DELHI PUBLIC SCHOOL
BANGALORE NORTH

Computer Science Project Report

"Virtual Shopping Application using C++"

TABLE OF CONTENTS

SYNOPSIS

This is a project based on e-commerce, specifically an electronics shopping application, called ‘MERX’. The program helps a user browse for different items based on his/her liking and manages a cart to handle the ordering of required items. The program also displays the invoice at the end of shopping. Customers can either login using existing credentials or by creating a new account. Employees have pre alotted credentials which they can use to handle inventory, feedback, etc.

The program includes various function programs to do the above mentioned tasks. Data file handling and classes have been effectively used in the program.

It provides standard features of an e-commerce site like:

CUSTOMERS:
Register (for new users)
Login (Existing users)
View products listed on the site
Select products and add them to cart
View cart contents
Checkout the cart contents for order processing
Input feedback, or complaint regarding their order(s).

EMPLOYEES:
Login
Add items to inventory
Update item(s) in inventory
Remove a product from inventory
View customer feedback, complaints

OBJECTIVE

To develop a C++ project to manage an online shopping system using object oriented programming and data file handling.

HEADER FILES USED

- **conio.h:** For clrscr() and getch()
- **stdio.h:** For standard I/O operations
- **string.h:** For string operations
- **process.h:** For exit()
- **fstream.h>**: For file handling operations,cin and cout

CLASSES USED

1)
```
class product
{   int serialno;
    char name[50],category[20];
    float price,qty;
    void create_product();
    void show product();
    int retserialno();
    float retprice();
    char* retname();
    char* retcategory();
};
```

2)
```
class cussig
{
    char fname[20], lname[20], addr[80],
    pno[15],email[100],city[20];
    long int pincode;
    public:
    void add();
    void show();
    void mod();
}o1;
```

3)
```
class cuslog
{
    char uname[20],password[20];
    public:
    void username();
    void pass();
    char* retusername();
    char* retpass();
}o2,o3;
```

DATA FILES USED

TEXT FILES USED

- ✓ **cusr.txt** – For storing customer usernames
- ✓ **cpss.txt** – For storing customer passwords
- ✓ **emp.txt** – For storing employee usernames
- ✓ **def.txt** – For storing employee passwords
- ✓ **cart.txt** – For storing items in cart

BINARY FILES USED

- ✓ **fdbck.dat** – For storing customer feedback
- ✓ **cmplnt.dat** – For storing customer complaint
- ✓ **csig.dat** – For storing customer details
- ✓ **shop.dat** – For storing inventory
- ✓ **temp.dat** – Used for binary file operations

SOURCE CODE

```
#include<conio.h>
#include<stdio.h>
#include<string.h>
#include<process.h>
#include<iostream.h>
#include<fstream.h>
#include<graphics.h>
#include<iomanip.h>
#define SLNO 3
#define NAME 23
#define PRICE 15
#define CATG 13
#define QNTY 17
void mainmenu();
void viewcart();
class product
{
    int serialno;
    char name[50],category[20];
    float price;
    public:
    float qty;
    void create_product()
    {
        cout<<"\n\n\tENTER PRODUCT DETAILS";
        cout<<"\n\n\tProduct ID: ";
        cin>>serialno;
        cout<<"\tProduct name: ";
        gets(name);
        cout<<"\tProduct price: ";
        cin>>price;
        cout<<"\tCategory: ";
```

```
    cin>>category;
    cout<<"\tQuantity: ";

    cin>>qty;
}
void show_product()
{
    cout<<"\n\n\tINVENTORY";
    cout<<"\n\n\tProduct ID: "<<serialno;
    cout<<"\n\tProduct name: ";
    puts(name);
    cout<<"\n\tProduct price: "<<price;
    cout<<"\n\tCategory: ";
    puts(category);
    cout<<"\n\tQuantity: "<<qty;
}
void menu2()
{
    cout<<setw(SLNO)<<serialno
        <<setw(NAME)<<name
        <<setw(PRICE)<<price
        <<setw(CATG)<<category
        <<setw(QNTY)<<qty<<endl;
}
int retserialno()
{
    return serialno;
}
float retprice()
{
    return price;
}
char* retname()
{
    return name;
}
char* retcategory()
```

```
    {
        return category;
    }
    float retqty()
    {
        return qty;
    }
};
class cussig
{
    char
fname[20],lname[20],addr[80],pno[15],email[100],c
ity[20];
    long int pincode;
    public:
    void add();
    void show();
    void mod();
}o1;
class cuslog
{
    char uname[20],password[20];
    public:
    void username();
    void pass();
    char* retusername()
    {
        return uname;
    }
    char* retpass()
    {
        return password;
    }
}o2,o3;
void cuslog::username()
{
    clrscr();
```

```
    cout<<"\n\n\n\n\n\n\tEnter username: ";
    gets(uname);
}
void cuslog::pass()
{
    cout<<"\n\tEnter password: ";
    gets(password);
}
void cussig::add()
{
    clrscr();
    cout<<"\n\n\n\n\tEnter first name: ";
    gets(fname);
    cout<<"\tEnter last name: ";
    gets(lname);
    cout<<"\tEnter address: ";
    gets(addr);
    cout<<"\tEnter phone no.: ";
    cin>>pno;
    cout<<"\tEnter email id: ";
    gets(email);
    cout<<"\tEnter pincode: ";
    cin>>pincode;
    cout<<"\tEnter city: ";
    gets(city);
}
void cussig::show()
{
    cout<<"\n\tFirst name: "<<fname
        <<"\n\tLast name: "<<lname
        <<"\n\tAddress: "<<addr
        <<"\n\tPhone no.:"<<pno
        <<"\n\tEmail id: "<<email
        <<"\n\tPincode: "<<pincode
        <<"\n\tCity: "<<city;
};              //class ends here
// global declarations for stream object, object
```

```
fstream fp;
product pr;
fstream cs;
// function to write in file
void write_product()
{
    char ab;
    do
    {
        fp.open("Shop.dat",ios::out|ios::app);
        pr.create_product();
        fp.write((char*)&pr,sizeof(pr));
        fp.close();
        cout<<"\n\t"<<pr.retname()<<" has been
added to the inventory.";
        cout<<"\n\n\tAdd more products? (Y/N)";
        ab=getche();
    }while(ab=='y'||ab=='Y');
    getch();
}
// function to read all records from file
void display_all()
{
    clrscr();
    cout<<"\n\t\tDisplaying all products..\n\n";
    fp.open("Shop.dat",ios::in);
    while(fp.read((char*)&pr,sizeof(product)))
    {
        pr.show_product();
        cout<<"\n\t";
        system("PAUSE");

    cout<<"\n\n================================
==\n";
    }
    fp.close();
    getch();
```

```
}
// function to read specific record from file
void display_sp(int n)
{
    int flag=0;
    fp.open("Shop.dat",ios::in);
    while(fp.read((char*)&pr,sizeof(product)))
    {

        if(pr.retserialno()==n)
        {
            clrscr();
            pr.show_product();
            flag=1;
        }
    }
    fp.close();
    if(flag==0)
    cout<<"\n\n\tPRODUCT DOES NOT EXIST";
    getch();
}
// function to modify record of file
void modify_product()
{
    int no,found=0;
    clrscr();
    cout<<"\n\tEnter product ID to modify de-
tails\n\t";
    cin>>no;
    fp.open("Shop.dat",ios::in|ios::out);
    while(fp.read((char*)&pr,sizeof(product)) &&
found==0)
    {
        if(pr.retserialno()==no)
        {
            pr.show_product();
```

```
            cout<<"\n\tPlease enter new details
of the product"<<endl;
            pr.create_product();
            int pos=-1*sizeof(pr);
            fp.seekp(pos,ios::cur);
            fp.write((char*)&pr,sizeof(product));
            cout<<"\n\tRecord Updated";
            found=1;
        }
    }
    fp.close();
    if(found==0)
    cout<<"\n\tRecord Not Found";
    getch();
}
// function to delete record of file
void delete_product()
{
    int no;
    clrscr();
    cout<<"\n\tPlease enter product ID to remove
from inventory\n\t";
    cin>>no;
    fp.open("Shop.dat",ios::in|ios::out);
    fstream fp2;
    fp2.open("Temp.dat",ios::out);
    fp.seekg(0,ios::beg);
    while(fp.read((char*)&pr,sizeof(product)))
    {
        if(pr.retserialno()!=no)
        {

    fp2.write((char*)&pr,sizeof(product));
        }
    }
    fp2.close();
    fp.close();
```

```
    remove("Shop.dat");
    rename("Temp.dat","Shop.dat");
    cout<<"\n\tProduct deleted from inventory!";
    getch();
}
// function to display all products price list
void menu()
{
    clrscr();
    fp.open("Shop.dat",ios::in);
    if(!fp)
    {
        cout<<"\tERROR 404\n\tGo to admin menu to
create file";
        getch();
        mainmenu();
    }
    qw:
    clrscr();
    cout<<"\n\n\n\n\tChoose a category to view
items\n\n\t\5 Mobiles\n\t\5 Tablets\n\t\5 Con-
soles\n\t";
    cout<<"\n\t->Press X to sign out\n\t";
    char chcat[100];
    int ctr=0;
    gets(chcat);
    if((strcmp(chcat,"x")==0)||(strcmp(chcat,"X")
==0))
        exit(0);
    else
if((strcmpi(chcat,"mobiles")!=0)&&(strcmpi(chcat,
"tablets")!=0)&&(strcmpi(chcat,"consoles")!=0))
    {
        cout<<"\n\tPlease enter a valid option";
        getch();
        goto qw;
    }
```

```
    else
    {
        clrscr();
        cout<<"\n\n\tPRODUCT MENU\n\n";

    cout<<"=====================================
=====================================\n";
        cout<<setw(SLNO)<<"P.NO."
            <<setw(NAME)<<"NAME        "
            <<setw(PRICE)<<"PRICE  "
            <<setw(CATG)<<"CATEGORY  "
            <<setw(QNTY)<<"  ITEMS LEFT IN
STOCK"<<endl;

    cout<<"=====================================
=====================================\n";

    while(fp.read((char*)&pr,sizeof(product)))
        {
            char* cat=pr.retcategory();
            if(strcmpi(cat,chcat)==0)
            {
                pr.menu2();
                ctr++;
            }
        }
        fp.close();
    }
}
// function to place order and generating bill
for Products
void place_order()
{
    int order_arr[50],c=0,y=0;
    int cart[50];
    float quan[50],amt,damt,total=0,stck;
    char ch,xz,ch1;
```

```
ofstream xout("cart.txt",ios::out,ios::app);
cout<<"\n\n\t============================";
cout<<"\n\tPLACE YOUR ORDER";
cout<<"\n\t============================\n";
do
{
    if(ch=='y'||ch=='Y')
    {
        menu();
        cout<<"\n\tAdd items to cart?(Y/N)";
        xz=getche();
        if(xz=='y'||xz=='Y')
        goto abc;
        if(xz=='n'||xz=='N')
        goto def;
    }
    abc: cout<<"\n\tEnter product ID: ";
    cin>>order_arr[c];
    cout<<"\tQuantity: ";
    cin>>quan[c];
    c++;
    if(!xout)
    {
        xout.open("cart.txt",ios::app);
    }
    int ghi=0;
    for(int x=y;x<=c;x++)
    {
        fp.open("Shop.dat",ios::in);
        fp.read((char*)&pr,sizeof(product));
        while(!fp.eof())
        {


if(pr.retserialno()==order_arr[x])
            {
```

```
                    xout<<"\nProduct ID: "<<or-
der_arr[x]<<"  ";
                    xout<<"\nProduct Name:
"<<pr.retname()<<"  ";
                    xout<<"\nPrice:
"<<pr.retprice()<<"  ";
                    ghi=1;
                }


    fp.read((char*)&pr,sizeof(product));
            }

            xout.close();
            fp.close();
        }
        if(ghi==0)
        cout<<"\n\tSorry, product does not ex-
ist";
        cout<<"\n\t->Press X to view cart\n\t";
        ch1=getche();
        if(ch1=='x'||ch1=='X')
        {
            clrscr();
            cout<<"\n\t-------------------------
-----------------------------------------
\t";
            cout<<"ITEMS IN YOUR CART\n\n\t";
            viewcart();
        }
        cout<<"\n\n\t\5Keep shopping?(Y/N)";
        ch=getche();
        y++;
    }while(ch=='y'||ch=='Y');
    getch();
    xxx:int conf;
```

```
    cout<<"\n\n\tPress 1 to confirm order, Press
2 to discard cart\n\t";
    cin>>conf;
    if(conf==1)
        goto def;
    if(conf==2)
        mainmenu();
    if((conf!=1)||(conf!=2))
    {
        cout<<"\n\tPlease choose either 1 or 2";
        goto xxx;
    }
    def: clrscr();
    cout<<"\n\n**********************************
INVOICE**********************************\n";
    cout<<"\nPr No."<<setw(NAME)<<"Pr Name
"<<setw(QNTY)<<"Quantity"<<setw(PRICE)<<"Price
"<<setw(PRICE)<<"Amount   ";
    for(int x=0;x<=c;x++)
    {
        fp.open("Shop.dat",ios::in);
        fp.read((char*)&pr,sizeof(product));
        while(!fp.eof())
        {
            if(pr.retserialno()==order_arr[x])
            {

                amt=pr.retprice()*quan[x];

    cout<<endl<<order_arr[x]<<setw(NAME)<<pr.retn
ame()<<setw(QNTY)<<quan[x]<<setw(PRICE)<<pr.retpr
ice()<<setw(PRICE)<<amt;
                total+=amt;
            }
            fp.read((char*)&pr,sizeof(product));
        }
        fp.close();
```

```
    }
    xout.close();
    cout<<"\n\n\t\t\t\t\t\tTOTAL = "<<total;
    char qw,feedback[100],complaint[100];
    char ab;
    cout<<"\n\n\n\tGot a complaint or a feedback?
Tell us! (Y/N) ";
    qw=getche();
    if(qw=='y'||qw=='Y')
    {
        clrscr();
        cout<<"\n\t1.Feedback\n\t2.Complaint\n";
        cout<<"\t";
        ab=getche();
        switch(ab)
        {
            case '1': clrscr();
                ofstream
fout("fdbck.dat",ios::out|ios::app);
                cout<<"\n\n\n\n\n\n\tPlease enter
your feedback\n";
                cout<<"\t";
                gets(feedback);
                fout<<feedback<<endl;
                cout<<endl;
                cout<<"\n\tThank you for your
feedback! We look forward to serving you
soon.\n";
                fout.close();
                break;
            case '2': clrscr();
                ofstream
gout("cmplnt.dat",ios::out|ios::app);
                cout<<"\n\n\n\n\n\n\tPlease enter
your complaint\n";
                cout<<"\t";
                gets(complaint);
```

```
                gout<<complaint<<endl;
                cout<<endl;
                cout<<"\n\tOur customer care ex-
ecutive will get in touch with you shortly.\n";
                gout.close();
                break;
            default:cout<<"Please enter legal op-
tion\n";
                break;
        }
    }
    else
    {
        cout<<"\n\tThank you for shopping with
us! We look forward to serving you soon.\n";
        cout<<"\n\tItems in your order will be
delivered in 3-5 business days";
    }
    getch();
}
//function to display the cart to user
void viewcart()
{
    ifstream xin;
    char mn[100];
    xin.open("cart.txt");
    while(!xin.eof())
    {
        xin.getline(mn,100,'\n');
        cout<<"\t"<<mn;
        cout<<endl;
    }
    xin.close();
    getch();
}
//function to display feedback from user
void feedback()
```

```
{
    clrscr();
    char fd[100];
    cout<<"\n\n\n\n\n\n\tVIEWING CUSTOMER FEED-
BACK\n";
    ifstream fin("fdbck.dat",ios::in);
    cout<<"\n\t";
    while(!fin.eof())
    {
        fin.getline(fd,100,'\n');
        cout<<"\n\t"<<fd<<endl;
    }
    getch();
}
//function to display complaints from user
void complaint()
{
    clrscr();
    char cm[100];
    cout<<"\n\n\n\n\n\n\tVIEWING CUSTOMER COM-
PLAINTS\n";
    ifstream gin("cmplnt.dat",ios::in);
    cout<<"\n\t";
    while(!gin.eof())
    {
        gin.getline(cm,100,'\n');
        cout<<"\n\t"<<cm<<endl;
    }
    getch();
}
//INTRODUCTION FUNCTION
void intro()
{
    clrscr();
    /*int driver=DETECT,mode;
    initgraph(&driver,&mode,"c:\\tc\\bgi");
    setbkcolor(9);*/
```

```
    cout<<"\n\t\t       Welcome to MerxElectron-
ics!\n";
    cout<<"\n\n";
    cout<<"\t\t\4\4     \4\4  \4\4\4\4\4\4\4\4\4
\4\4\4\4\4\4\4\4   \4\4     \4\4\n";
    cout<<"\t\t\4\4\4   \4\4\4  \4\4        \4\4
\4\4   \4\4   \4\4\n";
    cout<<"\t\t\4\4\4\4 \4\4\4\4  \4\4
\4\4     \4\4    \4\4 \4\4\n";
    cout<<"\t\t\4\4 \4\4\4 \4\4  \4\4\4\4\4\4\4
\4\4\4\4\4\4\4\4      \4\4\4\n";
    cout<<"\t\t\4\4     \4\4  \4\4        \4\4
\4\4      \4\4 \4\4\n";
    cout<<"\t\t\4\4     \4\4  \4\4        \4\4
\4\4    \4\4   \4\4\n";
    cout<<"\t\t\4\4     \4\4  \4\4\4\4\4\4\4\4\4
\4\4     \4\4  \4\4     \4\4\n";
    cout<<"\t\t
\4\4\n";
    cout<<"\t\t
\4\4\n";
    cout<<"\n\t\t\t Online shopping, redefined.";
    cout<<"\n\n\n\n\tPress any key to start shop-
ping";
    //cout<<"\n\n\n\n\tABOUT MERX ELECTRON-
ICS\n\tESTD. 2015\n\tFOUNDERS-Salman Mohamed,BM
Amitraj,Abhishek Arul,Aman Rai\n";
    getch();
    //closegraph();
}
// ADMINSTRATOR MENU FUNCTION
void admin_menu()
{
    clrscr();
    cout<<"\n\n\tADMIN MENU";
    cout<<"\n\n\t1. Add product";
    cout<<"\n\n\t2. Display all products";
```

```
    cout<<"\n\n\t3. Display by product";
    cout<<"\n\n\t4. Modify product";
    cout<<"\n\n\t5. Delete product";
    cout<<"\n\n\t6. View product menu";
    cout<<"\n\n\t7. View customer feedback";
    cout<<"\n\n\t8. View complaints";
    cout<<"\n\n\t9. Sign out";
    cout<<"\n\n\t Please enter an option(1-9)";
    int ch2;
    cin>>ch2;
    switch(ch2)
    {
        case 1: clrscr();
                write_product();
                admin_menu();
                break;
        case 2: display_all();
                admin_menu();
                break;
        case 3: int num;
                clrscr();
                cout<<"\n\n\tEnter product ID to
display details\n\t";
                cin>>num;
                display_sp(num);
                admin_menu();
                break;
        case 4: modify_product();
                admin_menu();
                break;
        case 5: delete_product();
                admin_menu();
                break;
        case 6: menu();
                getch();
                admin_menu();
                break;
```

```
        case 7: feedback();
              admin_menu();
              break;
        case 8: complaint();
              admin_menu();
              break;
        case 9: break;
        default: //cout<<"\a";
            admin_menu();
    }
}
//function for customer details
void write_custsig()
{
    cs.open("Csig.dat",ios::out|ios::app|ios::bin
ary);
    o1.add();
    cs.write((char*)&o1,sizeof(cussig));
    cs.close();
}
//function for customer username
void write_username()
{
    char* u2;
    ofstream cl;
    cl.open("Cusr.txt",ios::out|ios::app);
    o2.username();
    u2=o2.retusername();
    cl<<u2<<endl;
    cl.close();
}
//function for customer password
void write_pass()
{
    char* p2;
    ofstream cm;
    cm.open("Cpss.txt",ios::out|ios::app);
```

```
    o3.pass();
    p2=o3.retpass();
    cm<<p2<<endl;
    cm.close();
}
//function for customer login
void custlogin()
{
    clrscr();
    char un[50],pw[50],ghi[50],jkl[50],x;
    int flag3=0,flag4=0;
    ifstream pin,rin;
    pin.open("cusr.txt");
    if(!pin)
    {
        cout<<"\n\n\n\n\n\n\n\tDeveloper error,
please try again later.";
        getch();
        exit(0);
    }
    cout<<"\n\n\n\n\n\n\tCUSTOMER LOGIN";
    cout<<"\n\n\tEnter username\n\t";
    cin>>un;
    while(!pin.eof())
    {
        pin.getline(ghi,50,'\n');
        if(strcmp(ghi,un)==0)
        {
            flag3=1;
            break;
        }
    }
    pin.close();
    int i=0;
    cout<<"\n\tEnter password\n\t";
    do{
        x=getch();
```

```
        if(x==0x0d)  //enter ascii
            break;
        if(x==0x08)  //bckspc ascii
        {
            if(i!=0)
                i--;
            cout<<"\b";
            cout<<" ";
            cout<<"\b";
            continue;
        }
        cout<<"*";
        pw[i]=x;
        i++;
    }while(1);
    pw[i]='\0';
    rin.open("cpss.txt");
    while(!rin.eof())
    {
        rin.getline(jkl,50,'\n');
        if(strcmp(jkl,pw)==0)
        {
            flag4=1;
            break;
        }
    }
    rin.close();
    if(flag3!=1||flag4!=1)
    {
        cout<<"\n\tINVALID CREDENTIALS";
        getch();
        mainmenu();
    }
    else
    {
        cout<<"\n\n\tLOGIN SUCCESSFUL!";
        cout<<"\n\n\tWelcome, "<<un<<"!";
```

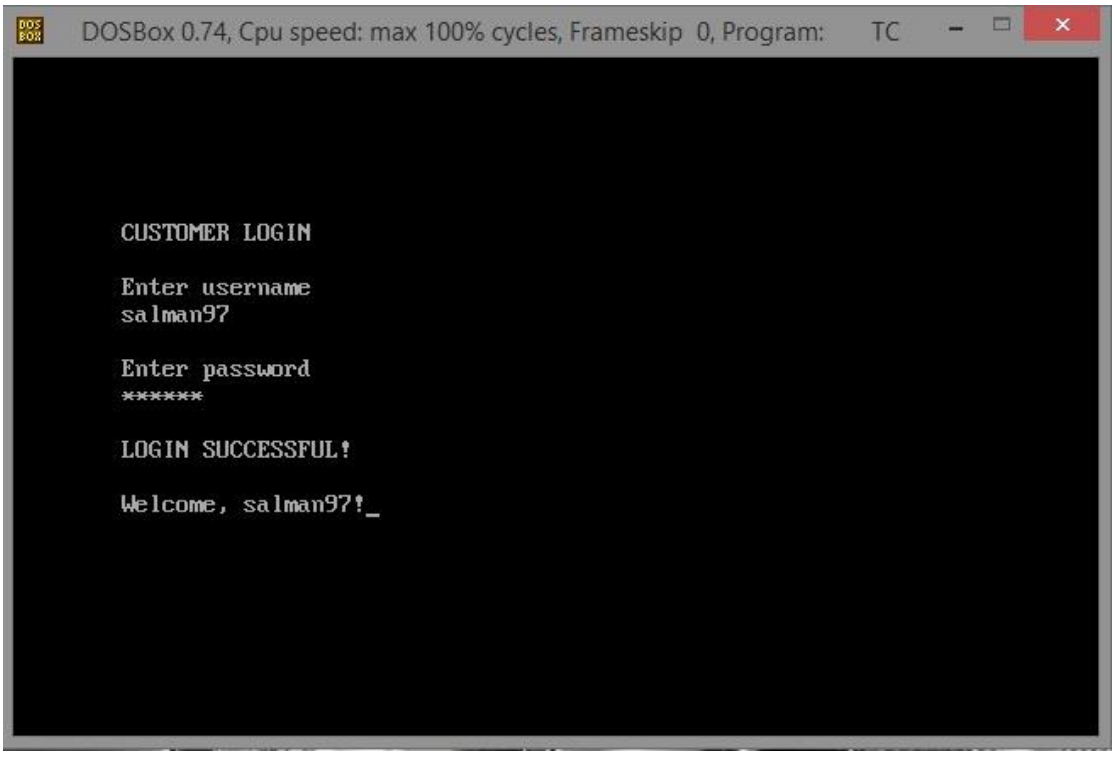
```
        getch();
    }
}
//function for employee login
void emplogin()
{
    clrscr();
    char eid[50],abc[50],epass[50],def[50],y;
    int flag1=0,flag2=0;
    cout<<"\n\n\n\n\n\n\tEMPLOYEE LOGIN";
    cout<<"\n\n\tEnter ID\n\t";
    cin>>eid;
    ifstream bin,tin;
    bin.open("emp.txt");
    while(!bin.eof())
    {
        bin.getline(abc,50,'\n');
        if(strcmp(abc,eid)==0)
        {
            flag1=1;
            break;
        }
    }
    bin.close();
    int j=0;
    cout<<"\n\tEnter password\n\t";
    do{
        y=getch();
        if(y==0x0d)  //enter ascii
            break;
        if(y==0x08)  //bckspc ascii
        {
            if(j!=0)
                j--;
            cout<<"\b";
            cout<<" ";
            cout<<"\b";
```

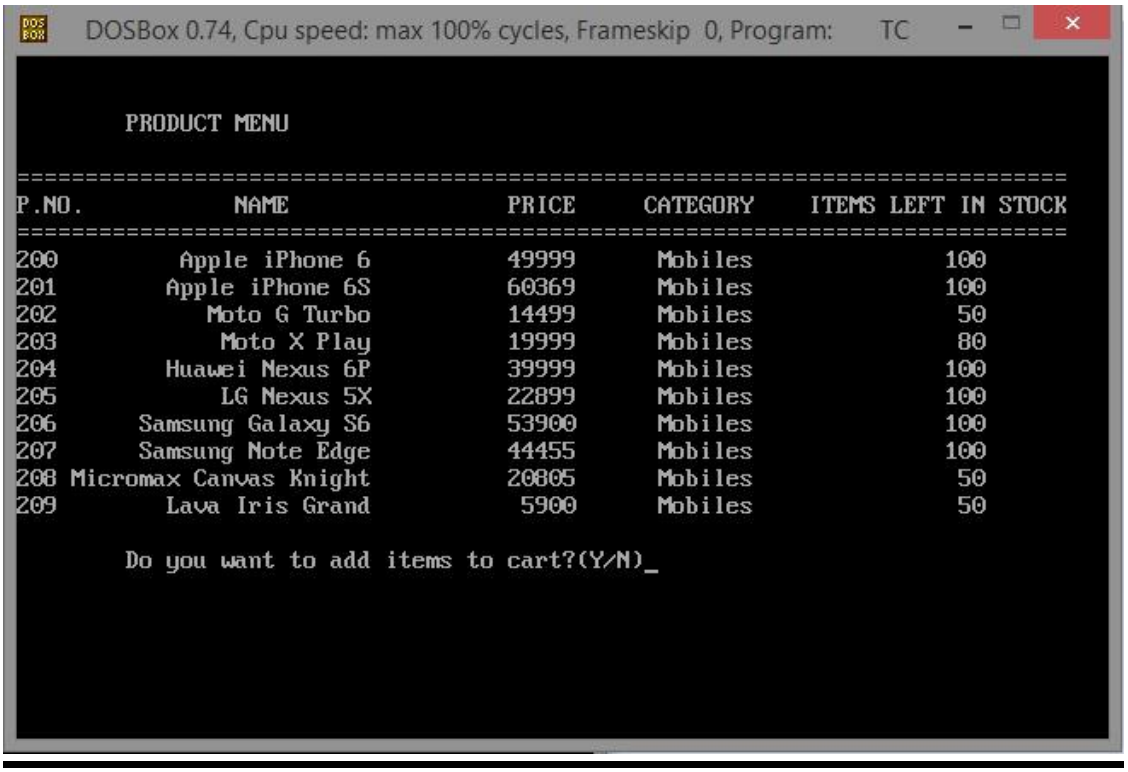
```
            continue;
        }
        cout<<"*";
        epass[j]=y;
        j++;
    }while(1);
    epass[j]='\0';
    tin.open("def.txt");
    while(!tin.eof())
    {
        tin.getline(def,50,'\n');
        if(strcmp(def,epass)==0)
        {
            flag2=1;
            break;
        }
    }
    tin.close();
    if(flag1!=1||flag2!=1)
    {
        cout<<"\n\tINVALID CREDENTIALS";
        getch();
        mainmenu();
    }
    else
    {
        cout<<"\n\tLOGIN SUCCESSFUL!";
        cout<<"\n\n\tWelcome, "<<eid<<"!";
        getch();
        admin_menu();
    }
}
//function for customer menu
void customer()
{
    char cc;
```

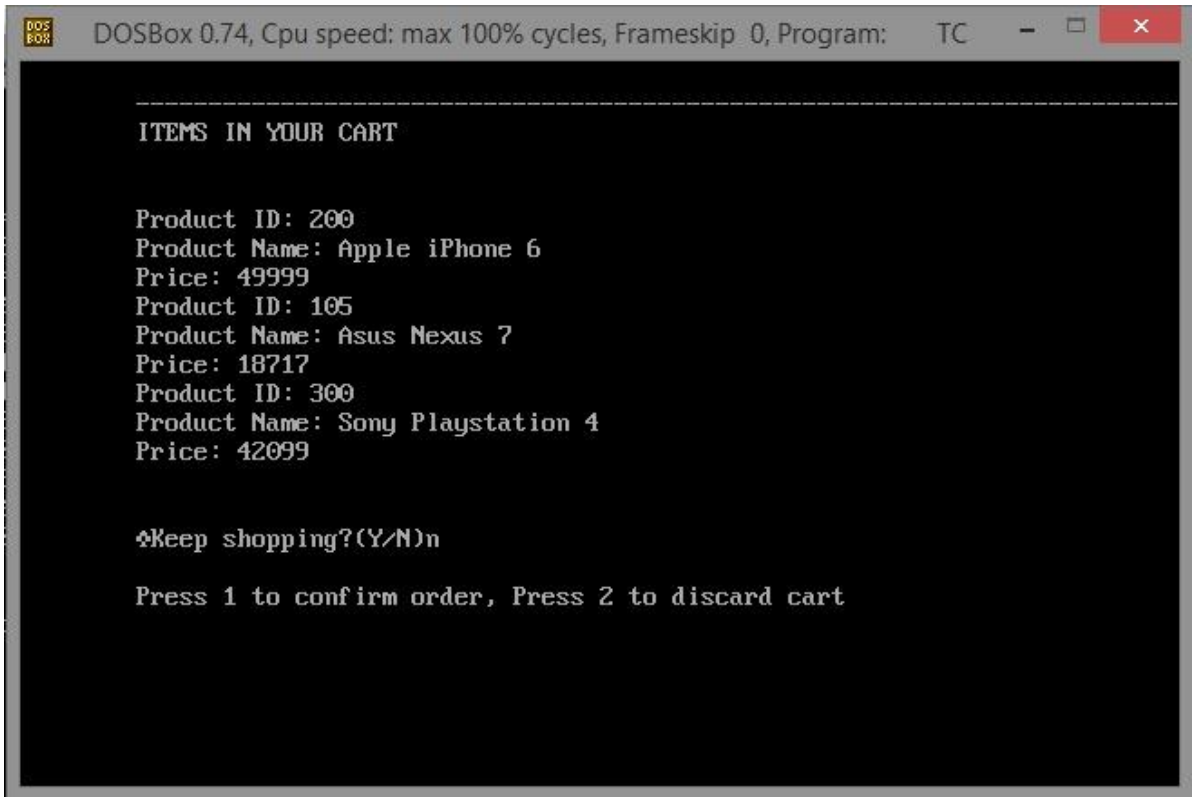
```
    cout<<"\n\n\n\n\n\n\n\n\n\t1.Login using ex-
isting account\n\t2.Don't have an account? Sign
up!\n\t3.Exit\n\t";
    cc=getche();
    switch(cc)
    {
        case '1': custlogin();
            break;
        case '2': write_custsig();
            write_username();
            write_pass();
            cout<<"\n\n\tYour account has been
created!";
            break;
        case '3': mainmenu();
            break;
        default:cout<<"Invalid choice";
            getch();
            mainmenu();
    }
}
void mainmenu()
{
    char ch,cart;
    do
    {
        clrscr();
        cout<<"\n\n\n\n\n\n\n\n\tMAIN MENU";
        cout<<"\n\n\t01. CUSTOMER LOGIN";
        cout<<"\n\n\t02. EMPLOYEE LOGIN";
        cout<<"\n\n\t03. EXIT";
        cout<<"\n\n\tPlease choose an option(1-
3)";
        ch=getche();
        switch(ch)
        {
            case '1': clrscr();
```

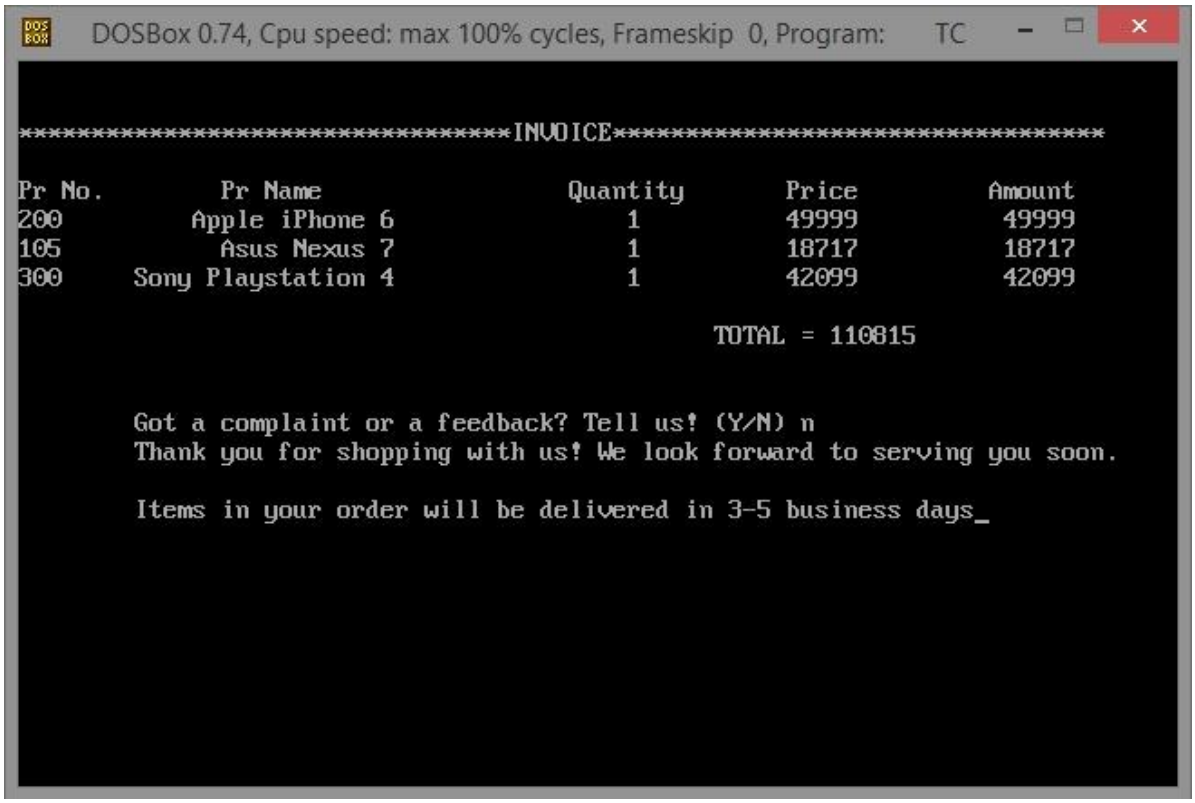
```
                    customer();
                    rer:menu();
                    cout<<"\n\tDo you want to add
items to cart?(Y/N)";
                    cart=getche();
                    if(cart=='y'||cart=='Y')
                    {
                      place_order();
                    }
                    else
                    {
                      goto rer;
                    }
                    break;
               case '2': emplogin();
                    break;
               case '3': exit(0);
               default : cout<<"Enter a valid
choice\n";
                    break;
          }
     }while(1);
}
// THE MAIN FUNCTION OF PROGRAM
int main()
{
     intro();
     mainmenu();
     return 0;
}
```

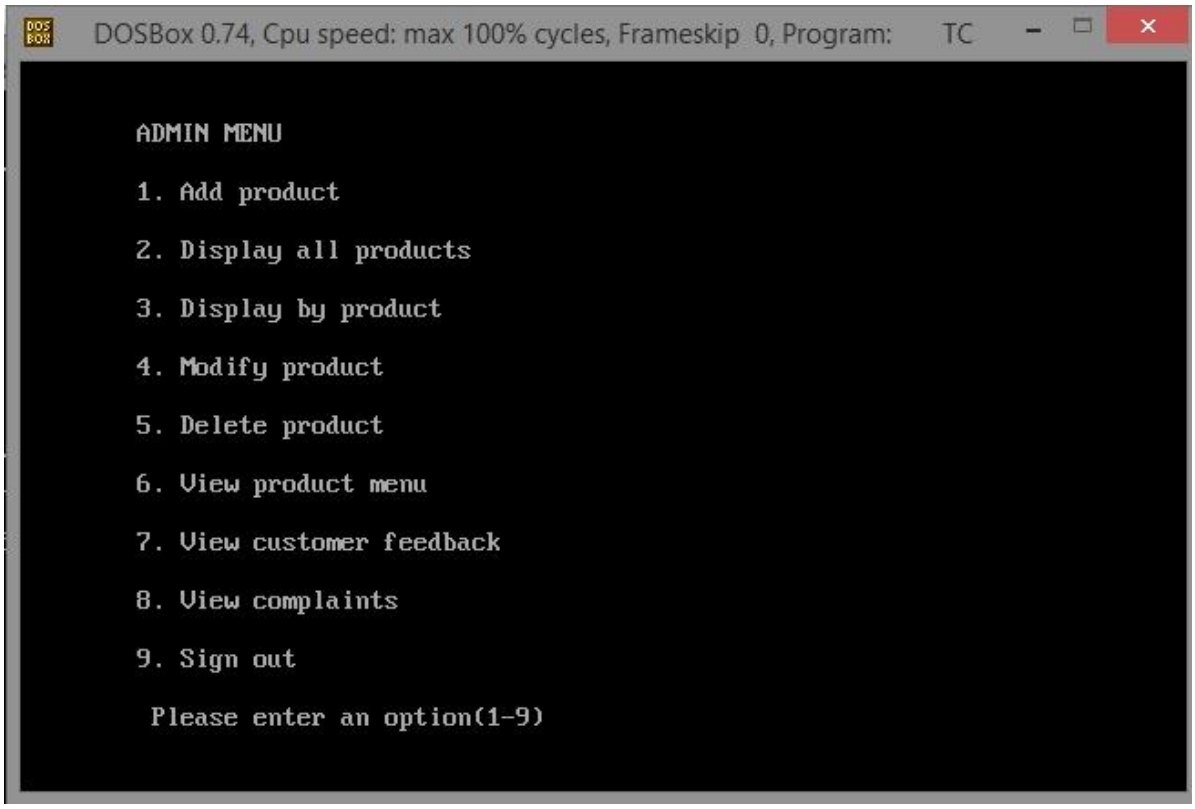
OUTPUT(S)

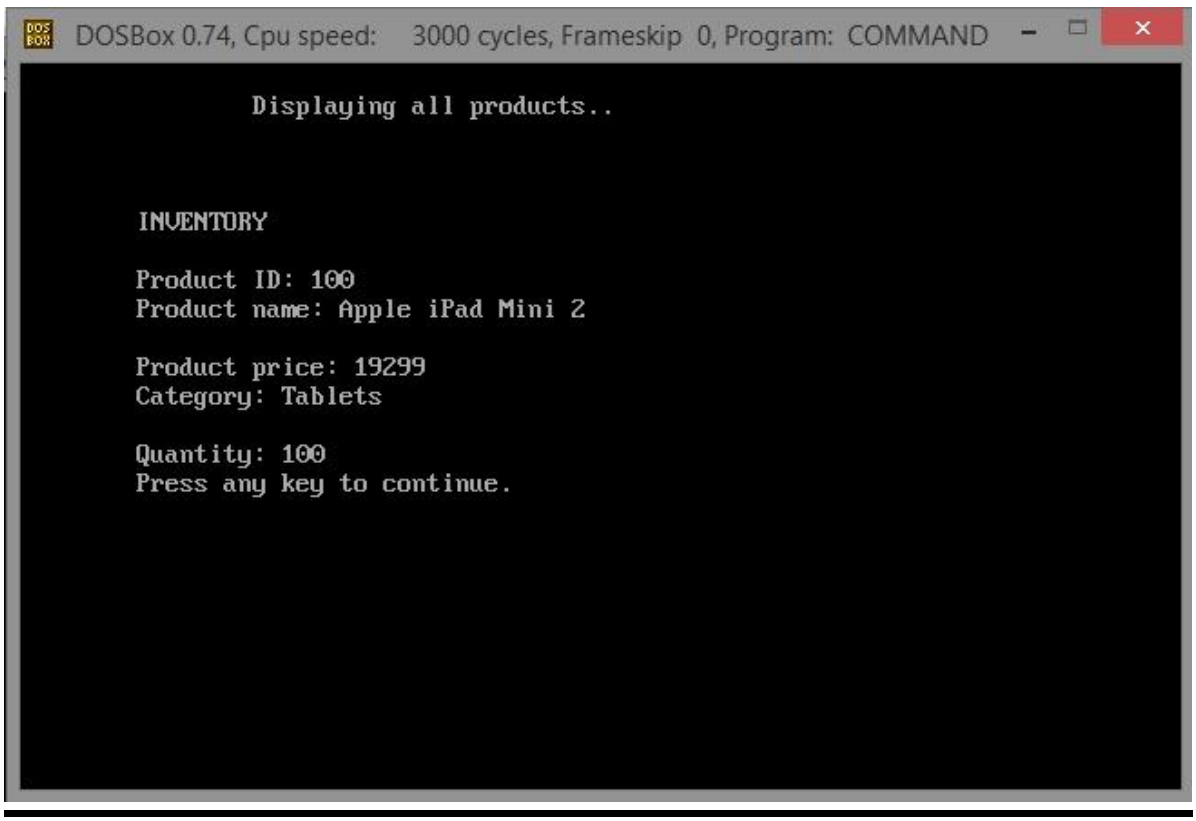

DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
CUSTOMER LOGIN
Enter username
salman97
Enter password
******
LOGIN SUCCESSFUL!
Welcome, salman97!_


DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
PRODUCT MENU
P.NO. NAME PRICE CATEGORY ITEMS LEFT IN STOCK
200 Apple iPhone 6 49999 Mobiles 100
201 Apple iPhone 6S 60369 Mobiles 100
202 Moto G Turbo 14499 Mobiles 50
203 Moto X Play 19999 Mobiles 80
204 Huawei Nexus 6P 39999 Mobiles 100
205 LG Nexus 5X 22899 Mobiles 100
206 Samsung Galaxy S6 53900 Mobiles 100
207 Samsung Note Edge 44455 Mobiles 100
208 Micromax Canvas Knight 20805 Mobiles 50
209 Lava Iris Grand 5900 Mobiles 50
Do you want to add items to cart?(Y/N)_


DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
ITEMS IN YOUR CART
Product ID: 200
Product Name: Apple iPhone 6
Price: 49999
Product ID: 105
Product Name: Asus Nexus 7
Price: 18717
Product ID: 300
Product Name: Sony Playstation 4
Price: 42099
Keep shopping?(Y/N)n
Press 1 to confirm order, Press 2 to discard cart


DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
INVOICE
Pr No. Pr Name Quantity Price Amount
200 Apple iPhone 6 1 49999 49999
105 Asus Nexus 7 1 18717 18717
300 Sony Playstation 4 1 42099 42099
TOTAL = 110815
Got a complaint or a feedback? Tell us! (Y/N) n
Thank you for shopping with us! We look forward to serving you soon.
Items in your order will be delivered in 3-5 business days_


DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
ADMIN MENU
1. Add product
2. Display all products
3. Display by product
4. Modify product
5. Delete product
6. View product menu
7. View customer feedback
8. View complaints
9. Sign out
Please enter an option(1-9)


DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: COMMAND
Displaying all products..
INVENTORY
Product ID: 100
Product name: Apple iPad Mini 2
Product price: 19299
Category: Tablets
Quantity: 100
Press any key to continue.

REQUIREMENTS

Hardware Specifications

Processor: Intel Pentium 4 or higher
Processor Speed: 250 MHz to 833MHz
RAM: 512MB or higher
Hard Disk: 1024 MBs or higher

Software Specifications

Software: Turbo C7 DOSBOX DOS Emulator
Language: C++
Operating System: Window 7 or higher

BIBLIOGRAPHY

We would like to give due credit to the following sources for helping us get through minute setbacks and obstacles.

- Computer Science with C++
  -Sumita Arora
- www.cplusplus.com
- www.stackoverflow.com