

Inventory System for Used Car Parts Store

By

Ahmad Salama & Tony Brandal

**Submitted to the Faculty of the Information Engineering Technology Program
in the Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Engineering Technology**

**University of Cincinnati
College of Applied Science**

March 2001

Inventory System for Used Car Parts Store




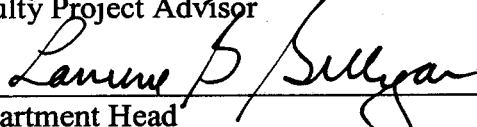
By

Ahmad Salama & Tony Brandal

Submitted to the Faculty of the Information Engineering Technology Program
in the Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Engineering Technology

@Copyright 2001 Ahmad Salama & Tony Brandal

The author grant to the Information Engineering Technology Program permission to reproduce and distribute copies of this document in whole or in part.

| | |
|-------------------------------------------------------------------------------------------------------------------------|-------------------------|
|  _____ Ahmad Salama | <u>3/1/01</u> Date |
|  _____ Tony Brandal | <u>03.01.01</u> Date |
|  _____ Faculty Project Advisor | <u>3-5-01</u> Date |
|  _____ Department Head | <u>3/5/01</u> Date |

Acknowledgements & Dedication

We would like to thank our advisor Professor Schlemmer, Professor McMahon for advising and helping us on our Senior Design Project as well as rest of the IET Faculty. We would like to thank God for making this project a reality. Finally, we would like to thank our parents for supporting us in college.

Table Of Contents

| | |
|------------------------------------------------------|--------------|
| Acknowledgements | i |
| Table of Contents | ii |
| List of Illustrations | iii |
| Abstract | iv |
| 1. Statement of Problem | 1-1 |
| 2. Review of Literature | 1-2 |
| 3. Description Of The Solution | 2-3 |
| 3.1 User Profile. | 3-3 |
| 4. Objectives Of The Project | 4-4 |
| 5. Design And Development | 4-6 |
| 5.1 Budget | 5-5 |
| 6. Proof of Design | 6-24 |
| 6.1 Databases | 7-10 |
| 6.2 Visual Basic | 10-24 |
| 7. Conclusion | 24-24 |
| 8. Appendix: Documentation Parts Manager 2002 | 25-36 |

List of Figures

| | |
|-----------------------------------|----|
| Figure 1. Database Model. | 7 |
| Figure 2. Waiting List Table. | 7 |
| Figure 3. Customers Table. | 8 |
| Figure 4. Orders Table. | 8 |
| Figure 5. PartNames. | 9 |
| Figure 6. Parts Table. | 9 |
| Figure 7. Cars Table. | 9 |
| Figure 8. Models Table. | 10 |
| Figure 9. Sales Person Screen | 11 |
| Figure 10. Make a sale. | 12 |
| Figure 11. Search for Parts. | 13 |
| Figure 12. Shopping Cart. | 14 |
| Figure 13. Waiting List. | 15 |
| Figure 14. View Waiting List. | 16 |
| Figure 15. Manager's Main Screen. | 17 |
| Figure 16. Add/Delete Users. | 18 |
| Figure 17. Register New Parts. | 19 |
| Figure 18. Add/delete New Items. | 20 |
| Figure 19. Add New Make. | 20 |
| Figure 20. Add New Model. | 21 |
| Figure 21. Add New Category. | 21 |
| Figure 22. Add New Part. | 22 |

Figure 23. Printing reports

23

Figure 24. Printing reports

23

Abstract

This project focuses on an inventory system for used car parts. The software application is called Parts Manager 2000. This is a product that features over 60,000 used car parts from some of the most popular imports such as Honda, Toyota, Nissan, Mazda, and many others. The product features car parts of close to 30 different makes and over 300 models from 1970 to present. The main functions of Parts Manager 2000 include: a sales function where the salesperson can use the system to search for parts for a customer, and then make a sale. An invoice will automatically be printed. Updating the inventory when new parts arrive, or old and damaged parts are removed. Creating new makes and models in the database. SQL Server authentication login required to use the system. Adding and deleting users from system; users will either have a salesperson account or a manager account, which can perform more advanced tasks. Finally, if a part is not available the customer will be put on a waiting list and then contacted when the part becomes available.

Inventory System for Used Car Parts Store

1. Statement Of The Problem.

Many of the used car parts stores in Cincinnati are poorly organized. Many have big lots with over 1000 wrecked cars and hundreds of thousands of parts. Customers can come and look for usable parts on savaged cars, and pay for whatever usable parts they find.

The problem is that the employees do not always know what specific parts they have available, they just tell the customers to look for the parts they need. By doing it this way, many customers decide to go elsewhere since they do not want to waste their time looking for parts in a big unorganized lot. Other problems can be that they do not have an easy way of getting an overview and there is no inventory list of parts available. This company does not have a customer database so they can send newsletters or other advertisements to their customers, and they do not have an easy way of creating invoices for the customers.

2. Review Of Literature.

We talked with the owners of several used parts stores. One of them, Robert Klamer of Bob's Used Parts, was very interested in our project. He told us that on any given week he buys several wrecked or damaged cars for use in selling used parts. He mentioned that his major problem was that he has a difficult time remembering every make, model and year of a car, and the parts that were reusable on that particular car. We told him this software product would meet his specifications because it would systematically organize parts for each car based on the make, model, and year of the car. After our discussions he told us this: *"We want something that is easy to use and is*

customized to meet our needs. We want to invest in something that will make us more profitable in the future.”

3. Description Of The Solution.

The software application we propose is called Parts Manager 2000. This is a product that features over 60,000 used car parts from some of the most popular imports such as Honda, Toyota, Nissan, Mazda, and many others. The product features car parts of close to 30 different makes and over 300 models from 1970 to present. The main functions of Parts Manager 2000 are:

- A sales function where the salesperson can use the system to search for parts for a customer, and then make a sale. An invoice will automatically be printed.
- Update inventory when new parts arrives, or old or damaged parts are removed.
- Create new makes and models in the database as time goes by.
- SQL Server authentication login required to use the system.
- Add and delete users from system. Users will either have a salesperson account or a manager account, which can perform more advanced tasks.
- Users can change their passwords.
- Print various reports.
- Add customers to a waiting list if the part they are looking for is not in stock.

3.1 User Profile.

The users are assumed to have low Information Technology literacy, which means they will need training on how to use the product and how to use a computer. Parts Manager 2000 is designed to be simple to use. We have put effort into making each function as simple as possible, with clearly labeled buttons giving the user the available

options at all times. There will be two categories of users. One is the manager, which will have administrative privileges, such as manually adding and deleting parts in the database. He will also be able to add and delete users, printing sales figures and reports for the month, as well as other structural changes. The other category will be a salesperson category with privileges limited to sales related functions only.

3.1.1 Design Protocols.

The purpose of this inventory system for used car parts is to offer the user the following features:

- An easy to use system that organizes parts in an orderly fashion.
- A robust system and powerful system that will allow the user to search for parts, purchase the parts, and update the database.
- Design the system to put a customer on a waiting list if the part is not available, and then contact the customer to notify him when part becomes available.
- An administrative role that is designed to allow a manager to perform administrative tasks like adding/deleting users and adding/deleting parts.
- The system is designed to print a receipt or invoice for the customer whenever a sale is made.
- The project is designed to offer a documentation and user guide menu to help the user troubleshoot any problem that may occur.

4. Objectives Of The Project.

We will deliver a Visual Basic and Database product that will meet the criteria for Senior Design. It will include:

- A SQL Server 7.0 database containing over 60,000 different car parts
- A Visual Basic 6.0 front end application with the following functions:
 - User login using SQL Server authentication
 - Advanced searching capabilities
 - Sales function with automatic invoicing and updating of database inventory
 - Customer waiting list for parts
 - Reporting functions
 - Function to add and delete users from system
 - Function for users to change their passwords
 - Function to manually update inventory of database
 - Function to manually create new items in database
- An user manual covering all the functions of the application

5. Design And Development.

5.1 Budget

Most of the resources we need to use are in the IET lab. The software we will use will be Visual Basic 6.0, SQL Server 7.0, Microsoft Office 2000, and MS Visio 5.0. The retail budget for the project is shown on the next page:

Retail Expenses For Project:

| | |
|---------------------------------|--------------------|
| Computer | \$ 1,400.00 |
| Printer | \$ 390.00 |
| Windows 2000 Professional | \$ 250.00 |
| Visual Basic 6.0 Enterprise | \$ 167.00 |
| SQL Server 7.0 (5 User license) | \$ 1,250.00 |
| Microsoft Office 2000 | \$ 530.00 |
| Books for education | \$ 150.00 |
| Total costs | \$ 4,137.00 |

Timeline

Below is a Gantt chart of the target dates and major milestones in the project. We will follow these dates strictly so that we can be sure that we will meet all the deadlines set for the project.

| ID | Task Name | Start Date | End Date | Duration | 2000 | | | | | | | | | | | | 2001 | | |
|----|-------------------------|------------|----------|----------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|------|---|--|
| | | | | | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec | Jan | Feb | Mar | | | | |
| 1 | Proposal | 5/9/00 | 5/15/00 | 5d | ■ | | | | | | | | | | | | | | |
| 2 | Oral Presentation | 5/15/00 | 5/31/00 | 13d | ■ | | | | | | | | | | | | | | |
| 3 | Feasibility of Project | 5/31/00 | 10/6/00 | 93d | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | | | | |
| 4 | Prototype | 10/6/00 | 10/26/00 | 15d | | | | | | ■ | ■ | ■ | ■ | | | | | | |
| 5 | Oral Presentation | 10/26/00 | 11/15/00 | 15d | | | | | | | ■ | ■ | ■ | ■ | ■ | | | | |
| 6 | Final Project | 11/15/00 | 2/6/01 | 60d | | | | | | | | | | ■ | ■ | ■ | ■ | ■ | |
| 7 | Final Oral Presentation | 2/6/01 | 3/2/01 | 19d | | | | | | | | | | | | | | ■ | |

Software

The software needed for this project include Windows 2000 Professional, Visual Basic 6.0, SQL Server 7.0, and Microsoft Office 2000.

Hardware

The hardware for this project includes two computers. The main computer used to develop the front-end application is a Pentium II 333 Mhz, with 192MB of RAM, 16 MB video adapter, Sound Blaster AWE 32 sound card,

network adapter, monitor, keyboard, mouse and printer. The other computer, which was mainly used to develop the database and populate the records, is a Pentium III 500 Mhz, with 64MB of RAM, 16 MB Video and Sound Card built into a Socket 7 motherboard, monitor, keyboard, and mouse.

6. Proof Of Design.

6.1 Database

We designed a sophisticated database that follows the three rules of normalization. The first norm is the atomic rule, each fields much be broken down into the smallest part. The second norm is that each field must be uniquely identified with a primary key. The third norm is that there should be no duplication of data in the tables in the database, except in key fields.

Data Model

On the next page in Figure 1 is an overall view of the data model for our SQL server Database. Overall we have seven tables that the application extracts the data. In the next section, the data architecture will be discussed in more detail.

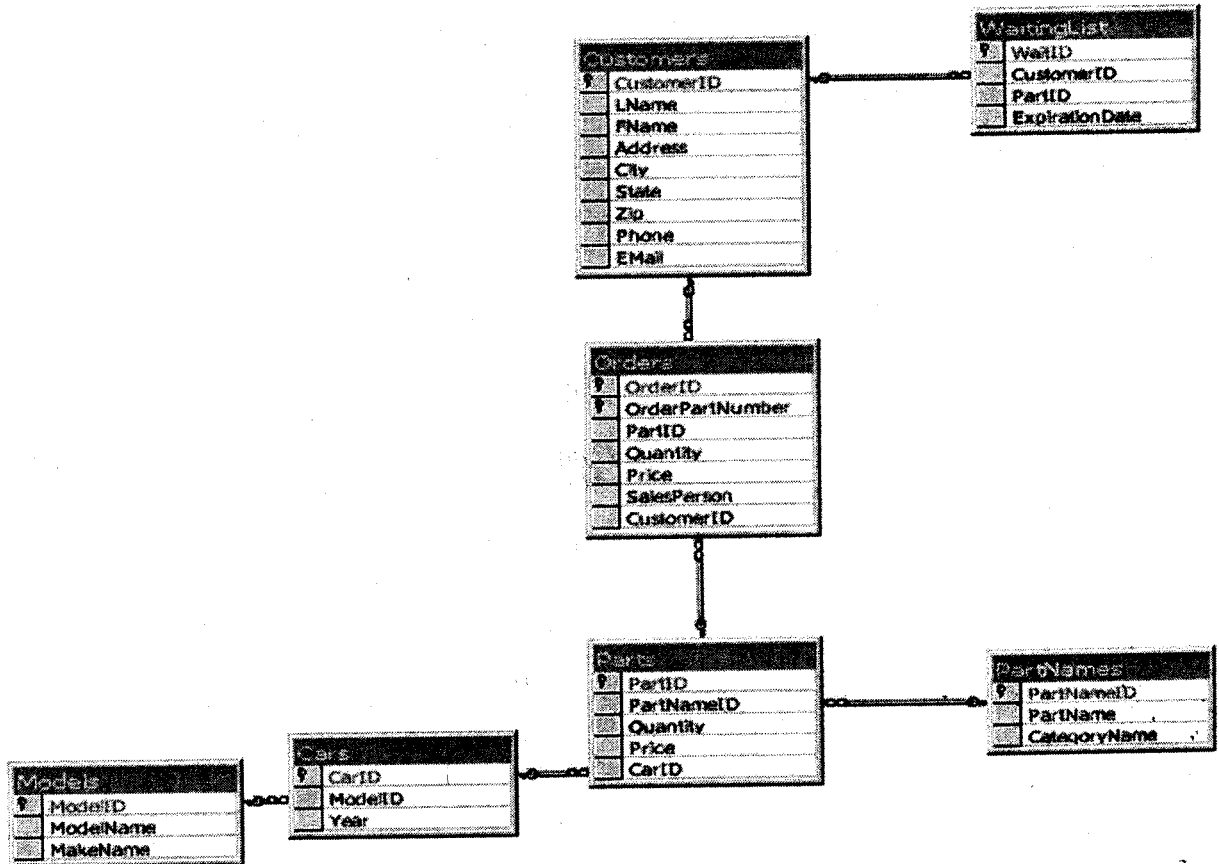


Figure 1. Database Model.

Data Architecture

The first table in the model is the waiting list table show in Figure 2 Below.

| | Column Name | Datatype | Length | Precision | Scale | Allow Nulls | Default Value | Identity | Identity Seed | Identity Increment | Is RowGuid |
|----|----------------|----------|--------|-----------|-------|-------------------------------------|---------------|-------------------------------------|---------------|--------------------|------------|
| PK | WaitID | int | 4 | 10 | 0 | | | <input checked="" type="checkbox"/> | 1 | 1 | |
| | CustomerID | int | 4 | 10 | 0 | <input checked="" type="checkbox"/> | | | | | |
| | PartID | int | 4 | 10 | 0 | <input checked="" type="checkbox"/> | | | | | |
| | ExpirationDate | datetime | 8 | 0 | 0 | <input checked="" type="checkbox"/> | | | | | |

Figure 2. Waiting List Table.

The purpose of the waiting list table is to put the customers on a waiting list if the part is out of stock or not available. The Wait ID field is the primary key that uniquely identifies the row in the table. Customer ID is a foreign key used to identify the customer on the waiting list. Part ID is another foreign key that identifies what part the customer is needs. The expiration date is the deadline the customer needs specific part.

Next is the customer's table shown in Figure 3 below.

| | Column Name | Datatype | Length | Precision | Scale | Allow Nulls | Default Value | Identity | Identity Seed | Identity Increment | Is RowGuid |
|----|-------------|----------|--------|-----------|-------|-------------------------------------|---------------|-------------------------------------|---------------|--------------------|------------|
| PK | CustomerID | int | 4 | 10 | 0 | | | <input checked="" type="checkbox"/> | 1000 | 1 | |
| | LName | varchar | 35 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | FName | varchar | 35 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | Address | varchar | 40 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | City | varchar | 25 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | State | varchar | 50 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | Zip | varchar | 10 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | Phone | varchar | 20 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | EMail | varchar | 30 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |

Figure 3. Customers Table.

The purpose of the customers table is to gather important information about the customer purchasing the auto parts. The Customer ID field is the primary key that uniquely identifies the row in the table. The LName is the last name of the customer, and FName is the first name of the customer. Then the Address, City, State, Zip, Phone and Email correspond to necessary information of the customer.

Next is the Waiting List table in Figure 4 shown below.

| | Column Name | Datatype | Length | Precision | Scale | Allow Nulls | Default Value | Identity | Identity Seed | Identity Increment | Is RowGuid |
|----|-----------------|----------|--------|-----------|-------|-------------------------------------|---------------|--------------------------|---------------|--------------------|------------|
| PK | OrderID | int | 4 | 10 | 0 | | | <input type="checkbox"/> | | | |
| PK | OrderPartNumber | int | 4 | 10 | 0 | | | <input type="checkbox"/> | | | |
| | PartID | int | 4 | 10 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | Quantity | varchar | 2 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | Price | money | 8 | 19 | 4 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | SalesPerson | varchar | 20 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |
| | CustomerID | int | 4 | 10 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | |

Figure 4. Orders Table.

The purpose of the orders table is to gather order information about the customers order. OrderID is a primary key that identifies the order customer place. OrderPartNumber is also a primary key that identifies the unique part within that order. Therefore OrderID and OrderPartNumber are together a primary key for this table. This will allow the customer to order more than one part for that specific order and can easily identify the part that is on hold if for a customer or a return, exchange, or refund for a specific part. Part ID is a foreign key from the uniquely identifies the make, model and year of a car.

Quantity identifies the number of parts the customer has in the order. Price corresponds to the price of the specific part. Salesperson refers to the name of the salesman that sold the part to the customer. CustomerID is a foreign key that identifies the customer.

Next is the PartNames table shown in Figure 5 below.

| Column Name | Data type | Length | Precision | Scale | Allow Nulls | Default Value | Identity | Identity Seed | Identity Increment | Is RowGuid |
|--------------|-----------|--------|-----------|-------|-------------------------------------|---------------|--------------------------|---------------|--------------------|--------------------------|
| PartNameID | int | 4 | 10 | 0 | <input type="checkbox"/> | (0) | <input type="checkbox"/> | | | <input type="checkbox"/> |
| PartName | nvarchar | 50 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | <input type="checkbox"/> |
| CategoryName | nvarchar | 50 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | <input type="checkbox"/> |

Figure 5. PartNames.

PartNameID is the primary key that uniquely identifies each part name in the car. PartName is the actual name of the part (i.e. muffler). CategoryName is the category the part is placed in (i.e. muffler is in the exhaust category).

The Parts table is the fifth table in the database as shown in Figure 6 below.

| Column Name | Data type | Length | Precision | Scale | Allow Nulls | Default Value | Identity | Identity Seed | Identity Increment | Is RowGuid |
|-------------|-----------|--------|-----------|-------|-------------------------------------|---------------|-------------------------------------|---------------|--------------------|--------------------------|
| PartID | int | 4 | 10 | 0 | <input type="checkbox"/> | | <input checked="" type="checkbox"/> | 1 | 1 | <input type="checkbox"/> |
| PartNameID | int | 4 | 10 | 0 | <input checked="" type="checkbox"/> | (0) | <input type="checkbox"/> | | | <input type="checkbox"/> |
| Quantity | int | 4 | 10 | 0 | <input checked="" type="checkbox"/> | (0) | <input type="checkbox"/> | | | <input type="checkbox"/> |
| Price | money | 8 | 19 | 4 | <input checked="" type="checkbox"/> | (0) | <input type="checkbox"/> | | | <input type="checkbox"/> |
| CarID | int | 4 | 10 | 0 | <input checked="" type="checkbox"/> | (0) | <input type="checkbox"/> | | | <input type="checkbox"/> |

Figure 6. Parts Table.

PartID is the primary key that uniquely identifies each part in the car. PartNameID is a foreign key uniquely identifies each part name in the car. Quantity refers to how many parts are available in the table for each specific part. Price is the price of the specific part for the car. CarID is the foreign key the make, model, and year of a car.

The Cars table is the sixth table in the car parts database as shown in Figure 7 on the next page.

| Column Name | Data type | Length | Precision | Scale | Allow Nulls | Default Value | Identity | Identity Seed | Identity Increment | Is RowGuid |
|-------------|-----------|--------|-----------|-------|-------------------------------------|---------------|-------------------------------------|---------------|--------------------|--------------------------|
| CarID | int | 4 | 10 | 0 | <input type="checkbox"/> | | <input checked="" type="checkbox"/> | 1 | 1 | <input type="checkbox"/> |
| ModelID | int | 4 | 10 | 0 | <input checked="" type="checkbox"/> | (0) | <input type="checkbox"/> | | | <input type="checkbox"/> |
| Year | nvarchar | 20 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | <input type="checkbox"/> |

Figure 7. Cars Table.

CarID is the primary key that uniquely identifies the make, model, and year id of a car. ModelID is the foreign key that uniquely identifies the model id of the car. Year corresponds to the year of the car.

The final table the database is the Models table as shown in Figure 8 below.

| Column Name | Data type | Length | Precision | Scale | Allow Nulls | Default Value | Identity | Identity Seed | Identity Increment | Is RowGUID |
|-------------|-----------|--------|-----------|-------|-------------------------------------|---------------|-------------------------------------|---------------|--------------------|--------------------------|
| CarID | int | 4 | 10 | 0 | <input type="checkbox"/> | | <input checked="" type="checkbox"/> | 1 | 1 | <input type="checkbox"/> |
| ModelName | nvarchar | 50 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | <input type="checkbox"/> |
| MakeName | nvarchar | 25 | 0 | 0 | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | | | <input type="checkbox"/> |

Figure 8. Models Table.

The ModelID is the primary key that identifies the model id of the car. ModelName corresponds to the model name of the car. MakeName corresponds to the make name or manufacturer of the car.

6.2 Visual Basic

When running Parts Manager 2000, a login screen will first prompt the user. After the user has typed in a correct user name and password, he will then enter the main screen. The main screen is designed differently for salespersons and for managers. Salespersons will enter directly into the Search For Parts-screen, while the managers will get a screen with several different options.

The screen for the salesman that includes limited functionality is shown below in Figure 9 on the top of the next page.

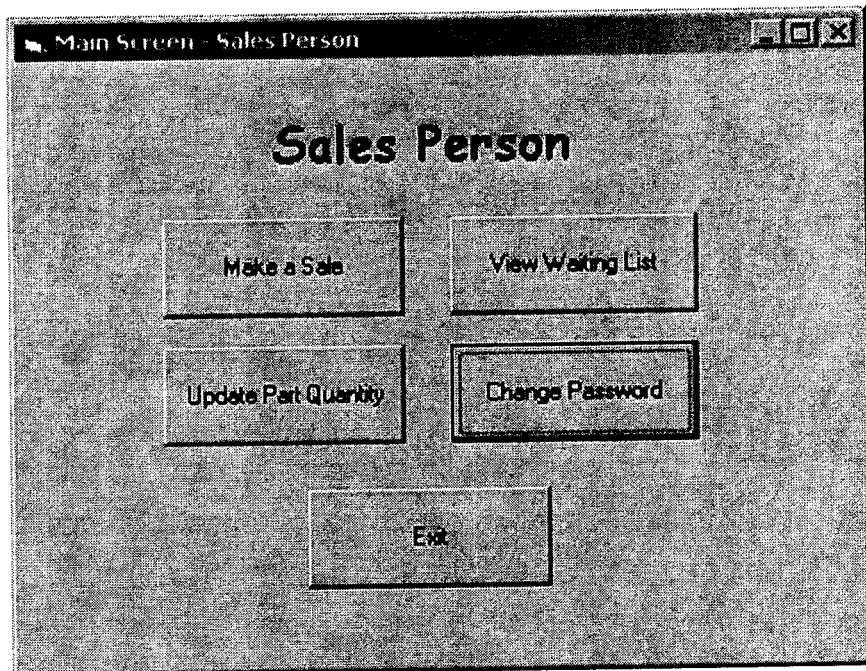


Figure 9. Sales Person Screen.

When you click on Make a sale, you are prompted to enter Customer registration information as shown below in Figure 10 on the top of the next page.

The image shows a web browser window titled "Customer Registration" with a form titled "Enter Customer Information". The form contains the following fields and values:

| | | | | | |
|-----------------|---------------|------------|-------------------|-----|-------|
| Customer Number | 1000 | | | | |
| Last Name | Smith | First Name | John | | |
| Address | 123 Abbey Rd. | | | | |
| City | Cincinnati | State | OH | Zip | 45220 |
| Telephone | [513]681-0987 | Email | smith@hotmail.com | | |

At the bottom of the form are three buttons: "Submit", "Clear Fields", and "Back to Main Screen".

Figure 10. Make a sale.

Once you enter in all the customer information you click on Submit. If the customer already exists in the database, the form will automatically populate itself after the first name and last name, or customer number, has been filled in. The salespersons' screen is shown in Figure 11 on the top of the next page.

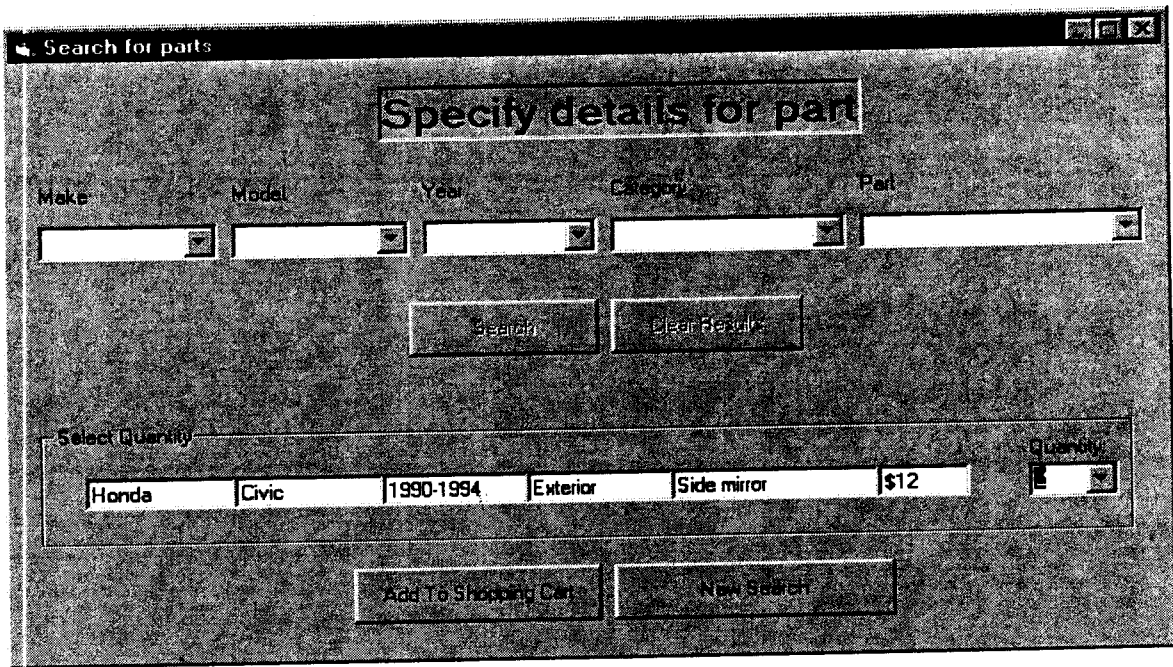


Figure 11. Search for Parts.

The salesperson will select a make, model and year of the part they are searching for, using the designated dropdown boxes on the top of the screen. When a make is selected, then the model-dropdown box is automatically populated with all existing models of the selected make. When a model is selected, the year-dropdown box is automatically populated with the current available years for the selected model. In order to make the selection of a specific part easier, the parts are grouped in categories. When a category is selected, the part-dropdown box is automatically populated with all the different parts in that category. When all the dropdown boxes are filled, the search-button is enabled. When the search button is selected, a query is created that searches through the parts database for that specific part. If the part is not found, the user will be prompted with a message box. If the part is found, the textboxes in the lower part of the screen are populated, as is the quantity-dropdown box, which is populated with numbers from 1 to the number of items in stock for that part. The salesperson can then select a quantity to be

sold from the quantity-dropdown box. The user will be able to do another search by clicking the New Search button, or adding the part to the Shopping Cart by clicking the Add to Shopping Cart button. Then the user will be shown a screen similar to Figure 12 shown below.

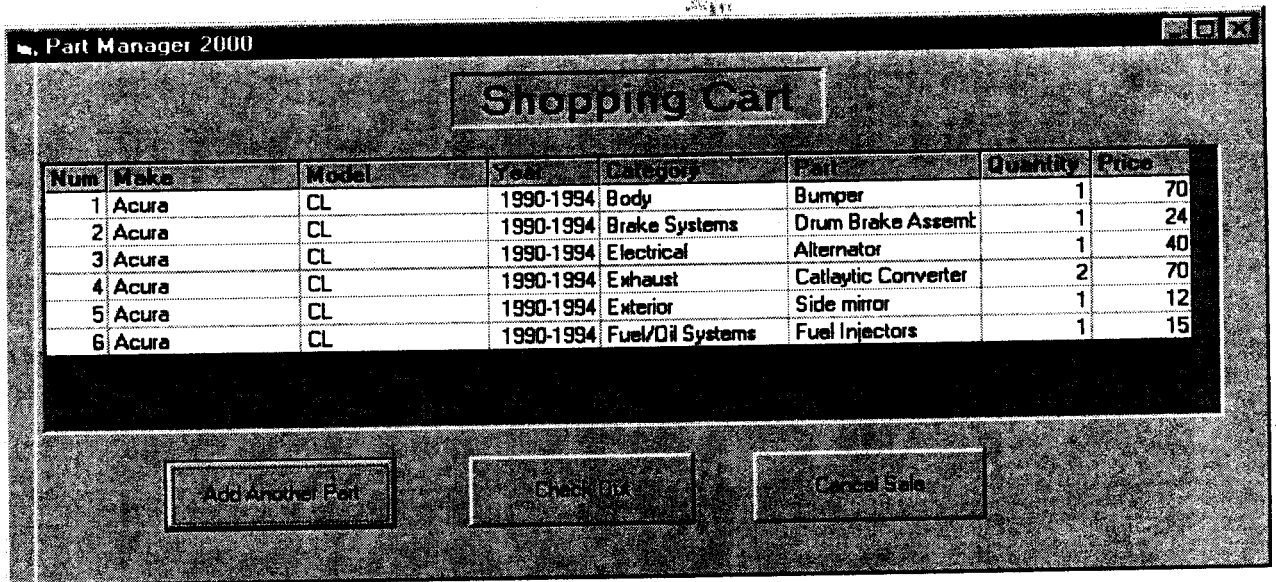


Figure 12. Shopping Cart.

The user is presented with three options: Add another part, Check out, or Cancel Sale. By clicking Add Another Part, he will be returned to the Search page, where he can search for another part to put into the shopping cart. When all the parts that the customer wants are placed in the shopping cart, the salesperson clicks on Check Out an invoice is automatically printed and the inventory is automatically updated. If the customer changes his mind and does not want to buy anything, the transaction can be cancelled by clicking Cancel Sale. Another feature of the sales module is the Waiting List for parts. If a customer comes in and is looking for a part that is currently not in stock, he can be put on a waiting list to be notified when the part is entered into the system again. The Waiting List option is shown on the top of the next page in Figure 13.

Waiting List

Customer Details

Customer ID: 1010

Last Name: Uskov First Name: Vladimir

Address: 2134 Programming St.

City: Chicago State: IL Zip: 56821

Email: Uskov@chicago.edu Phone: [401]555-1234

Waiting List Details

Make: Isuzu Model: Amigo

Year: 2000-2004 Partname: Bumper

Preferred way of notice:

Phone E-Mail US Mail

How many months on waiting list? 3

Buttons: Edit Customer Details, Add To Waiting List, Cancel

Figure 13. Waiting List.

This option is available if a car part is not available. The customer has the option to be put on this waiting list for up to 12 months. The manager will then be notified when the car part becomes available for that specific customer, and finally contact the customer by Phone, E-mail, or US Mail.

Finally, once the clicks the *Add to Waiting List* button the following screen in appears as shown in Figure 14 on the top of the next page.

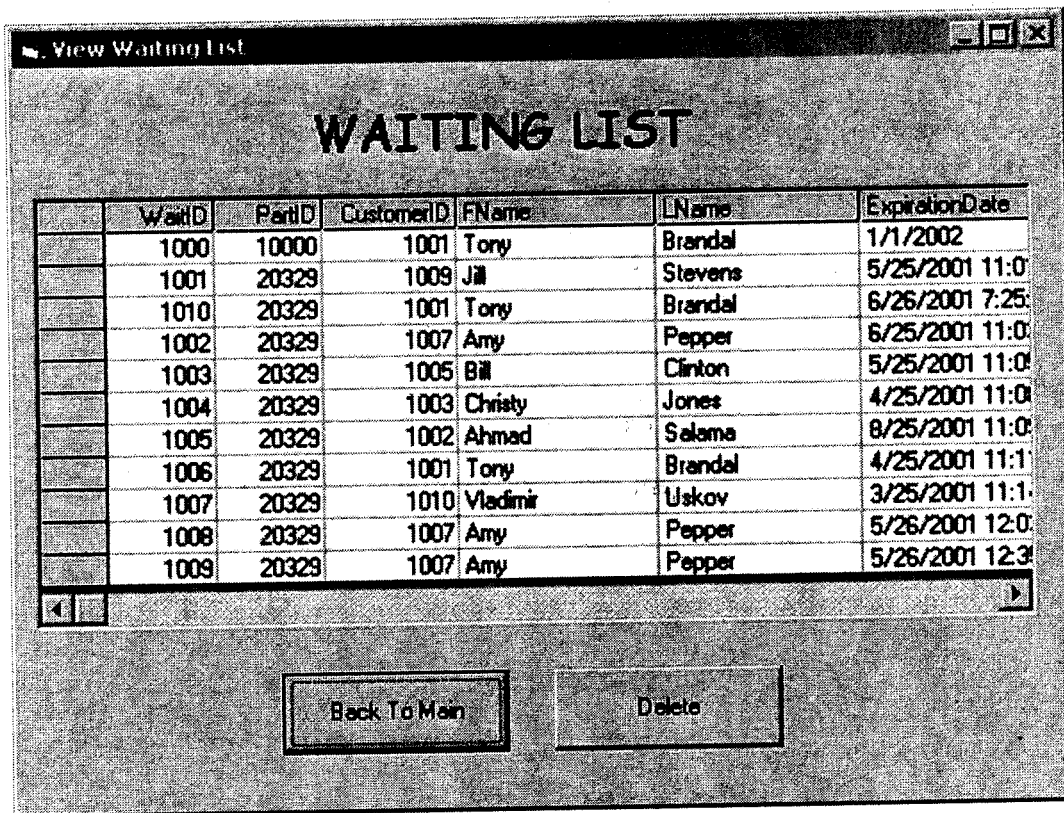


Figure 14. View Waiting List.

The waiting list features all the customers that currently are waiting for parts, it list the Wait ID, Part ID, Customer ID, First Name, Last Name and Expiration Date

The manager category's main page looks different from the salesperson's main page, and is shown in Figure 15 on the top of the next page.

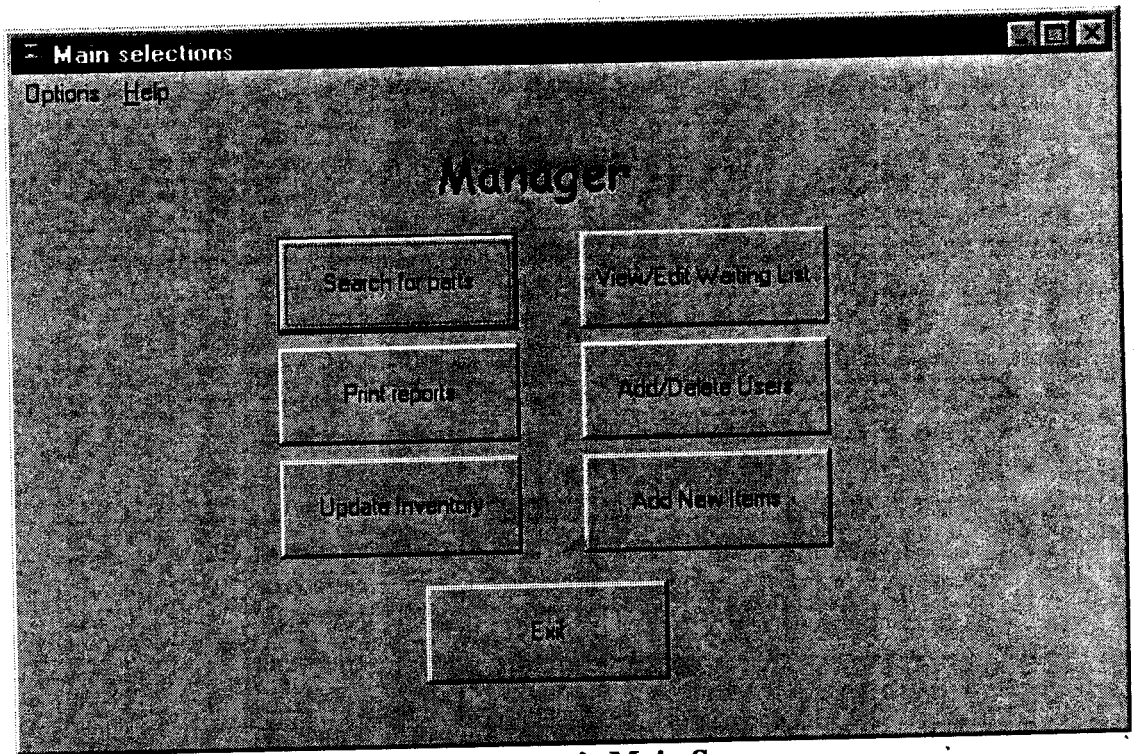


Figure 15. Manager's Main Screen.

The manager can also perform sales functions, as well as several other administrative functions. He can view the Waiting List and edit it (for example, deleting persons from it,) or manually add new entries. Another feature that has not yet been implemented is the reporting function. The manager can print several pre-defined reports about weekly and monthly sales, how much each of his salespersons have sold, or statistics on what makes or parts have been selling the most within a given time period. There will be an option to customize reports for more specific purposes. The manager can also add and delete users from the system, giving them either salesperson or manager privileges, using a screen similar to the one displayed in Figure 16 on the top of the next page.

Add / Delete Users

Add User Drop User

Sales Person Manager

Enter Username:

Enter Password:

Repeat Password:

Figure 16. Add/Delete Users.

Another function the manager can perform is to update the inventory of the database whenever new parts come in, or old or damaged parts are removed from the store, as shown in Figure 17 on the next page. Due to the large numbers of parts on each car, we are using tabs on the data entry sheet, so the user gets a better overview over the different parts.

Specify details for part to update

 Add Parts Remove Parts

Make Model Year
 Daewoo Lanos 1995-1999

| Parts 1 | Parts 2 |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Body <input checked="" type="checkbox"/> Bumper 1 <input type="checkbox"/> Headlight <input checked="" type="checkbox"/> Hood 1 <input type="checkbox"/> Left Fender <input checked="" type="checkbox"/> Right Fender 1 <input checked="" type="checkbox"/> Spoiler 1 <input type="checkbox"/> Trunk Brake System <input type="checkbox"/> Rotor <input checked="" type="checkbox"/> Emergency Brake 1 <input checked="" type="checkbox"/> Master Cylinder 1 <input type="checkbox"/> Drum Brake Assembly | Exterior <input type="checkbox"/> Doors <input type="checkbox"/> Lug Nuts <input checked="" type="checkbox"/> Rear Brake Lamp 1 <input type="checkbox"/> Rear Light <input checked="" type="checkbox"/> Side Marker Light 1 <input type="checkbox"/> Side mirror <input type="checkbox"/> Spare Tire <input checked="" type="checkbox"/> Tires 4 <input checked="" type="checkbox"/> Wheel Cap 1 <input checked="" type="checkbox"/> Windows 1 <input checked="" type="checkbox"/> Wiper Motor 1 |
| | HVAC <input type="checkbox"/> A/C Compressor <input checked="" type="checkbox"/> Heater Core 1 <input type="checkbox"/> Radiator Fuel/Oil Systems <input type="checkbox"/> Fuel Injector <input checked="" type="checkbox"/> Fuel Pump 1 <input type="checkbox"/> Fuel Tank <input checked="" type="checkbox"/> Oil Pan 1 <input type="checkbox"/> Oil Pump |
| <input type="button" value="Add Parts"/> <input type="button" value="Clear Fields"/> <input type="button" value="To Back To Main"/> | |

Figure 17. Register New Parts.

In addition to editing the inventory, the manager can also create new makes, models and parts as they are introduced on the market. The first option is Add/Deleting Items, that function is shown on figure 18 on the next page.

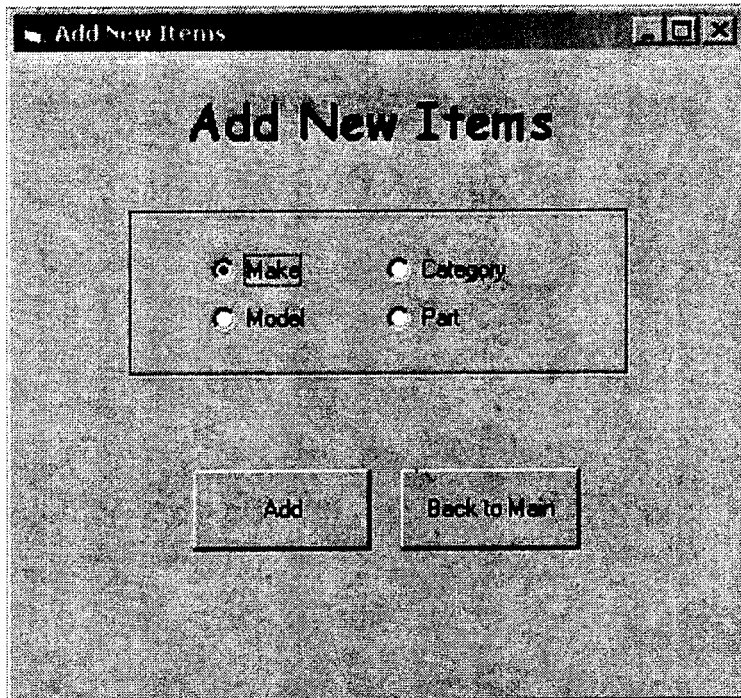


Figure 18. Add/delete New Items.

Another feature is adding specific makes to the database as shown below in Figure 19.

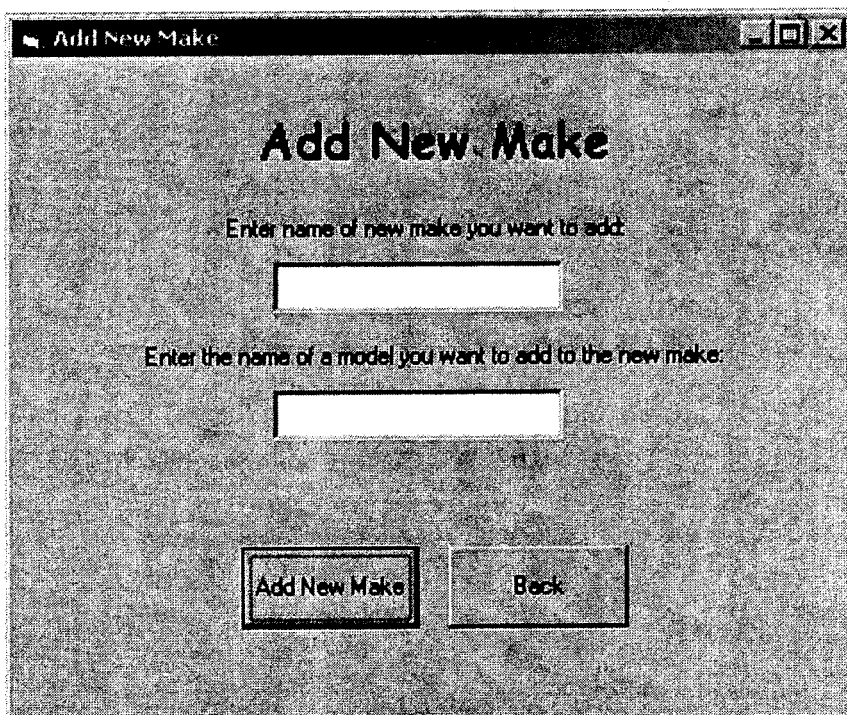
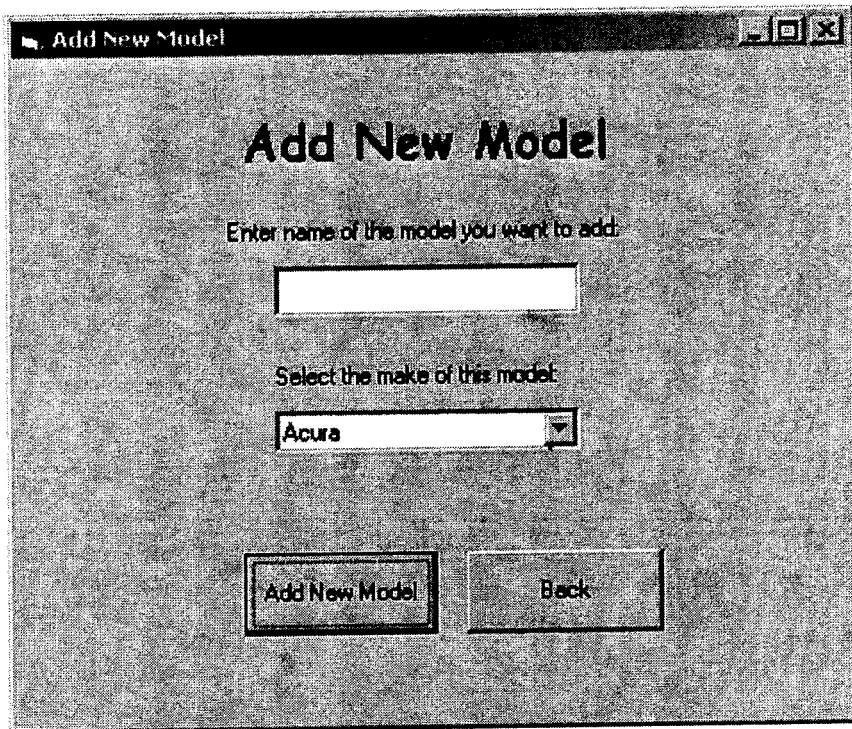


Figure 19. Add New Make.

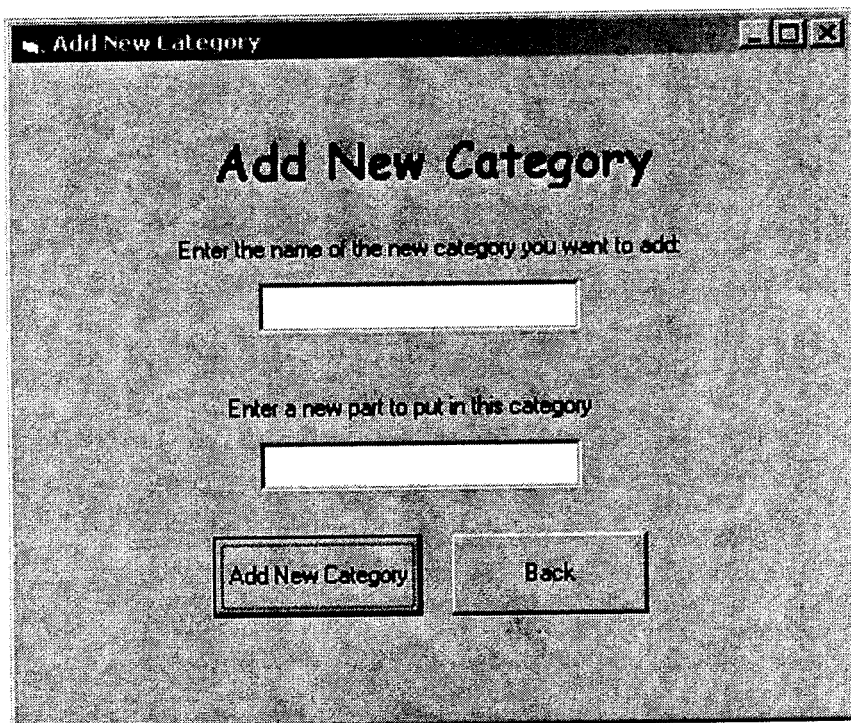
Another feature is adding specific Model to the database as shown below in Figure 20.



The screenshot shows a window titled "Add New Model". The main heading is "Add New Model". Below the heading, there is a text prompt "Enter name of the model you want to add:" followed by an empty text input field. Below that is another text prompt "Select the make of this model:" followed by a dropdown menu currently displaying "Acura". At the bottom of the window, there are two buttons: "Add New Model" and "Back".

Figure 20. Add New Model.

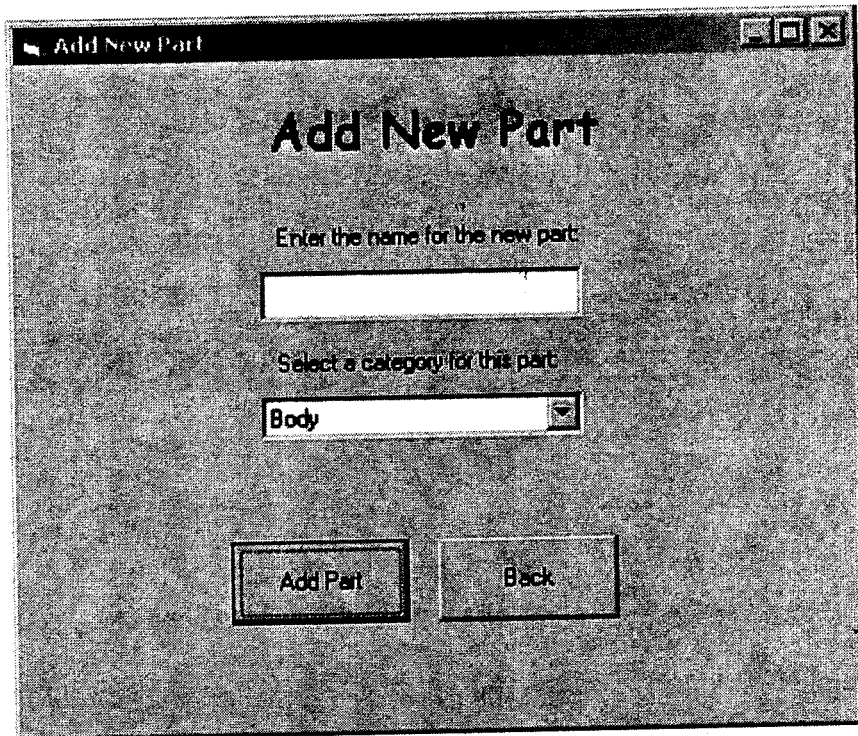
Another feature is adding specific Category to the database as shown below in Figure 21.



The screenshot shows a window titled "Add New Category". The main heading is "Add New Category". Below the heading, there is a text prompt "Enter the name of the new category you want to add:" followed by an empty text input field. Below that is another text prompt "Enter a new part to put in this category:" followed by an empty text input field. At the bottom of the window, there are two buttons: "Add New Category" and "Back".

Figure 21. Add New Category.

Another feature is adding specific new part to the database as shown below in Figure 22.



The image shows a graphical user interface window titled "Add New Part". The window contains the following elements:

- Title Bar:** "Add New Part" with standard window control icons (minimize, maximize, close).
- Heading:** "Add New Part" in a large, bold font.
- Text Prompt:** "Enter the name for the new part:"
- Input Field:** An empty rectangular text box for entering the part name.
- Text Prompt:** "Select a category for this part:"
- Dropdown Menu:** A menu box with "Body" selected and a small arrow icon on the right.
- Buttons:** Two buttons at the bottom: "Add Part" and "Back".

Figure 22. Add New Part.

Another feature is reporting of daily, weekly and monthly sales, either grouped by salesperson or displaying a total sale. The form for that function is shown in figure 23 below.

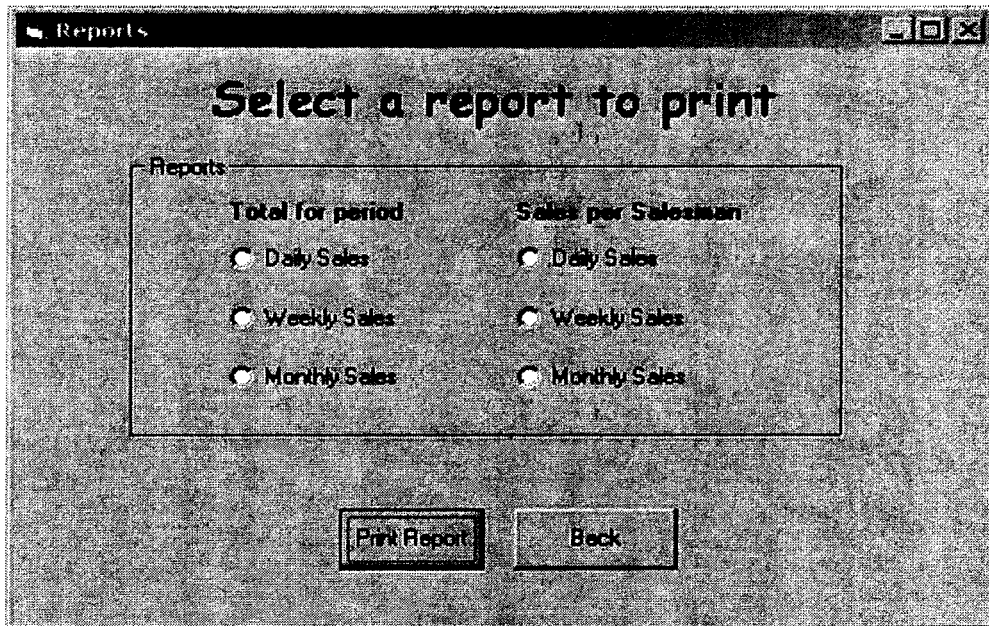


Figure 23. Printing reports

Finally below is a screen shot of sales as shown below in Figure 24.

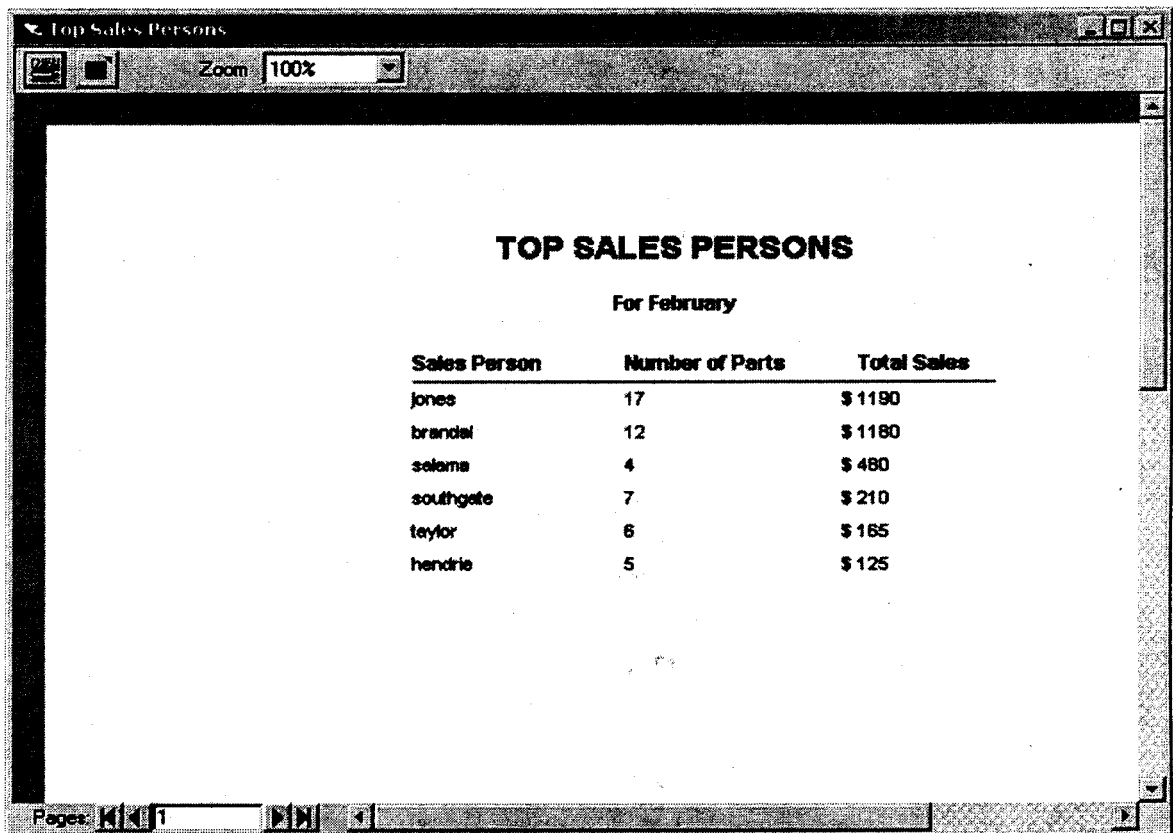


Figure 24. Printing reports

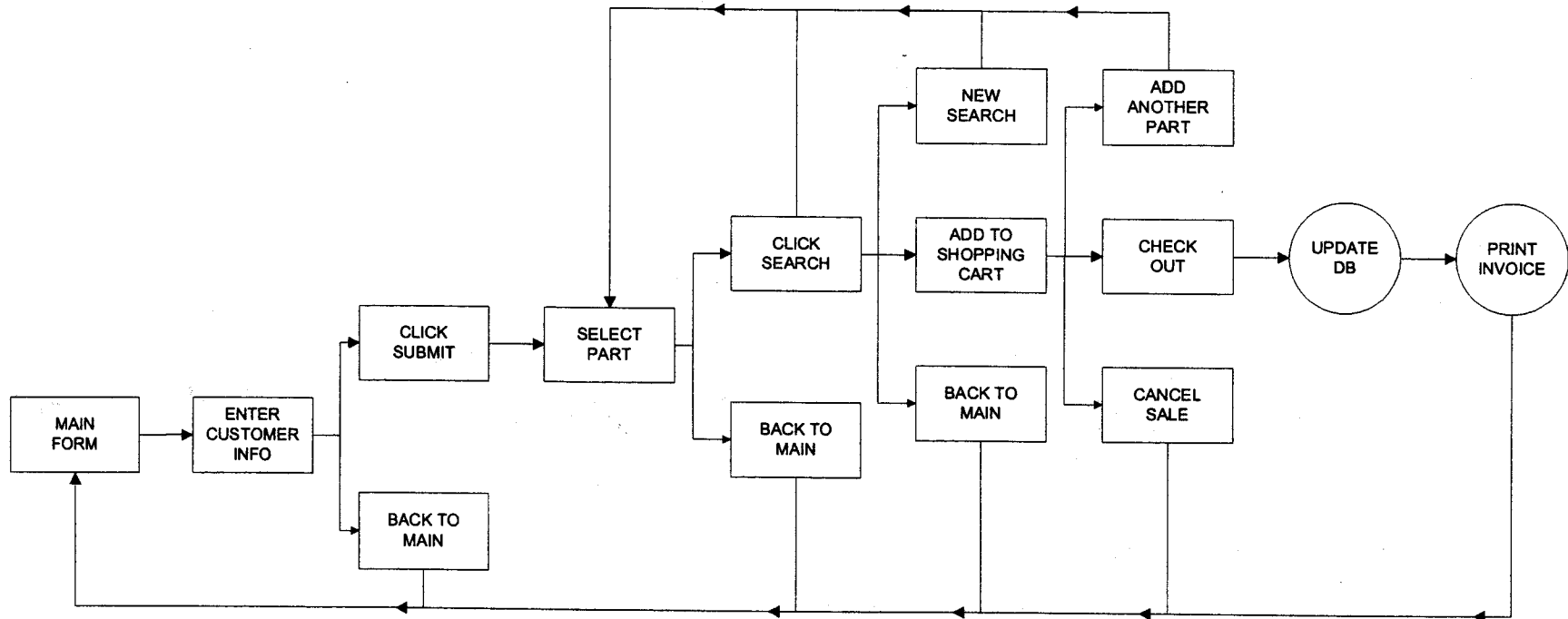
We are using the regular Microsoft Windows color scheme with gray tones, which is familiar to most people, and is relaxing for the eyes. We have focused more on the functionality of the project, but we will continue to improve the look and feel of the project. We will produce a user manual to go with this application, which will assist the user in installing the product as well as supporting all the functions implemented into the product.

7. Conclusions and Recommendations.

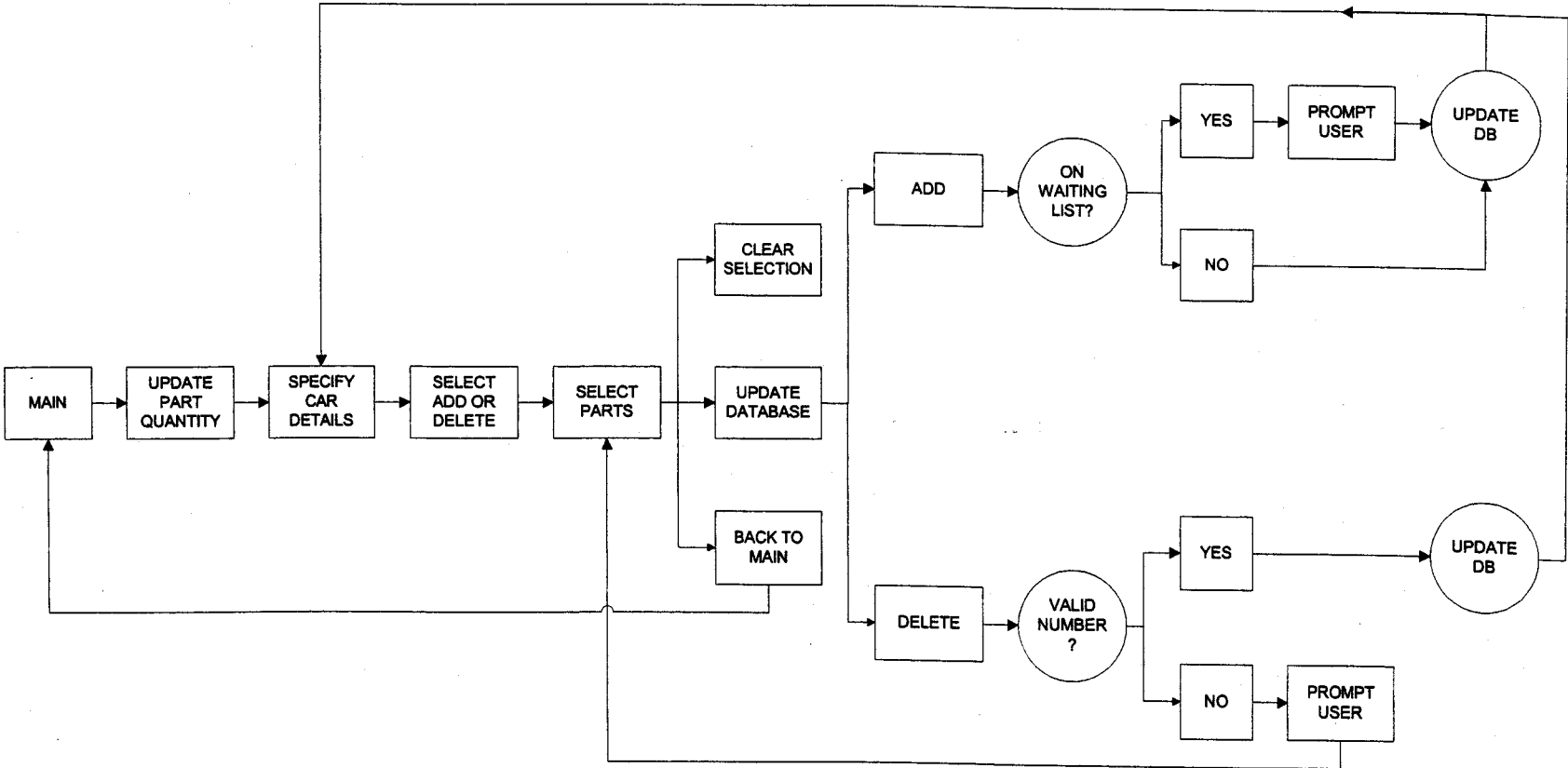
Overall we believe we have produced a professional product that meets and exceeds Senior Design requirements. The scope of this project was good for two people because there was a lot of work to do in this time frame and the work was very challenging. During the past three quarters we have learned a lot about Visual Basic and SQL Server and plan to use this knowledge in the future.

**APPENDIX: DOCUMENTATION
PARTS MANAGER 2000
AND PROJECT FLOWCHARTS**

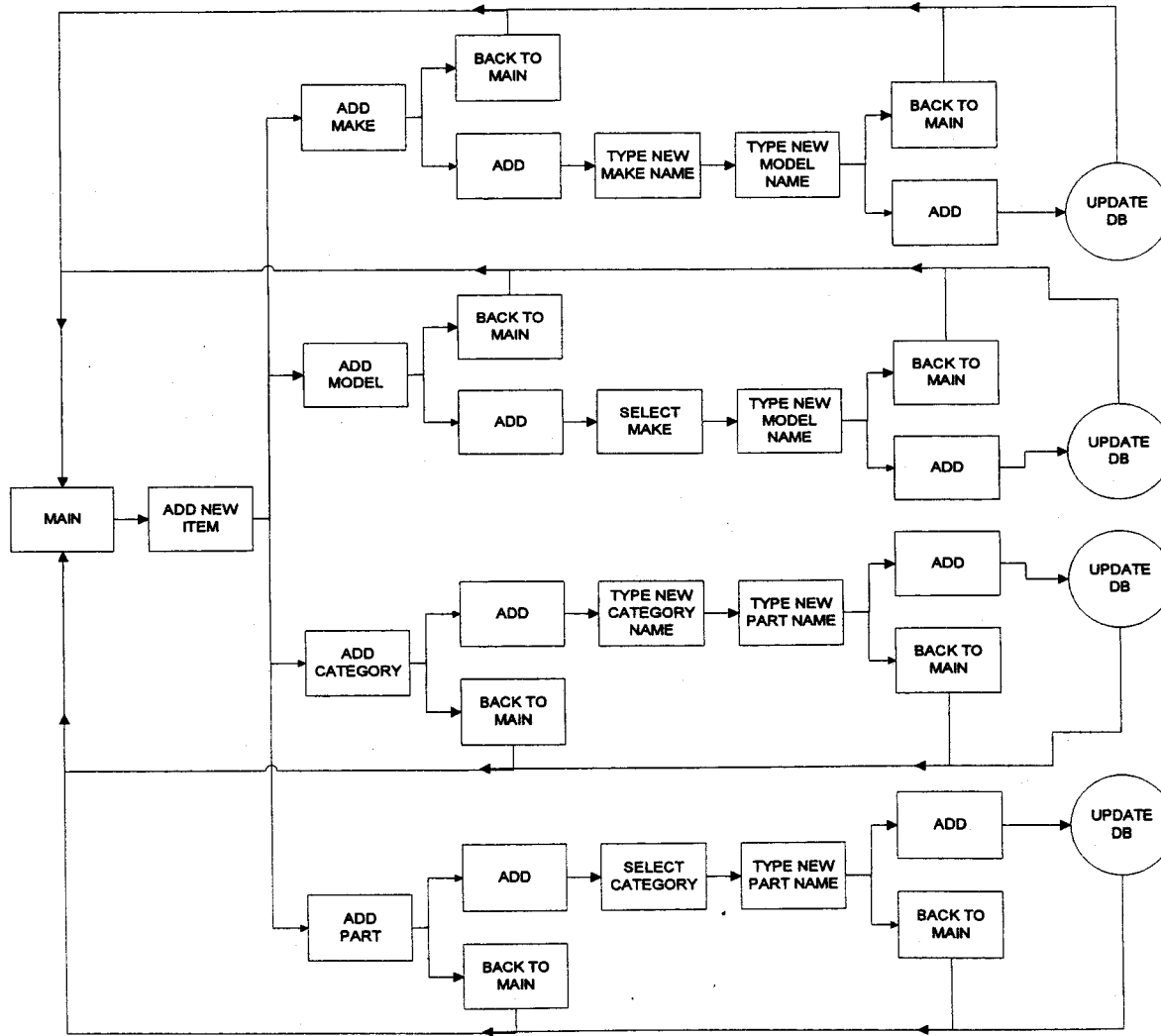
SEARCH FOR PARTS



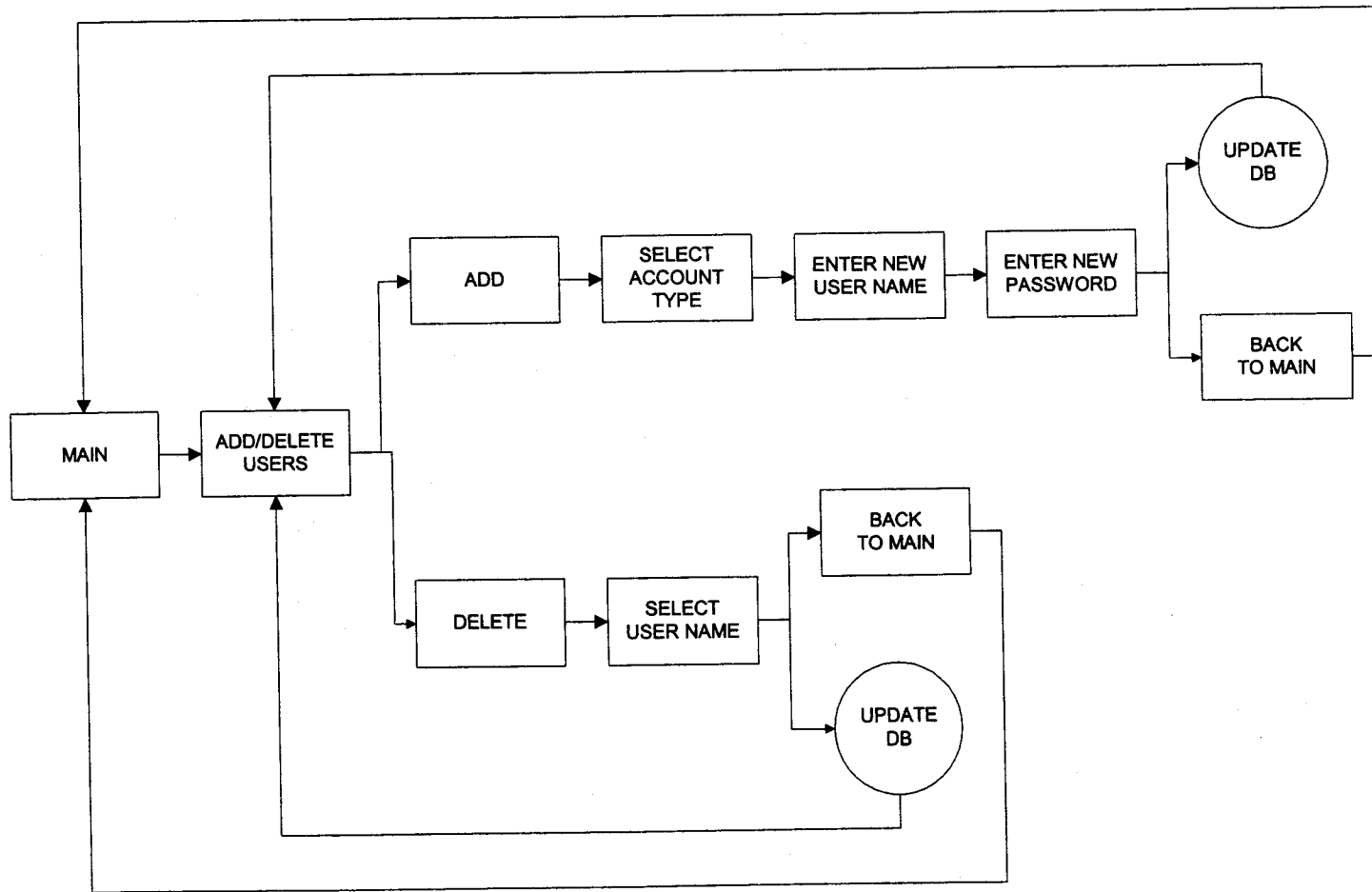
UPDATE PART QUANTITY



ADD NEW ITEMS



ADD/DELETE USERS



We are using the regular Microsoft Windows color scheme with gray tones, which is familiar to most people, and is relaxing for the eyes. We have focused more on the functionality of the project, but we will continue to improve the look and feel of the project. We will produce a user manual to go with this application, which will assist the user in installing the product as well as supporting all the functions implemented into the product.

7. Conclusions and Recommendations.

Overall we believe we have produced a professional product that meets and exceeds Senior Design requirements. The scope of this project was good for two people because there was a lot of work to do in this time frame and the work was very challenging. During the past three quarters we have learned a lot about Visual Basic and SQL Server and plan to use this knowledge in the future.

APPENDIX: DOCUMENTATION
PARTS MANAGER 2000

Documentation
Parts Manager 2000
Version 1.0.0

1. A database restore file is provided on the CD, it is called cpart.bkp. To restore the database you first need SQL Server 7 install on your computer. Once you do that you proceed to the open SQL Server and open Enterprise Manager. In Enterprise Manager you first create the database name to have SQL Server point and restore an initial database. Then you right click databases in Enterprise Manager and select *restore*. Then you proceed to the Restore Database, go to the *general* tab, under the *restore database as* drop down box and select the database name you initially created. Then if it is on a removable disk (zip disk), you hit the *restore from device* radio button. Then you will click on *Select Devices*, you will then be prompted to add you database by clicking *Add*. Your file must be placed on the server hard disk (i.e. C:\MSSQL7\BACKUP\) then you click *OK* and the restore will execute.
2. Launch partmgr.exe from the CD.
3. You will briefly see a splash screen as seen below in Figure 1.

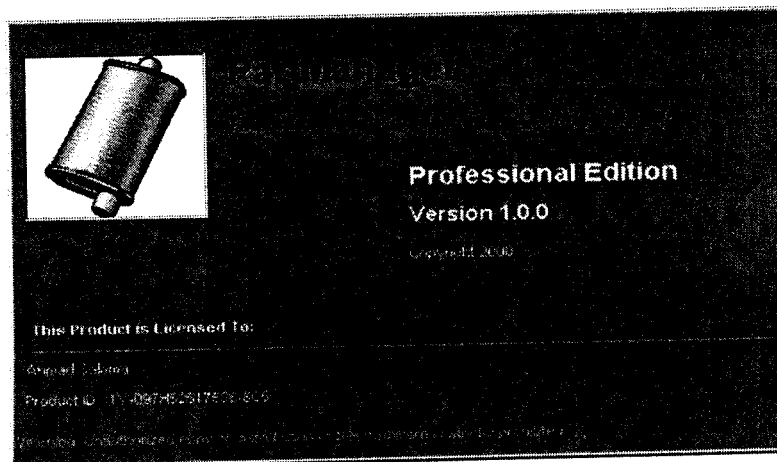


Figure 1. Splash Screen.

You will then be prompted to login as shown below in Figure 2.

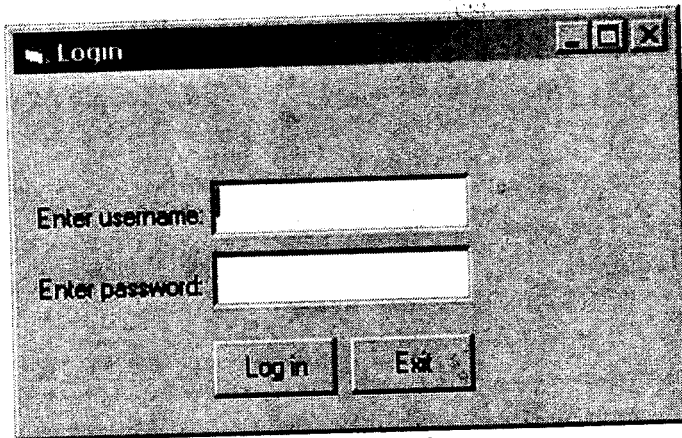


Figure 2. Login.

4. Based upon the rights set to your account you are either set as a salesman or manager. Below are the options set for the Salesman as shown below in Figure 3.

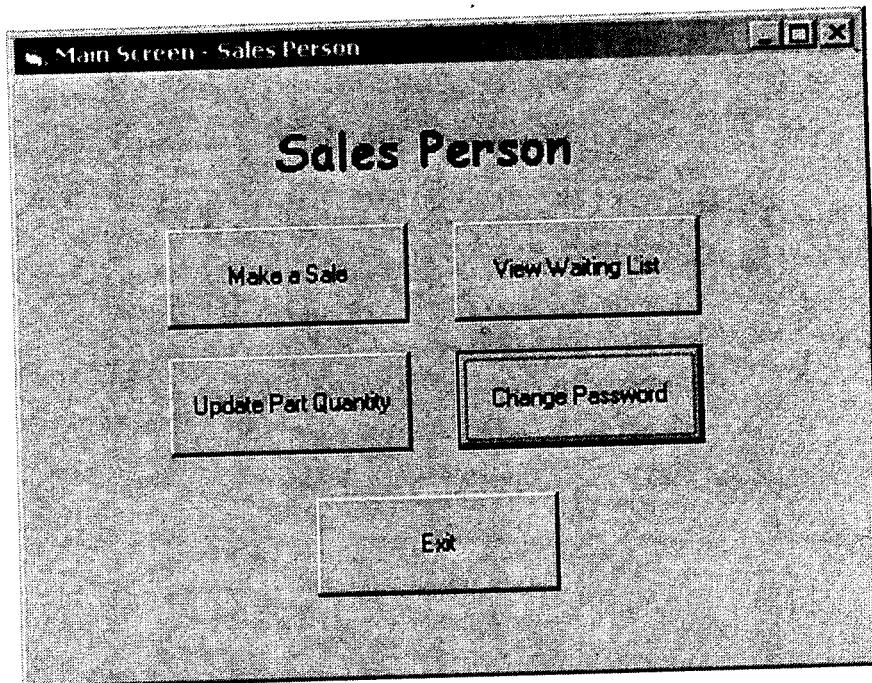


Figure 3. Salesman screen.

5. You will then go to the Manager screen and have 8 options to choose from as shown in Figure 4 on the top of the next page. Your options are Make a Sale,

Update Part Quantity, Add/Delete Parts, View Waiting List, Print Reports,
Add/Delete Users, Change Password, and Exit.

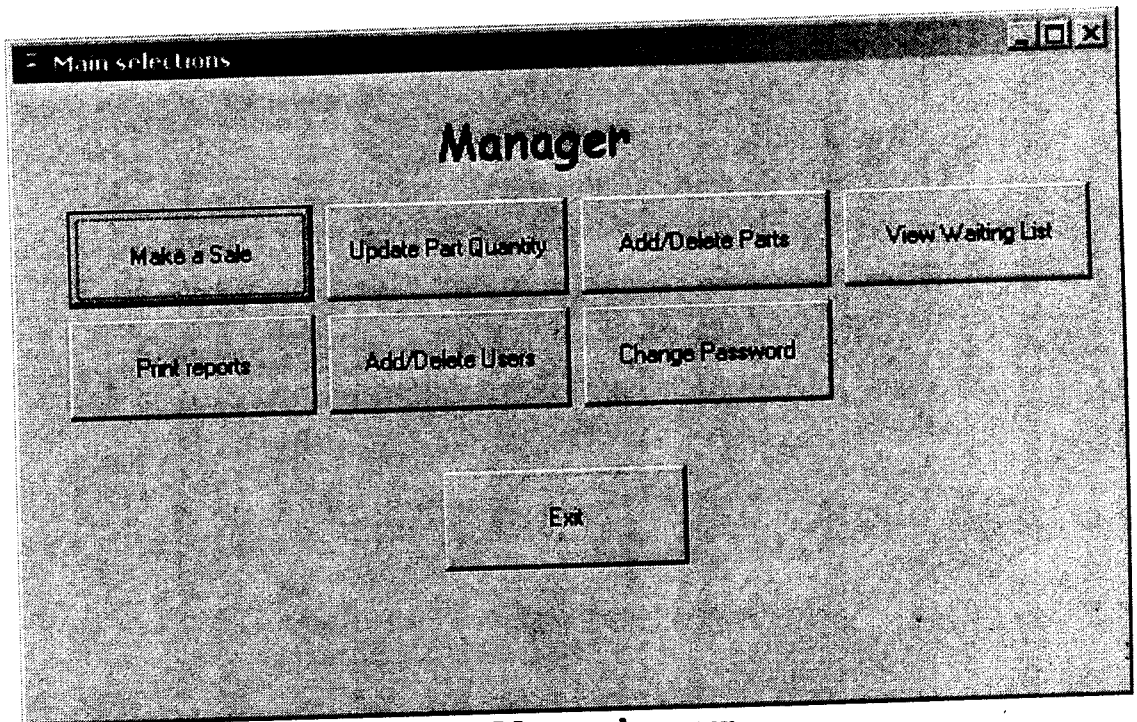


Figure 4. Manager's screen.

6. The first option is the Make a sale button as shown in Figure 5 on the top of the next page. When you click on Make a sale, you are prompted to enter Customer registration information.

Customer Registration

Enter Customer Information

Customer Number
1000

Last Name: Smith First Name: John

Address
123 Abbey Rd.

City: Cincinnati State: OH Zip: 45220

Telephone: (513)681-0987 Email: smith@hotmail.com

Submit Clear Fields Back to Main Screen

Figure 5. Make a sale.

7. Once you enter in all the customer information you click on Submit, you will then be prompted Search for parts screen as shown in Figure 6 on the top of the next page.

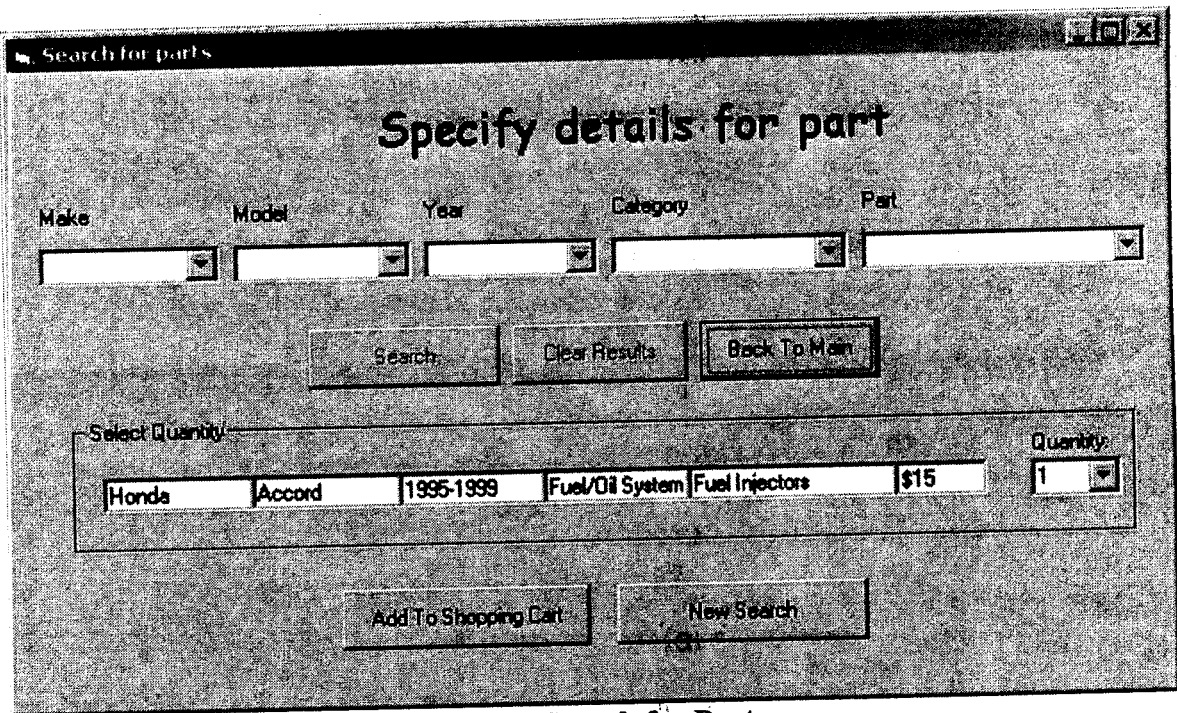


Figure 6. Search for Parts.

8. You now enter in the Make, Model, Year, Category of the part, and the specific part in the category and then hit search.
9. Next you hit Add to Shopping Cart, and you will see the following screen below in Figure 7.

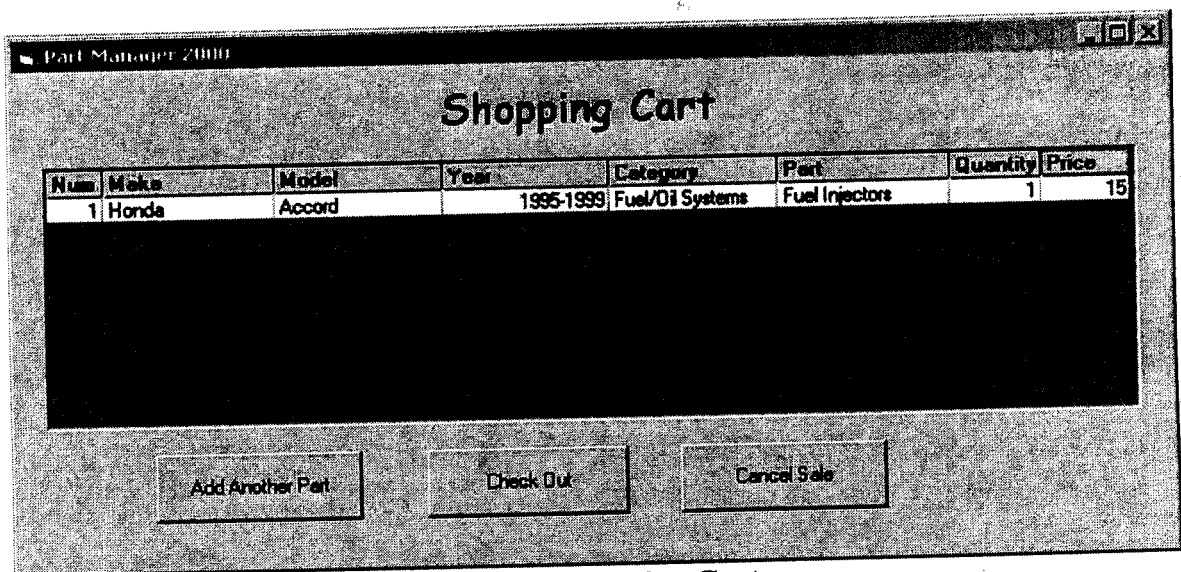


Figure 7. Shopping Cart.

The Shopping Cart will verify the quantity of the part, the Make, Model, Year, and price of the part. When you hit checkout it will print an invoice of your order. Cancel a Sale will return to the Manager's screen.

7. The next option is update part quantity as shown below in Figure 8.

Update Car Parts

Specify details for part to update

Add Parts Remove Parts

Make:
 Model:
 Year:

| Parts 1 | Parts 2 | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Body</p> <p><input checked="" type="checkbox"/> Bumper <input type="text" value="1"/></p> <p><input type="checkbox"/> Headlight <input type="text" value="0"/></p> <p><input type="checkbox"/> Hood <input type="text" value="0"/></p> <p><input checked="" type="checkbox"/> Left Fender <input type="text" value="1"/></p> <p><input type="checkbox"/> Right Fender <input type="text" value="0"/></p> <p><input checked="" type="checkbox"/> Spoiler <input type="text" value="1"/></p> <p><input type="checkbox"/> Trunk <input type="text" value="0"/></p> <p>Brake System</p> <p><input type="checkbox"/> Rotor <input type="text" value="0"/></p> <p><input type="checkbox"/> Emergency Brake <input type="text" value="0"/></p> <p><input type="checkbox"/> Master Cylinder <input type="text" value="0"/></p> <p><input type="checkbox"/> Drum Brake Assembly <input type="text" value="0"/></p> | <p>Exterior</p> <p><input type="checkbox"/> Doors <input type="text" value="0"/></p> <p><input type="checkbox"/> Lug Nuts <input type="text" value="0"/></p> <p><input type="checkbox"/> Rear Brake Lamp <input type="text" value="0"/></p> <p><input type="checkbox"/> Rear Light <input type="text" value="0"/></p> <p><input type="checkbox"/> Side Marker Light <input type="text" value="0"/></p> <p><input checked="" type="checkbox"/> Side mirror <input type="text" value="1"/></p> <p><input type="checkbox"/> Spare Tire <input type="text" value="0"/></p> <p><input type="checkbox"/> Tires <input type="text" value="0"/></p> <p><input type="checkbox"/> Wheel Cap <input type="text" value="0"/></p> <p><input type="checkbox"/> Windows <input type="text" value="0"/></p> <p><input type="checkbox"/> Wiper Motor <input type="text" value="0"/></p> | <p>HVAC</p> <p><input type="checkbox"/> A/C Compressor <input type="text" value="0"/></p> <p><input type="checkbox"/> Heater Core <input type="text" value="0"/></p> <p><input checked="" type="checkbox"/> Radiator <input type="text" value="1"/></p> <p>Fuel/Oil Systems</p> <p><input type="checkbox"/> Fuel Injector <input type="text" value="0"/></p> <p><input type="checkbox"/> Fuel Pump <input type="text" value="0"/></p> <p><input type="checkbox"/> Fuel Tank <input type="text" value="0"/></p> <p><input type="checkbox"/> Oil Pan <input type="text" value="0"/></p> <p><input type="checkbox"/> Oil Pump <input type="text" value="0"/></p> |

Figure 8. Update Car Parts.

This function allows you to add/remove parts from the database. You can select the type of part and update or remove the quantity. Two tabs have been provided to list all of the types of parts in the database.

8. The next option is Add/Delete Parts as shown below in Figure 9.

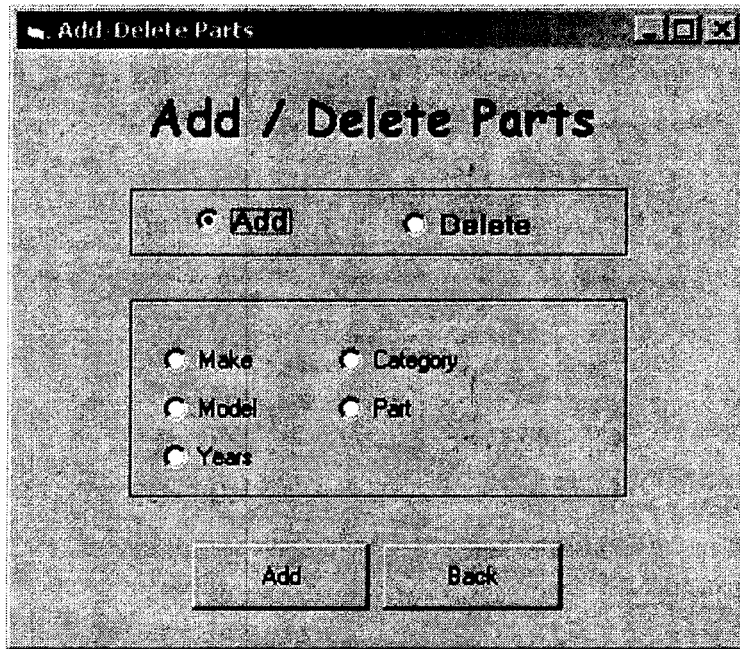


Figure 9. Add/Delete Parts.

This function allows you to add/delete parts based upon the Make, Model, Year, Category or type of part in the database. This function is helpful to add a new manufacturer (Make) or a new Model to an existing car manufacturer. It also helps to add new parts to an existing category or create a new category.

9. Next is the Print report option as shown on the top of the next page in Figure 10.

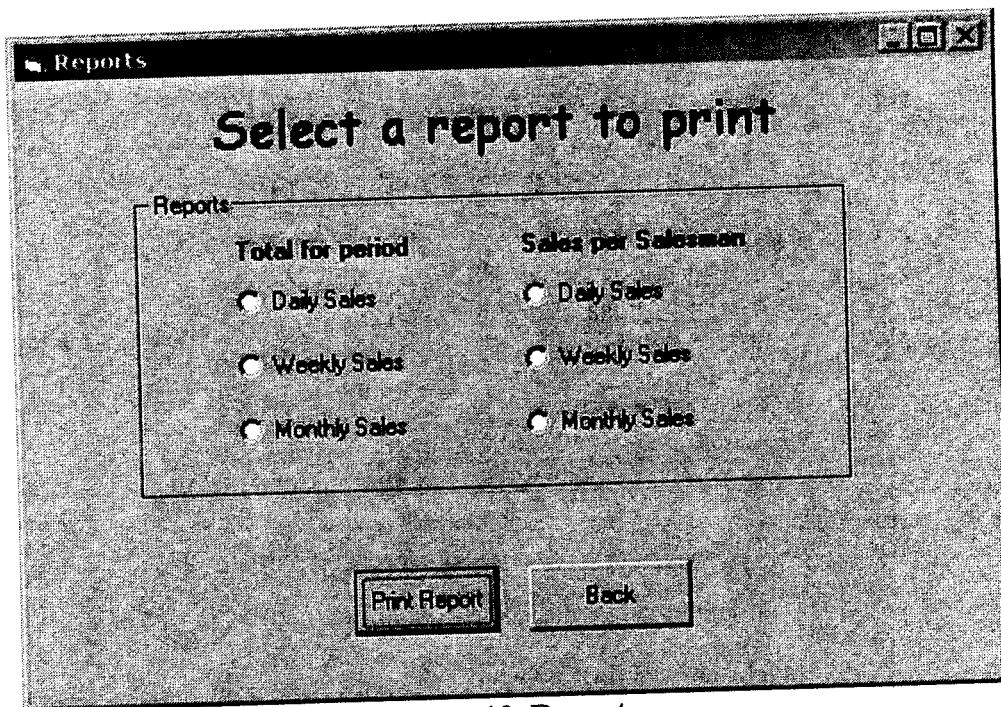


Figure 10. Reports.

The report function allows you to print report based upon daily, weekly, and monthly sales. The report function also allows you to print sales based on the salesman daily, weekly, or monthly sales.

10. The next option is Add/Delete Users as shown on the top of the next page in Figure 11.

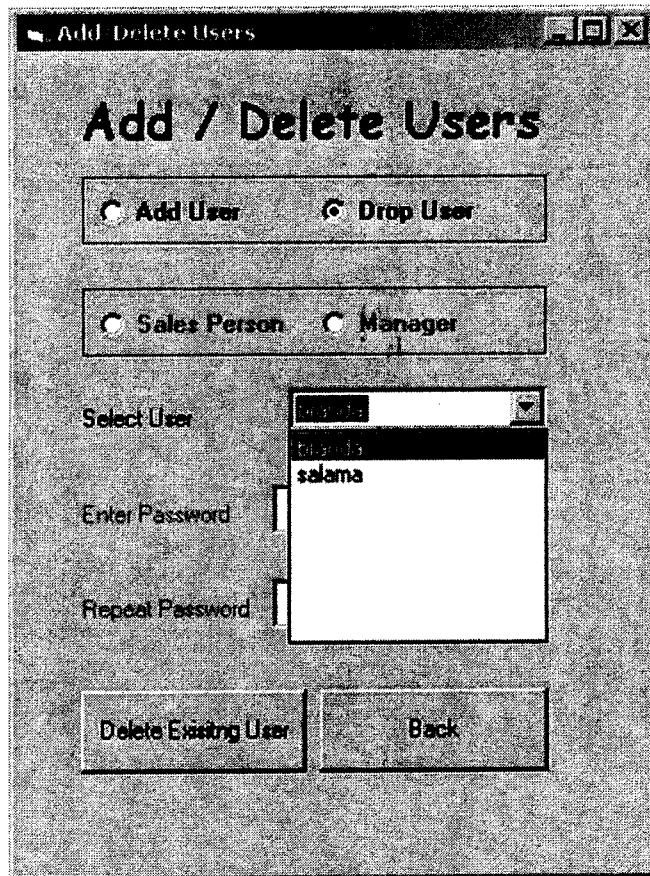


Figure 11. Add/Delete Users.

11. Next is the Change Password option as shown on the top of the next page Figure 12.

Change Password

Username: sa

Enter Current Password: []

Enter New Password: []

Repeat New Password: []

Change Password Back

Figure 12. Change Password.

12. The Waiting List option is the Waiting List as shown below in Figure 13.

Waiting List

Customer Details

Customer ID: 1010

Last Name: Uskov First Name: Vladimir

Address: 2134 Programming St.

City: Chicago State: IL Zip: 56821

E-mail: Uskov@chicago.edu Phone: (401)555-1234

Waiting List Details

Make: Isuzu Model: Amigo

Year: 2000-2004 Partname: Bumper

Preferred way of notice: Phone E-Mail US Mail

How many months on waiting list? 3

Edit Customer Details Add To Waiting List Cancel

Figure 13. Waiting List.

This option is available if a car part is not available. The customer has the option to be put on this waiting list for up to 12 months. The manager will then be notified when the car part becomes available for that specific customer, and finally contact the customer by Phone, E-mail, or US Mail.

Finally, once the clicks the *Add to Waiting List* button the following screen in Figure 14 appears below.

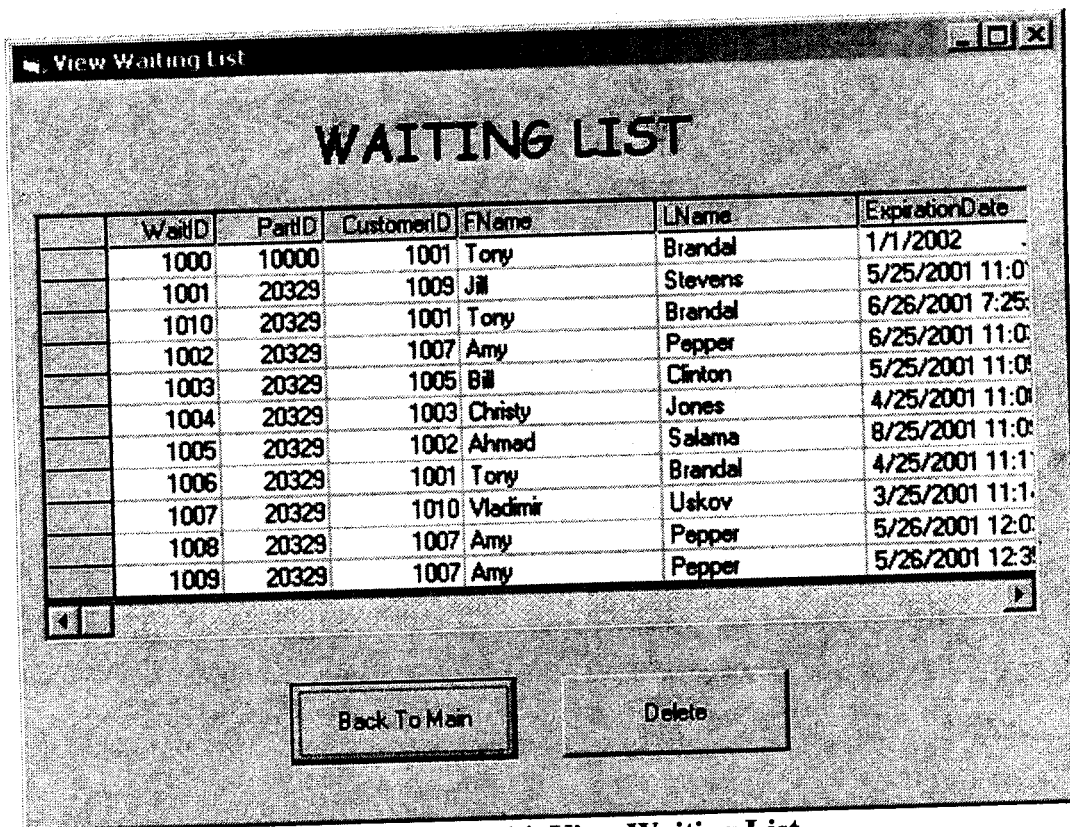


Figure 14. View Waiting List.

The waiting list features all the customers that currently are waiting for parts, it list the Wait ID, Part ID, Customer ID, First Name, L