A MINI PROJECT

ON

“MOBILE STORE MANAGEMENT SYSTEM”

Submitted By:

Batch no: 4, 5, 6

B Prathyusha	(17121a1210)
C Karthik Naidu	(17121a1211)
C Sravani	(17121a1212)
C.Kranthi Kumar	(17121a1213)
C Naga Sai	(17121a1214)
D Sashi Kumar	(17121a1215)
D Venkateshwara chowdary	(17121a1216)
D Pushpanjali	(17121a1217)
D Sumedh	(17121a1218)

SREE VIDYANIKETHAN ENGINEERING COLLEGE

(AUTONOMOUS)

(Affliated by JNTU Anantapur, Approved by AICTE, Accredited by NBA,NAAC with ‘A’ grade)

Sree Sainath Nagar, A.Rangampet, Tirupari , AP - 517102

TABLE OF CONTENT

1. ABSTRACT

The objective of the Computerized Mobile Store Management System is to develop a software for monitoring and controlling the transactions of mobile phones in a Mobile Store. The project "Computerized Mobile Store Management System" is developed in java , and it is window based application.

Existing System:

The Existing system is manual. The Mobile details and bill details will be maintain in the books. The following are the drawbacks with the manual system.

- Delay in Searching a Mobile Phones in Mobile Store.
- Unable to maintain huge Mobile Phone details in the books.
- Unable to maintain the perfect Billing Details.
- Less security for Mobile phone details.

Proposed System:-

Proposed system is a computerized Mobile Store. Through our software user can add Mobile Phone details, search for Mobile Phone, Update Mobile Phone details, and Generate credit and normal bills.Our proposed system has following advantages.

- User friendly interface
- Fast access to database
- Search facility
- Look and feel environment

All manual difficulties in managing the Mobile store have been rectified by implementing computerization.

MODULE DESCRIPTION

The 'Computerized Mobile Store Management System' has four main Modules.

1.1 Mobile Phone Administration

It provides the user to maintain the Mobile Phone Details Like

i. Add a new Mobile Phone
ii. Search for a Mobile Phone
iii. Update a Mobile Phone Details
iv. Delete a Mobile Phone
v. View the mobile phones

Fig 1.1.1 mobile phone administration

2 INTRODUCTION

- The Mobile Store Management System is developed for desktop systems to facilitate mobile shop owners management of mobile details, which will include mobile phones .It can be used efficiently for physically separated shops in different locations. This software will provide in a simple and easy to operate user interface, which can be managed by any user without having any depth knowledge of the computer system.
- One can use this software to get a mobile's report. Administrators can add data, search data, view data, edit data and delete data.
- Therefore, creating an application with the basic requirement of low cost is essential for small organizations.
- Mobile shop management system is application software designed to take advantage of today's technology and reduce or avoid the burden of storing data on paper and in files.
- User-friendly prompts are provided for user.

3 MODULES

There are five modules in this project. They are

- Adding mobile record
- View mobile record
- Search mobile record
- Edit mobile record
- Delete mobile record

MODULE 3.1:

ADDING MOBILE RECORD

Objective:-

- The main objective is to add new mobile record.
- In this module it shows various mobile brands (such as Apple, Nokia, etc …..), so that it make easier to enter the data regarding to that brand.
- We give mobile id, Brand name, Quantity, Prize etc…

Operations:-

- Using getdata() reading mobile id, model name, quantity , availability status, price.

Concepts:-

- Files.
- Structures.
- Functions.

MODULE 3.2:-

VIEWING MOBILE RECORD

Objective:-

- Viewing the mobile records.

Operations:-

- Displaying the various categories of mobiles id, quantity ,model name, and price and availability status.

Concepts:-

- File concepts.
- Control statements.

MODULE 2.3:

SEARCH MOBILE RECORD

Objective:-

- In order to make the work easier to the user for viewing a particular mobile record , it can be done through the search module.
- We will be giving as our convenience we can search either by giving id , brand so that we can search easily.

Operations:

- Searching Based on mobile id.
- Searching based on model name.

Concepts:-

- File handling concepts
- control statement
- Strings concepts.

MODULE 2.4:

EDITING   MOBILE RECORD

Objective:-

- Corrections regarding to the mobiles can be done through the edit module.
- Again   we need to give mobile id, Brand name, Quantity etc…

Operations:-

- Reading the mobile id which wanted to edit and comparing mobile id with the existing id's . If found editing it.

Concepts:-

- File handling concepts
- Checkid() function.
- Return function

MODULE 2.5:-

DELETE MOBILE RECORD

Objective:-

- A mobile record is removed using this delete module.

Operations:-

- Reading the mobile id and comparing with the existing Id's, if it found deleting the record.

Concepts:-

- File concepts.
- Control statements.
- Relational operator.

SOURCE CODE:

```
#include<stdio.h>
#include <stdlib.h>
#include<string.h>
#include<ctype.h>
#include<time.h>
#define RETURNTIME 15
char catagories[][15]={"Samsung","Apple","Nokia","Sony","LG","HTC"};
void returnfunc(void);
 void mainmenu(void);
 void addstaff(void);
 void deletestaff(void);
 void editstaff(void);
 void searchstaff(void);
 void viewstaff(void);
 void closeapplication(void);
     int  getdata();
     int  checkid(int);
     int t(void);
     void Password();
     void issuerecord();
     void loaderanim();
     FILE *fp,*ft,*fs;
     int s;
     char findstaff;
     char password[10]={"pass"};
     struct meroDate
     {
                 int mm,dd,yy;
     };
     struct staff
     {
                 int id;
                 char stname[20];
                 char name[20];
                 char available[20];
                 int qty;
                 int price;
                 int count;
                 char *cat;
                 struct meroDate issued;
                 struct meroDate duedate;
     };
```

```
        struct staff a;

        int main()
        {
                Password();
                return 0;
        }
        void mainmenu()
        {
                int i;
                int choice;
                printf(" \t\tMAIN MENU \n");
                printf("\t\t\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\
                xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\
                xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\n");
                printf("<1> Add Mobile Record\n");
                printf("<2> View Mobile list\n");
                printf("<3> Search Mobile Record\n");
                printf("<4> Edit Mobile Record\n");
                printf("<5> Delete Mobile Record\n");
                printf("<6> Close The Application\n");
                t();
                printf("Enter your choice:");
                fflush(stdin);
                scanf("%d",&choice);
                switch(choice)
                {
                case 1:
                        addstaff();
                        break;
                case 2:
                        viewstaff();
                        break;
                case 3:
                        searchstaff();
                        break;
                case 4:
                        editstaff();
                        break;
                case 5:
                        deletestaff();
                        break;
                case 6:
```

```
        {
            printf("\tMobile Store Management System\n");
            printf("\tProject in C");
            printf("is brought to you by\n");
            printf("\tBATCH-2\n");
            exit(0);
        }
        default:
            {
            printf("\n\aWrong Entry!!Please re-entered correct option");
            mainmenu();
            }
        }
    }
void addstaff(void)
    {
        int i;
        int temp;
        printf("SELECT BRANDS\n");
        printf("<1> Samsung\n");
        printf("<2> Apple\n");
        printf("<3> Nokia\n");
        printf("<4> Sony\n");
        printf("<5> LG\n");
        printf("<6> HTC\n");
        printf("<7> Back to main menu\n");
        printf("Enter your choice:");
        scanf("%d",&s);
        if(s==7)
            mainmenu() ;
        fp=fopen("stf.dat","ab+");
        if(getdata()==1)
        {
            a.cat=catagories[s-1];
            fseek(fp,0,SEEK_END);
            fwrite(&a,sizeof(a),1,fp);
            fclose(fp);
            printf("The record is sucessfully saved\n");
            printf("Save any more?(1-y / 0-n):");
            fflush(stdin);
            scanf("%d",&temp);
            if(temp==0)
                mainmenu();
            else
                addstaff();
        }
    }
    void deletestaff()
    {
        int d,b;
        int another=1;
        while(another==1)
        {
            findstaff='f';
            printf("Enter the Mobile ID to  delete:");
            scanf("%d",&d);
            fp=fopen("stf.dat","rb+");
            rewind(fp);
        while(fread(&a,sizeof(a),1,fp)==1)
        {
            if(a.id==d)
            {
                printf("\nThe Mobile record is available");
                printf("\nStaff name is %s",a.name);
                findstaff='t';
            }
        }
        if(findstaff!='t')
        {
            printf("\nNo record is found modify the search\n");
            mainmenu();
        }
    if(findstaff=='t' )
    {
        printf("\nDo you want to delete it?(1-Y/0-N):");
        fflush(stdin);
        scanf("%d",&b);
        if(b==1)
        {
            ft=fopen("test.dat","wb+");
            rewind(fp);
            while(fread(&a,sizeof(a),1,fp)==1)
                {
                if(a.id!=d)
                {
                    fseek(ft,0,SEEK_CUR);
                    fwrite(&a,sizeof(a),1,ft);
                }
            }
            fclose(ft);
            fclose(fp);
            remove("stf.dat");
            rename("test.dat","stf.dat");
            fp=fopen("stf.dat","rb+");
            if(findstaff=='t')
            {
                printf("\nThe record is sucessfully deleted\n");
                printf("\n\tDelete another record?(Y/N)");
            }
            fclose(fp);
        }
        else
            mainmenu();
            fflush(stdin);
            printf("\n\tDelete another record?(Y/N)");
            scanf("%d",&another);
            }
        }
        mainmenu();
    }
    void searchstaff()
    {
        int d;
        int c,e;
        printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2xB2\Xb
    2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2Search
    Mobile\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\
    xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");
        printf("\n1. Search By ID\n");
        printf("2. Search By model Name\n");
        printf("Enter Your Choice:");
        fflush(stdin);
        scanf("%d",&c);
        fp=fopen("stf.dat","rb+");
        rewind(fp);
        switch(c)
        {
        case 1:
            {
```

```
        printf("\xB2\xB2\xB2\xB2\xB2\xB2Search Mobile By
        Id\xB2\xB2\xB2\xB2\xB2\xB2");
        printf("Enter the Mobile id:");
        scanf("%d",&d);
        while(fread(&a,sizeof(a),1,fp)==1)
        {
        if(a.id==d)
        {
        printf("The Mobile is available\n");
        printf("\nID:%d",a.id);
        printf("\nBrand:%s",a.cat);
        printf("\nName:%s",a.name);
        printf("\nAvailability:%s ",a.available);
        printf("\nQuantity:%i ",a.qty);
        printf("Price:%i ",a.price);
        findstaff='t';
        }
    }
    If(findstaff!='t')
    {
        printf("\aNo Record Found\n");
    }
    fclose(fp);
    printf("Try another search?(1-y/0-n)");
    fflush(stdin);
    scanf("%d",&e);
    if(e==1)
        searchstaff();
    else
        mainmenu();
    break;
    }
case 2:
    {
    char s[15];
    printf("\xB2\xB2\xB2\xB2\xB2\xB2Search Mobile By model
Name\xB2\xB2\xB2\xB2\xB2\xB2");
    printf("\nEnter Model Name:");
    scanf("%s",s);
    int d=0;
    while(fread(&a,sizeof(a),1,fp)==1)
    {
    if(strcmp(a.name,(s))==0)
```

```
                {
                        printf("\nThe Staff is available");
                        printf("\nID:%d",a.id);
                        printf("\nName:%s",a.name);
                        printf("\nAvailability:%s",a.available);
                        printf("\nQuantity:%i",a.qty);
                        printf("\nPrice:%i",a.price);
                        d+=6;
                }
                }
                if(d==0)
                        printf("\n\aNo Record Found");
                        printf("\nTry another search?(1-Y/0-N)");
                        fflush(stdin);
                        scanf("%d",&e);
                if(e==1)
                        searchstaff();

                else
                        mainmenu();
                break;
                }
        default :
                searchstaff();
    }
        fclose(fp);
}
void viewstaff(void)
{
        int i=0,j;
        printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\
        xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2Mobile
        List\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB
        2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");
        printf("\n BRAND\t\tID\tMODEL
        NAME\tAVAILABILITY\tQUANTITY\tPRICE\n");
        j=4;
        fp=fopen("stf.dat","rb");
        while(fread(&a,sizeof(a),1,fp)==1)
        {
                printf("%s",a.cat);
                printf("\t\t%d",a.id);
                printf("\t%s",a.name);
                printf("\t\t%s",a.available);
                printf("\t%i",a.qty);
                printf("\t\t%i",a.price);
                printf("\n\n");
                j++;
        }
        fclose(fp);
        returnfunc();
}
void editstaff(void)
{
        int c=0;
        int d,e;
        printf("\n\xB2\xB2\xB2\xB2\xB2\xB2Edit Mobile Section
        \xB2\xB2\xB2\xB2\xB2\xB2");
        int another=1;
        while(another==1)
        {
                printf("\nEnter Mobile Id to be edited:");
                scanf("%d",&d);
                fp=fopen("stf.dat","rb+");
            while(fread(&a,sizeof(a),1,fp)==1)
             {
                if(checkid(d)==0)
                {
                    printf("\nThe Mobile is availble");
                    printf("\nThe Mobile ID:%d",a.id);
                    printf("\nEnter new name:");
                   scanf("%s",a.name);
                   printf("\nEnter new Availability Status:");
                   scanf("%s",a.available);
                   printf("\nEnter new Quantity:");
                   scanf("%i",&a.qty);
                   printf("\nEnter new Price:");
                   scanf("%i",&a.price);
                  printf("\nThe record is modified");
                  fseek(fp,ftell(fp)-sizeof(a),0);
                  fwrite(&a,sizeof(a),1,fp);
                  fclose(fp);
                c=1;
              }
                if(c==0)
                {
                    printf("\nNo record found");
                 }
         }
           printf("\nModify another Record?(1-Y/0-N)");
           fflush(stdin);
           scanf("%d",&another);
           //another=getch() ;
         }
          returnfunc();
    }
void returnfunc(void)
{
        mainmenu();
 }
int getdata()
{
        int t;
        printf("\nEnter the Information Below");
        printf("\nBrand:");
        printf("%s",catagories[s-1]);
        printf("\nMBL ID:\t");
        scanf("%d",&t);
        if(checkid(t) == 0)
        {
            printf("\n\aThe id already exists\a\n\n");
            mainmenu();
            return 0;
        }
        a.id=t;
        printf("\nModel Name:");
        scanf("%s",a.name);
        printf("\nAvailability:");
        scanf("%s",a.available);
        printf("\nQuantity:");
        scanf("%i",&a.qty);
        printf("\nPrice:");
        scanf("%i",&a.price);
        return 1;
}
int checkid(int t)
{
         rewind(fp);
        while(fread(&a,sizeof(a),1,fp)==1)
```

```
      if(a.id==t)
    return 0;
    return 1;
}



int t(void)
{
    time_t t;
    time(&t);
    printf("\nDate and time:%s\n",ctime(&t));
    return 0 ;
}
void Password(void)
{
    char d[25]="Password Protected";
    char ch,pass[10];
    int i=0,j;
        printf("\t\t\t\tWELCOME\n\t\t\t\t  To \n\t\t   \xB2\xB2\xB2\xB2\xB2\xB2
    Mobile Store Management System \xB2\xB2\xB2\xB2\xB2\xB2\n");
        printf("\t \n\n\n Enter Password:");
       scanf("%s",&pass);
    /*while(ch!=13)
    {
        ch=getch();
        if(ch!=13 && ch!=8)
        {
            putch('*');
            pass[i] = ch;
            i++;
        }
    }
    pass[i] = '\0';   */


    if(strcmp(pass,password)==0)
    {
        printf("\n\n\n\t\tPassword matched!!");
        printf("\n\n\tPress any key to countinue.....");
        mainmenu();
    }
    else
```

```
        {
                printf("\n\n\n\t\t\aWarning!! \n\t   Incorrect Password");
                Password()
        }
```

SYSTEM REQUIREMENTS

Hardware   Requirements

PROCESSESOR	:	intel i3 or more 64-bit operating system
RAM	:	256 MB or More

Software   Requirements

Operating System	:	Ubuntu 16.04 LTS or Any linux OS
Language used	:	C

RESULT:

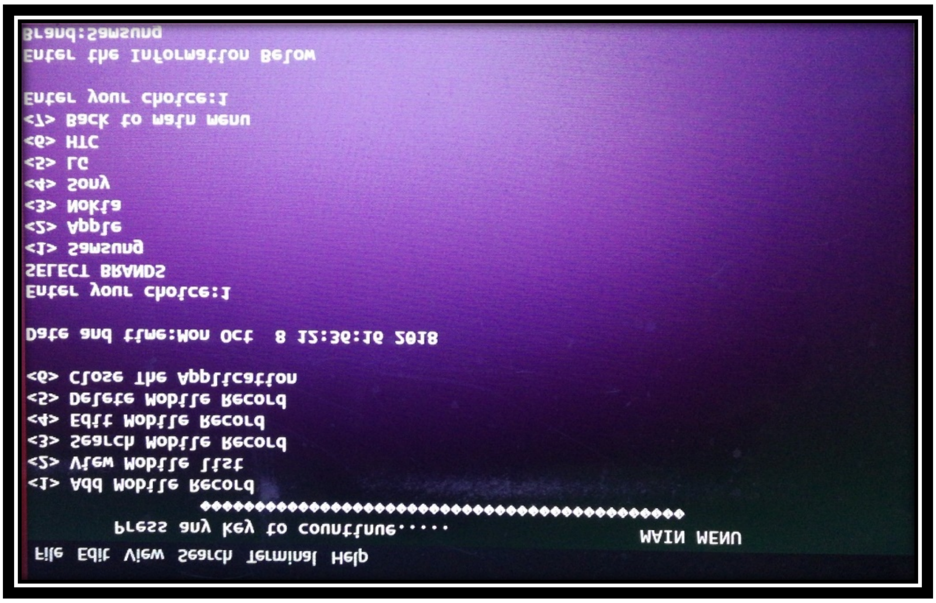
- This system is very simple in design and to implement . This system requires very low system resources.

  1 .security of data

  2 .user friendly and interactive

  3. minimum time required

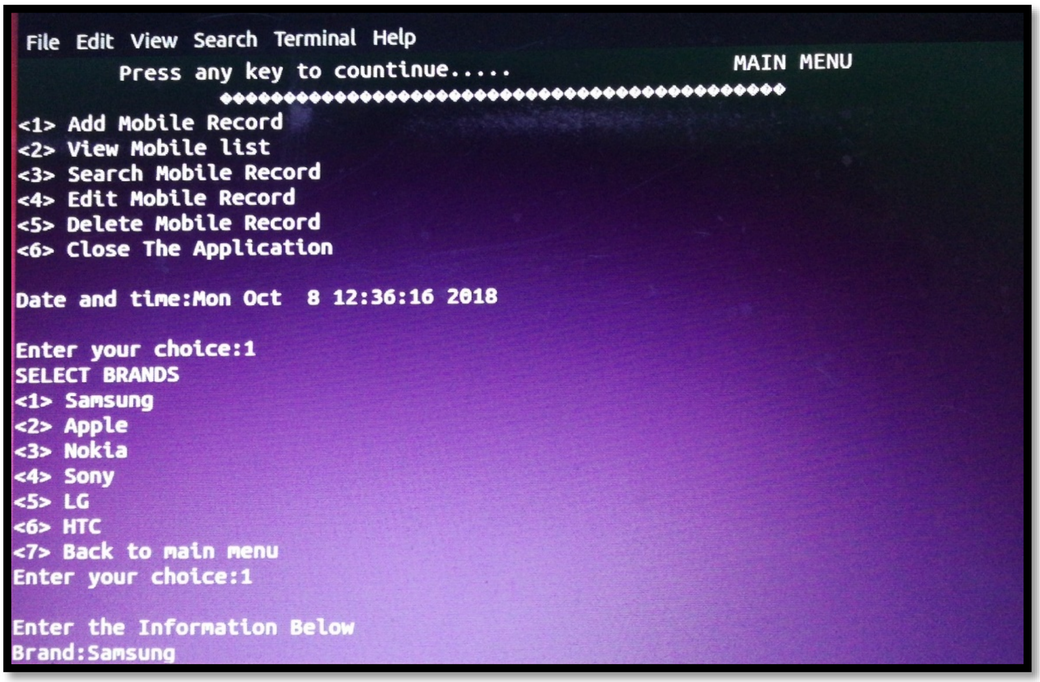
  - This project main intention is to be more user friendly to the users who doesn’t have prior knowledge to this application.



```
        MAIN MENU
    press any key to continue.....
    **********************************************
<1> Add Mobile Record
<2> View Mobile list
<3> Search Mobile Record
<4> Edit Mobile Record
<5> Delete Mobile Record
<6> Close The Application

Date and time:Mon Oct  8 12:36:16 2018

Enter your choice:1
SELECT BRANDS
<1> Samsung
<2> Apple
<3> Nokia
<4> Sony
<5> LG
<6> HTC
<7> Back to main menu
Enter your choice:1

Enter the Information Below
Brand:Samsung
```

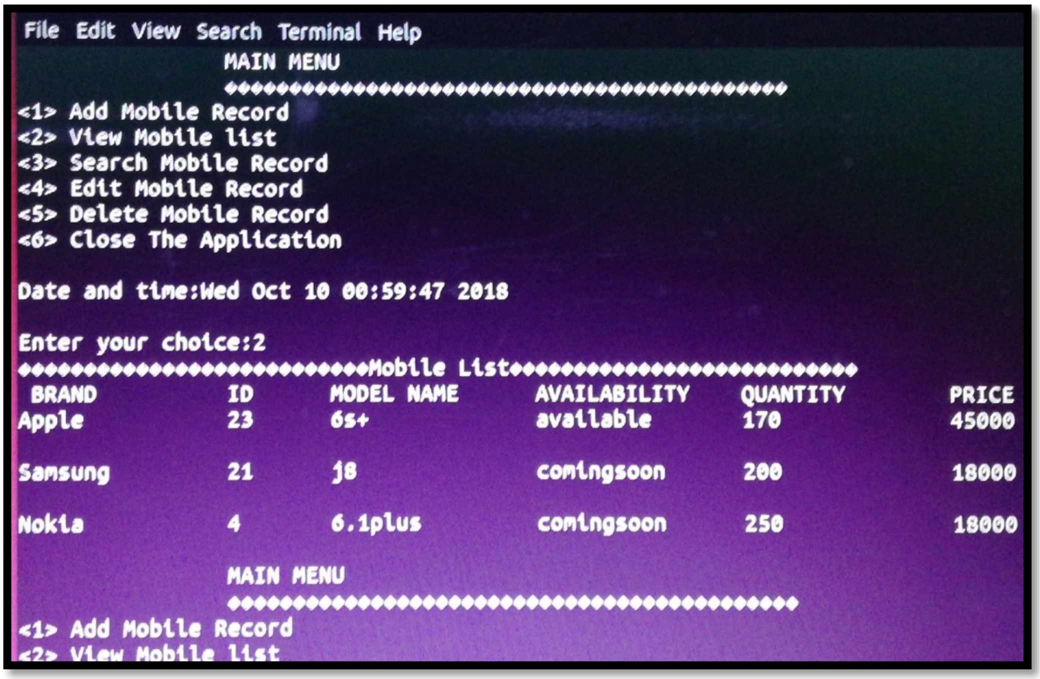
- Easy handling of huge amount of data regarding mobiles.
- Though the is unaware of computer system it may be friendly to everyone.

Fig 2.1.1 adding mobile record



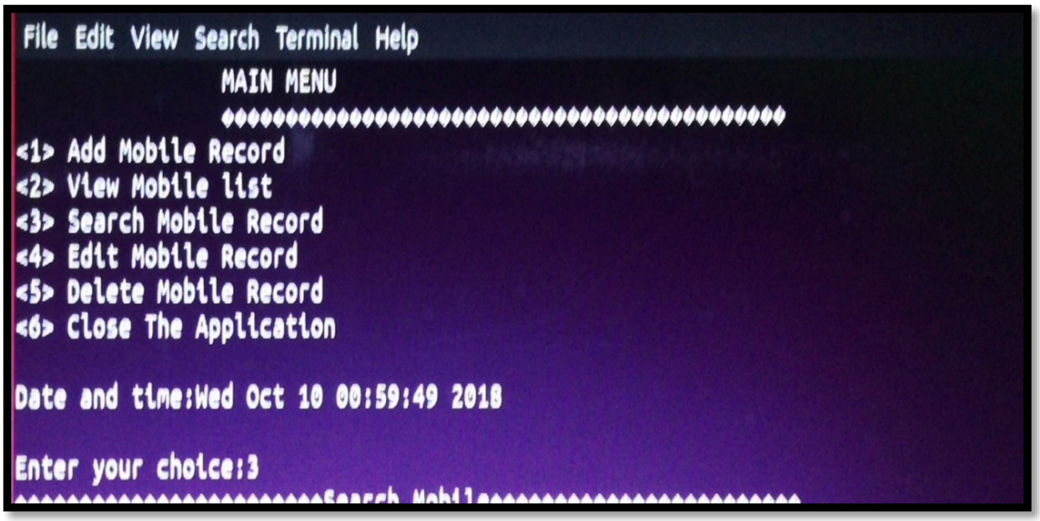
```
        Press any key to countinue.....                   MAIN MENU
                ◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
<1> Add Mobile Record
<2> View Mobile list
<3> Search Mobile Record
<4> Edit Mobile Record
<5> Delete Mobile Record
<6> Close The Application

Date and time:Mon Oct  8 12:36:16 2018

Enter your choice:1
SELECT BRANDS
<1> Samsung
<2> Apple
<3> Nokia
<4> Sony
<5> LG
<6> HTC
<7> Back to main menu
Enter your choice:1

Enter the Information Below
Brand:Samsung
```

Fig 2.1.2   adding mobile record



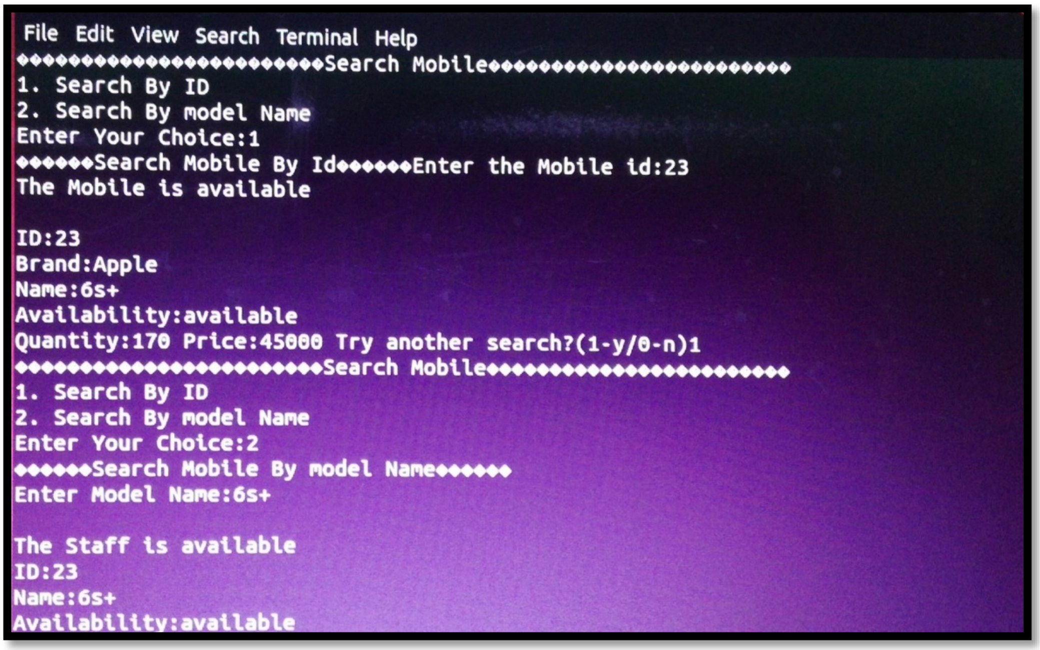
```
                MAIN MENU
                ♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦
<1> Add Mobile Record
<2> View Mobile list
<3> Search Mobile Record
<4> Edit Mobile Record
<5> Delete Mobile Record
<6> Close The Application

Date and time:Wed Oct 10 00:59:47 2018

Enter your choice:2
♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦Mobile List♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦
 BRAND          ID      MODEL NAME      AVAILABILITY    QUANTITY        PRICE
Apple           23      6s+             available       170             45000

Samsung         21      j8              comingsoon      200             18000

Nokia           4       6.1plus         comingsoon      250             18000

                MAIN MENU
                ♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦
<1> Add Mobile Record
<2> View Mobile list
```

Fig 2.2.1 viewing mobile record



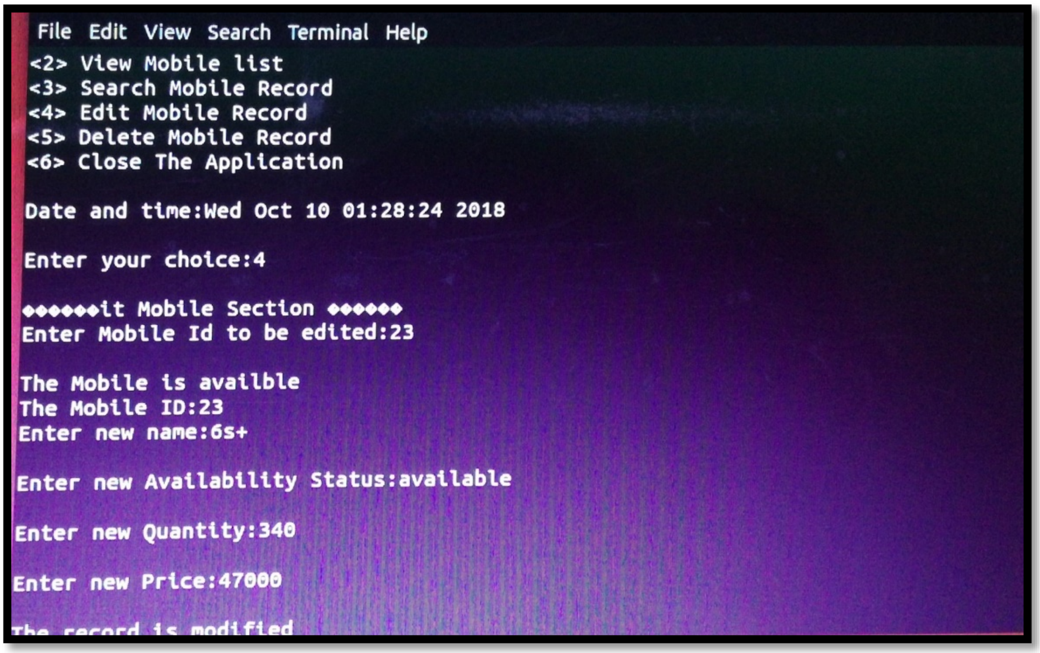
```
                MAIN MENU
                ♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦
<1> Add Mobile Record
<2> View Mobile list
<3> Search Mobile Record
<4> Edit Mobile Record
<5> Delete Mobile Record
<6> Close The Application

Date and time:Wed Oct 10 00:59:49 2018

Enter your choice:3
♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦Search Mobile♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦♦
```

Fig 2.3.1 searching mobile record



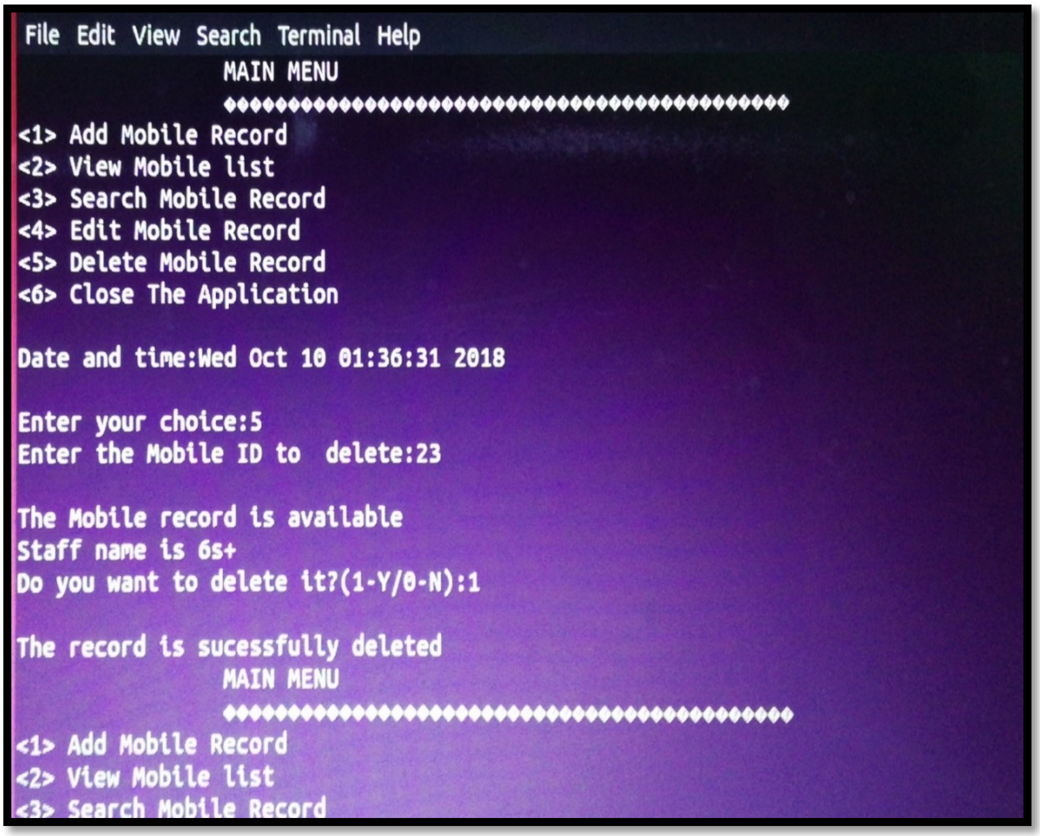
```
◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆Search Mobile◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
1. Search By ID
2. Search By model Name
Enter Your Choice:1
◆◆◆◆◆◆Search Mobile By Id◆◆◆◆◆◆Enter the Mobile id:23
The Mobile is available

ID:23
Brand:Apple
Name:6s+
Availability:available
Quantity:170 Price:45000 Try another search?(1-y/0-n)1
◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆Search Mobile◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
1. Search By ID
2. Search By model Name
Enter Your Choice:2
◆◆◆◆◆◆Search Mobile By model Name◆◆◆◆◆◆
Enter Model Name:6s+

The Staff is available
ID:23
Name:6s+
Availability:available
```

Fig 2.3.2 searching mobile record



```
<2> View Mobile list
<3> Search Mobile Record
<4> Edit Mobile Record
<5> Delete Mobile Record
<6> Close The Application

Date and time:Wed Oct 10 01:28:24 2018

Enter your choice:4

◆◆◆◆◆◆◆it Mobile Section ◆◆◆◆◆◆◆
Enter Mobile Id to be edited:23

The Mobile is availble
The Mobile ID:23
Enter new name:6s+

Enter new Availablity Status:available

Enter new Quantity:340

Enter new Price:47000

The record is modified
```

Fig 2.4.1 Editing mobile record



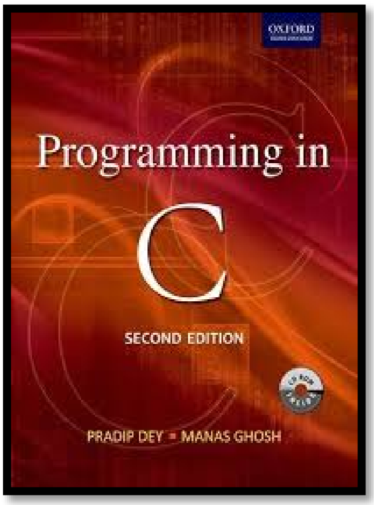
```
                MAIN MENU
                ◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
<1> Add Mobile Record
<2> View Mobile list
<3> Search Mobile Record
<4> Edit Mobile Record
<5> Delete Mobile Record
<6> Close The Application

Date and time:Wed Oct 10 01:36:31 2018

Enter your choice:5
Enter the Mobile ID to  delete:23

The Mobile record is available
Staff name is 6s+
Do you want to delete it?(1-Y/0-N):1

The record is sucessfully deleted
                MAIN MENU
                ◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
<1> Add Mobile Record
<2> View Mobile list
<3> Search Mobile Record
```

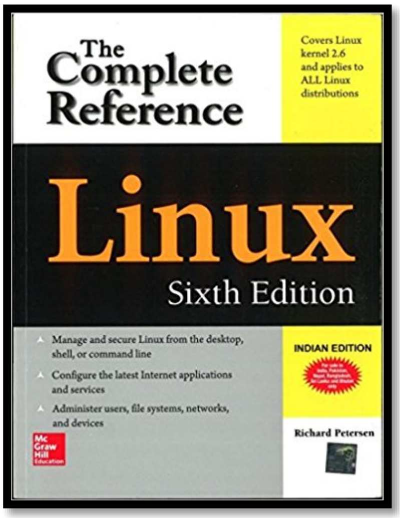
Fig:2.5.1.Deleting mobile record

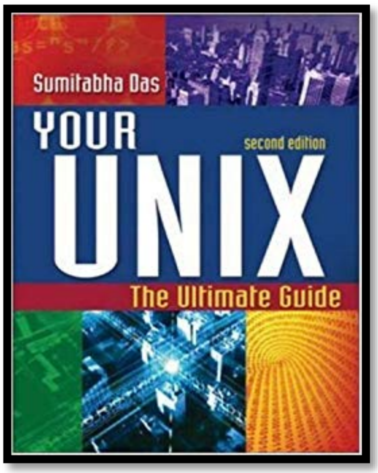
REFERENCES:-

TEXTBOOK:-

- SUMITABHA DAS,YOUR UNIX:THE ULTIMATE GUIDE,TATA McGraw - HILL.SIXTH EDITION,2007
- PRADIP DEY AND MANAS GHOSH,PROGRAMING IN C,SECOND EDITION,OXFORD UNIVERSITY PRESS,2011.
- RICHARD PETERSEN,LINUX:THE COMPLETE REFERENCE,TATA McGRAW-HILL,SIXTH-EDITION,2007.




The Complete Reference
Covers Linux kernel 2.6 and applies to ALL Linux distributions
Linux
Sixth Edition
Manage and secure Linux from the desktop, shell, or command line
Configure the latest Internet applications and services
Administer users, file systems, networks, and devices
INDIAN EDITION
Mc Graw Hill Education
Richard Petersen



CONCLUSION:

From this project we conclude that this mobile store management application is very useful for the users where they can store the data regarding mobiles comfortably Without requirement of in-depth knowledge

- This software has been made keeping it in mind that software is made for end users, not for programmers. So user friendliness keeping eye candy was first in the list. Wish this mobile store management system will serve satisfactory for which it is made. Easiness of mastering the whole process
- With the use of linux terminal the process of controlling the whole system would be much easier and those can keep full control on the overall process without facing much trouble . With the user friendly GUI the software is very

much user friendly and would be easy to use for anybody with a meager computer knowledge.