp_return

```

```

    rs.Fields("fromdate").Value = Me.txtfrom

```

```

    Me.txtresID = rs.Fields("serial").Value

```

```

    rs.Fields("reserved").Value = True

```

```

    rs.Fields("rented").Value = True

```

```

    rs.Update

```

```

    s = "SELECT * FROM cars WHERE carreg="" & Me.txtcarreg & """

```

```

    Set rs = db.OpenRecordset(s)

```

```

    If rs.EOF And rs.BOF Then

```

```

        Else

```

```

            rs.Edit

```

```

            rs.Fields("rented").Value = True

```

```

            rs.Update

```

```

        End If

```

```

        Set rs = db.OpenRecordset("accounts")

```

```

        rs.AddNew

```

```

        rs.Fields("serial").Value = Me.txtresID

```

```

        rs.Fields("memid").Value = Me.txtID

```

```

        rs.Fields("carreg").Value = Me.txtcarreg

```

```

        rs.Fields("debit").Value = Me.txtrent * Me.txtduration

```

```

        rs.Fields("acc_date").Value = Me.txtdate

```

```

        rs.Update

```

```

        MsgBox "Car rented successfull"

```

```

    Else

```

```

        MsgBox "Car can not be rented out in these dates"

```

```

    End If

```

```

Else

```

```

    MsgBox "Car already rented out"

```

```

End If

```

```

rs.Close: Set rs = Nothing

```

```

db.Close: Set db = Nothing

```

```

End If

```

```

End Sub

```

Car reg. After Update Event

```
Private Sub txtcarreg_AfterUpdate()  
Set db = CurrentDb()  
s = "SELECT * FROM cars WHERE carreg=" & Me.txtcarreg & ""  
Set rs = db.OpenRecordset(s)  
If rs.EOF And rs.BOF Then  
    MsgBox ("This car registration number does not exists in the database"), vbCritical,  
Me.Caption  
    Me.txtID.SetFocus  
    Me.txtcarreg = ""  
    Me.txtcarreg.SetFocus  
Else  
    Me.txtmade = rs.Fields("made").Value  
    Me.txtmodal = rs.Fields("modal").Value  
    Me.txtcolor = rs.Fields("color").Value  
    Me.txtrent = rs.Fields("rent").Value  
End If  
rs.Close: Set rs = Nothing  
db.Close: Set db = Nothing  
End Sub
```

Member ID After Update Event

```
Private Sub txtID_AfterUpdate()  
Set db = CurrentDb()  
s = "SELECT * FROM members WHERE memid=" & Me.txtID & ""  
Set rs = db.OpenRecordset(s)  
If rs.EOF And rs.BOF Then  
    MsgBox ("No record found"), vbCritical, Me.Caption  
    Me.txtcarreg.SetFocus  
    Me.txtID = ""  
    Me.txtID.SetFocus  
Else  
    Me.txtname = rs.Fields("name").Value  
    Me.txtsname = rs.Fields("sname").Value  
    Me.txtlicense = rs.Fields("licenseNo").Value  
    Me.txtNIC = rs.Fields("NICNo").Value  
End If  
rs.Close: Set rs = Nothing  
db.Close: Set db = Nothing  
End Sub
```

EDIT RENT FORM

Option Compare Database

```
Public db As DAO.Database
Public rs As DAO.Recordset, rs1 As DAO.Recordset
Public s As String, s1 As String
```

Back Button

```
Private Sub cmdBack_Click()
DoCmd.Close acForm, "EditRent"
End Sub
```

Cancel Button

```
Private Sub cmdcancel_Click()
Set db = CurrentDb()
If Me.txtfrom >= Me.txtdate Then
    MsgBox "Sorry you can not cancel this rental", , Me.Caption
Else
s = "SELECT * FROM rentals WHERE serial=" & Me.txtserial & " AND rented=True"
Set rs = db.OpenRecordset(s)
If rs.EOF And rs.BOF Then
    MsgBox "No such active rental found in the database", vbCritical, Me.Caption
Else
    If MsgBox("Are you sure you want to do the selected operation", vbYesNo) = vbYes
Then
        rs.Edit
        rs.Fields("canceled").Value = True
        rs.Fields("reserved").Value = False
        rs.Fields("rented").Value = False
        rs.Update
        MsgBox "Rent has been canceled", , Me.Caption
    End If
End If
End Sub
```

Private Sub cmdNew_Click()

```
Me.txtID = "": Me.txtname = "": Me.txtsname = ""
Me.txtlicense = "": Me.txtNIC = ""
```

```
Me.txtcarreg = "": Me.txtcolor = "": Me.txtmade = ""
Me.txtrent = "": Me.txtmodal = ""
```

```
Me.txtduration = "": Me.txtfrom = "": Me.txtserial = ""
Me.txtserial.SetFocus
End Sub
```

Rent ID After Update Event

```
Private Sub txtserial_AfterUpdate()  
Set db = CurrentDb()  
s = "SELECT * FROM rentals WHERE serial=" & Me.txtserial & " AND rented=True"  
Set rs = db.OpenRecordset(s)  
If rs.EOF And rs.BOF Then  
    MsgBox "No such reservation number found in the database"  
Else  
    Me.txtcarreg = rs.Fields("carreg").Value  
    Me.txtID = rs.Fields("memid").Value  
    Me.txtfrom = rs.Fields("fromdate").Value  
    Me.txtduration = rs.Fields("duration").Value  
    s1 = "SELECT * FROM members WHERE memid=" & Me.txtID & ""  
    Set rs1 = db.OpenRecordset(s1)  
    If rs1.EOF And rs1.BOF Then  
        MsgBox "No member found"  
    Else  
        Me.txtname = rs1.Fields("name").Value  
        Me.txtsname = rs1.Fields("sname").Value  
        Me.txtlicense = rs1.Fields("licenseNo").Value  
        Me.txtNIC = rs1.Fields("NICNo").Value  
    End If  
    s1 = "SELECT * FROM cars WHERE carreg=" & Me.txtcarreg & ""  
    Set rs1 = db.OpenRecordset(s1)  
    If rs1.EOF And rs1.BOF Then  
        MsgBox "Nothing found"  
    Else  
        Me.txtcolor = rs1.Fields("color").Value  
        Me.txtmade = rs1.Fields("made").Value  
        Me.txtmodal = rs1.Fields("modal").Value  
        Me.txtrent = rs1.Fields("rent").Value  
    End If  
End If  
End Sub
```

REPORTS FORM

Back Button

```
Private Sub cmdBack_Click()  
DoCmd.Close acForm, "report"  
End Sub
```

Available Cars Report

```
Private Sub cmdavailable_Click()  
Me.Label4.Visible = True  
Me.Label3.Visible = True
```

```
Me.Label5.Visible = True
Me.listfrom.Visible = True
Me.listto.Visible = True
Me.subform1.Visible = True
Me.Label2.Visible = True
If MsgBox("Do you want to print the list", vbYesNo) = vbYes Then
    DoCmd.OpenReport "availablerpt", acViewPreview
End If
End Sub
```

Cars List Report

```
Private Sub cmdPrintord_Click()
DoCmd.OpenReport "carslist", acViewPreview
Me.subform1.Visible = False
Me.Label2.Visible = False
End Sub
```

Members Report Button

```
Private Sub cmdPrintrec_Click()
DoCmd.OpenReport "membersrpt", acViewPreview
Me.subform1.Visible = False
Me.Label2.Visible = False
End Sub
```

Reserved Cars Report

```
Private Sub cmdPrintsup_Click()
DoCmd.OpenReport "reservedrpt", acViewPreview
Me.subform1.Visible = False
Me.Label2.Visible = False
End Sub
```

Rented Car Report Button

```
Private Sub cmdVAcc_Click()
DoCmd.OpenReport "rentedrpt", acViewPreview
Me.subform1.Visible = False
Me.Label2.Visible = False
End Sub
```