

A Novel Approach for Automobile Customer Requisites

Dr.S.Mohan kumar,

New Horizon College of Engineering, Research Centre, Bangalore, India.

1.Introduction

Customer database refers to managing the entire lifecycle of activities linked to the fulfillment of a service requests made by and other stakeholders. It can eliminate a great deal of paperwork for business, providing a single repository for valuable client information that can be used by sales teams. It helps in unifying all customer information in one place as per need, ensuring faster responses and enabling customer satisfaction. It helps in registering and monitoring users, enforcing data security, monitoring performance, maintaining data integrity, dealing with concurrency control and recovering information corrupted by unexpected failure. Attracting and retaining customers requires a clear understanding of customer requirements.

When a customer requests for an automobile spare part, if the part is not available, the merchant can take the order by storing necessary information like customer name, mobile number, vehicle name, part required etc., into a database where the merchant can view, manipulate, store data required in an ordered and efficient manner. This process saves time as referring to a manual record is time consuming compared to searching on a digital database. With minimal computer knowledge a person can easily use this program.

1.1Purpose of Study

Data can be stored in some database but what matters is how efficiently the stored data can be retrieved. Raw data has no meaning unless it is put up in an ordered fashion. The purpose of this project is to demonstrate simple ways to store, manipulate and retrieve meaningful data as in when required.

1.2Problem Statement

- To design and implement an efficient way to store and retrieve order and pre order information of the customer with their details.
- Develop an application to store and retrieve client data.
- To design and implement a customer database with password protection and string comparisons.

1.3 Motivation of project

Every company wants to serve their customers better, and one of the best ways to achieve customer satisfaction is by serving them “in time”. To ensure an effective solution to provide requested order and to make precise business decisions, it was necessary to develop an effective logic that reduces human efforts and increases efficiency.

1.4 Methodology

- User enters a password to login.
- Main menu will be displayed if password is correct.
- User can enter customer details by selecting create client option.
- All the data is stored in a word document.
- Now based on the query, the user can perform operations like search, modify, delete etc.,
- If required user can also create a backup file using the create back up option.
- Exit (user logout).

2.System Requirements

2.1 Hardware and Software Requirements

Hardware System Configuration:

Processor - Intel Pentium

Speed - 1.6 GHz

RAM - 2 GB

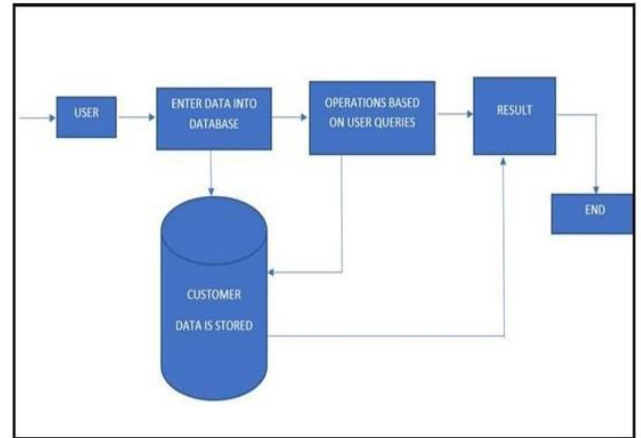
Hard Disk - 10 GB

Software System Configuration:

Operating System - Windows

Programming Language - C

Compiler - C Compiler



3.1.1 Architecture diagram

2.2 About the Language

C is a procedural programming language. It was initially developed by Dennis Ritchie between 1969 and 1973. It was mainly developed as a system programming language to write operating system. The main features of C language include low-level access to memory, simple set of keywords, and clean style, these features make C language suitable for system programming like operating system or compiler development. Many later languages have borrowed syntax/features directly or indirectly from C language. Like syntax of Java, PHP, JavaScript and many other languages is mainly based on C language. C++ is nearly a superset of C language. There are many compilers available freely for compilation of C programs like Code Blocks and Dev-CPP.

Advantages of c language:-

- Easy to learn
- Structured language
- It produces efficient programs
- It can handle low-level activities
- It can be compiled on a variety of computer platforms

C was initially used for system development work, particularly the programs that make-up the operating system. C was adopted as a system development language because it produces code that runs nearly as fast as the code written in assembly language.

3. System Design

3.1 Architecture

3.2 Algorithm

Step 1: Start

Step 2: Display the options to –

1. User Input
2. List records
3. Search records
4. Modify record
5. Delete record
6. Modify advance received
7. Create backup
6. Exit

Step 3: If the entry is „1“, then take the details of the customer and store it into the file.

If entry is „2“, then load the details of all the clients from the file and display the details.

If entry is „3“, then load the details of particular clients from the file and display the details.

If entry is „4“, then take the name and search, load the details from the file. If found then ask all the details of the customer and update the file with the same.

If entry is „5“, then take the name and search, if found then copy rest of the customers details into a temp file except the current one and at the end rename the file to the original name

and delete the old file.

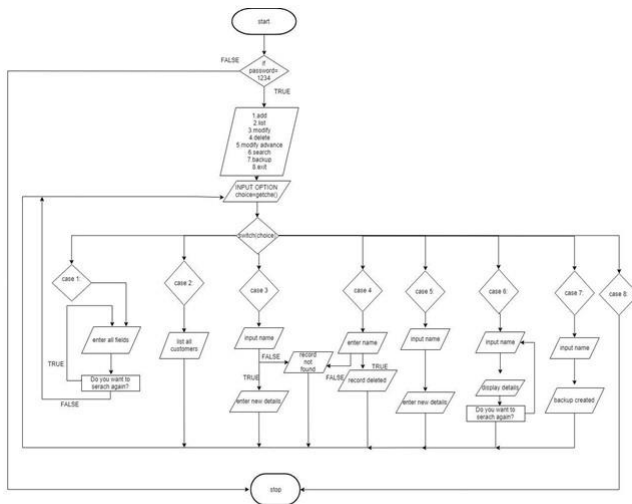
If entry is „6“, then take the name and search, load the details from the file. If found then ask amount received at delivery and update the file.

If entry is „7“, then create a new file with existing data and save it for backup.

If entry is „8“, then exit from the program.

Step 4: Stop

3.3 Flow Chart



3.3.1 Flow Control

3.4 Code

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include<string.h>
#include<ctype.h>
#include<malloc.h>
#include<dos.h>
struct account {
    int number,day,month,year;
    char name[100];
```

```
char item_name[100];
float mobile_no;
char vn[100];
char acct_type;
float tokenadvance;
float tokenadvance1;
float newbalance;
float payment;
char date[100];
char date1[100];
int dd,mm,yy;
struct account *next;
struct account *prev;
}customer;
```

```
struct account
*head=NULL,*temp,*disp,*add,*trav,*del,*hnext,*hprev;
long fileSize;
int numEntries,loop;
int g;
```

```
int checkdate(int aa, int bb,int cc,int dd,int ee,int ff)
{
    if(dd>=aa && ee>=bb && ff>=cc)
```

```
return 1;
else
return 0;
}
int check(char a[50])
{
    int i;
    for(i=0;i<strlen(a);i++)
    {
        if(isdigit(a[i])!=0)
        {
            return 0;
        }
    }
```

```

    } }

//for searching a client

void PrintList(struct account *start)
{
    struct account *current = start;
    int count = 0;
    char nam[100];
    struct account *ahead = NULL;
    struct account *behind = NULL;
    while(current != NULL) {
        count++;
        ahead = current->next;

        behind = current->prev;
        printf(" Name  :%s\n",current->name);
        printf(" Mobile no  :%.f\n",current->mobile_no);
        printf(" Item name  :%s\n",current->item_name);
        printf(" Vehicle name  :%s\n",current->vn);
        printf("Balance to be paid :%.f\n",current->newbalance);
        printf("payment                                date
:%d/%d/%d\n",current->day,current->month,current->year);
        printf("delivery                                date
:%d/%d/%d\n",current->dd,current->mm,current->yy);
        printf("\n Press any key to continue\n");
        getchar();

        current = current->next;
        ahead = NULL;
        behind = NULL;
    } }

void WriteListToNewFile(struct account *start)
{
    FILE *pt;

    pt= fopen("pq.dat", "ab+");

    if(pt != NULL)
    {
        trav=start;
        hnext = NULL;
        hprev = NULL;

        while(trav != NULL)
        {
            hnext = trav->next;
            hprev = trav->prev;

            trav->next = NULL;
            trav->prev = NULL;
            fseek(pt, 0, SEEK_END);
            fwrite(trav, sizeof(struct account), 1, pt);

            printf("Writing:%s to file\n",trav->name);

            trav->next = hnext;
            trav->prev = hprev;

            hnext = NULL;
            hprev = NULL;
            trav = trav->next;
        }
        fclose(pt);
        pt= NULL;
    }
    else
    {
        printf("FILE OPEN ERROR\n");
    }
}

```

```

}
void WriteListToFile(struct account *start)
{
FILE *pF;
pF = fopen("pp.dat", "ab+");
if(pF != NULL)
{
trav=start;
hnext = NULL;
hprev = NULL;
while(trav != NULL)
{
hnext = trav->next;
hprev = trav->prev;

trav->next = NULL;
trav->prev = NULL;
fseek(pF, 0, SEEK_END);
fwrite(trav, sizeof(struct account), 1, pF);

printf("Writing:%s to file\n",trav->name);

trav->next = hnext;
trav->prev = hprev;

hnext = NULL;
hprev = NULL;

```

4.Results and Discussion

4.1 Summary of result obtained

The goal of this project is to store , retrieve and manipulate customer data in a efficient way. A merchant can store necessary information like customer name, phone number, vehicle name, item requested, item value, token advance received etc., the program has a very simple user interface which can be used by a person with minimal computer

knowledge. As data is stored in ordered fashion, this helps the merchant to concentrate better on procuring the required parts and avoid the hassle of maintaining a manual record where at times a bad handwriting would create lot of inconsistent information which would eventually effect the merchant client relationship.This program has a simple menu based interface, initially to access the features, the merchant should enter a preset password, upon authentication the menu page is displayed. Here options like create record, modify client record, modify amount received, create back up etc., options are displayed using which the merchant can meet his requirements. All the data is stored in a word document. Features like string validation for name and date are developed i.e., client name cannot take a numeric value and delivery date cannot be prior to the date the order was placed.

4.2 Output (Snapshots)



4.1 Results

5.Conclusion

- The proposed system is an attempt to remove existing flaws in the manual method of data storage by opting to store information on a digital database.
- For automobile merchants this program will help in organizing data in effective manner.
- A merchant can keep track of the orders, search for customer details, and modify the records as well. This program is very user friendly.
- A person with minimal computer knowledge can make use of this program.

- This helps unifying all customer information in one place as per need, ensuring faster responses and enabling customer satisfaction.

6.References

1. Modern C++ Design book by Andreai alexandrescu by Addison wesley
2. C programming language by Brian Kernighan and Dennis Ritchie
3. C++ primer plus by Stephen Preta published by Scott Meyers

Citation:

1. <https://www.geeksforgeeks.org/a-codelinked-intro/>
2. https://en.programmerhack/Customer_data_management
3. <https://www.geeksforgeeks.org/c-language-set-1-introduction/>
4. <https://www.hackerearth.com/practice/data-structures/linked-list/singly-linked/>
5. https://en.linkedlist/Customer_data.org
6. https://en.hackout/cprog/files_data_store/