**PURBANCHAL UNIVERSITY**

**Biratnagar Nepal**

****

A Project report on

**“CONTACT MANAGEMENT SYSTEM”**

In the partial fulfillment for the requirement of the 2nd Semester Project-II (subject code- BIT 179CO) in the completion of **Bachelor of Information Technology (BIT)** degree at **KIST college** **of Information Technology**, under **Purbanchal University.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Submitted By:-** | | |  | **Submitted To:-** | |
| **1)** |  | **Anup Shrestha** |  |  |  |  |
| **2)** |  | **Rahul Neupane** | | **Purbanchal University** | | |

1. **Rojan Pant**
2. **Subash Danuwar**

**Under The Guidance of**

**Mr. Prawesh Dhungana**

**Lecturer, BIT**

**KIST COLLEGE OF INFORMATION AND TECHNOLOGY KAMALPOKHARI, KATHMANDU NEPAL**

**KIST COLLEGE OF INFORMATION AND TECHNOLOGY KAMALPOKHARI, KATHMANDU NEPAL**

****

**CERTIFICATE**

This is to certify that the project work entitled **“CONTACT MANAGEMENT SYTEM”** is carried out by **ANUP SHRESTHA (4712), RAHUL NEUPANE (4725), ROJAN PANT (4730),** **SUBASH DANUWAR (4715),** bona fide students of **KIST COLLEGE OF INFORMATION AND TECHNOLOGY** in partial fulfillment for the award of **BACHELOR IN INFORMATION AND TECHNOLOGY** of the **PURBANCHAL UNIVERSITY, BIRATNAGAR NEPAL**, during the year **2015-2016**. It is certified that all corrections indicatedfor internal assessment have been incorporated in the report submitted in the department library. The project report has been approved, as it satisfied the academic requirements in respect of the project work prescribed for the said degree.

The details of the students are as follows:-

|  |  |  |
| --- | --- | --- |
| **NAME** | **REGISTRATION NO.** | **SYMBOL NO.** |
| Anup Shrestha | 058-3-2-02784-2019 | 325126 |
| Rahul Neupane | 058-3-2-02801-2019 | 325143 |
| Rojan Pant | 058-3-2-02804-2019 | 325146 |
| Subash Danuwar | 058-3-2-02813-2019 | 325155 |

Course Semester: - 2nd Semester

Subject: - Project-II

Subject Code: - BIT (179CO)



Mr. Deepak Khadka

Program Coordinator, BIT



**KIST COLLEGE OF INFORMATION AND TECHNOLOGY KAMALPOKHARI, KATHMANDU**

**Examiner’s Certification**

The Project Report

On

**“CONTACT MANAGEMENT SYSTEM”**

**Developed by**

**Anup Shrestha**

**Rahul Neupane**

**Rojan Pant**

**Subash Danuwar**

Is approved and is acceptable in qualify form.

**Internal Examiner**

Name:

Designation:

**External Examiner**

Name:

Designation:

**ACKNOWLEDGEMENT**

It is with greatest satisfaction and euphoria that we are submitting our project report entitled **“CONTACT MANAGEMENT SYSTEM”.** We have completed it as a part of the curriculum of **PURBANCHAL UNIVERSITY.**

We also take this opportunity to express a deep sense of gratefulness to our **BIT Coordinator Mr.** **Deepak Khadka** and **BIT Lecturer Mr. Prawesh Dhungana** for their amiable support, valuableinformation and guidance which helped us in completing this task throughout its various stages. We are indebted to all members of **KIST College,** for the valuable support and suggestion provided by them using their specific fields’ knowledge. We are grateful for their cooperation during the period of our project.

Finally we would also like to express our gratefulness towards **Purbanchal University** for designing such a wonderful course structure. It will help us to get more knowledge in the field of Information Technology & help us to have a bright future in the field of technology.

We hope our university will accept this attempt as a successful project.

Last but not the least, our sincere thanks to our parents, teaching and non-teaching staffs of our college and also my friends.

**ANUP SHRESTHA (325126)**

**RAHUL NEUPANE (325143)**

**ROJAN PANT (325146)**

**SUBASH DANUWAR (325155)**

**STUDENT’S DECLARATION**

We hereby declare that the project report entitled “**CONTACT MANAGEMENT SYTEM**” is a result of our own work. If we are found guilty of copying any other report or published information and showing as our original work, we understand that we shall be liable and punishable by **Purbanchal University**.

We further certify that this Project submitted in partial fulfillment of the requirement for the award of Bachelor in Information Technology (**BIT**) of the **Purbanchal University** is our original work and has not been submitted for award of any other degree or other similar title or prize.

|  |  |  |  |
| --- | --- | --- | --- |
| S.N. | Name | Registration No. | Symbol No. |
| 1 | Anup Shrestha | 058-3-2-02784-2019 | 325126 |
| 2 | Rahul Neupane | 058-3-2-02801-2019 | 325143 |
| 3 | Rojan Pant | 058-3-2-02804-2019 | 325146 |
| 4 | Subash Danuwar | 058-3-2-02813-2019 | 325155 |

**TO WHOM IT MAY CONCERN**

This is to certify that Mr. Anup Shrestha, Mr. Rahul Neupane, Mr. Rojan pant and Mr. Subash Danuwar of Bachelor in Information Technology (BIT) has studied as per the curriculum of BIT 2nd Semester and completed the project entitled “**”**.This project is the original work of Mr. Anup Shrestha, Mr. Rahul Neupane, Mr. Rojan Pant and Mr. Subash Danuwar and was carried out under the supervision of Mr. Prawesh Dhungana as per the guidelines provided by Purbanchal University and certified as per the student’s declaration that project “**Contact Management System**” has not been presented anywhere as a part of any other academic work.

The detail of the student is as follows:

Name of Students : Anup Shrestha

Rahul Neupane

Rojan Pant

Subash Danuwar

Semester : 2nd

Subject Code : BIT 178C0

Project Title : **Contact Management System**

…………………………….

Mr. Deepak Khadka  
Program Coordinator, BIT  
KIST College of Information Technology

**TABLE OF CONTENTS**

[ABSTRACT 1](#_Toc75172186)

[INTRODUCTION 2](#_Toc75172187)

[OBJECTIVES 3](#_Toc75172188)

[FEATURES 4](#_Toc75172189)

[SYSTEM REQUIREMENTS 4](#_Toc75172190)

[ALGORITHM 6](#_Toc75172191)

[FLOWCHART 7](#_Toc75172192)

[FUNCTIONAL REQUIREMENT 8](#_Toc75172193)

[LIMITATIONS 9](#_Toc75172194)

[CONCLUSION 10](#_Toc75172195)

[FUTURE SCOPE 10](#_Toc75172196)

[REFERENCES 12](#_Toc75172197)

[APPENDIX 13](#_Toc75172198)

[SNAPSHOTS 13](#_Toc75172199)

[SOURCE CODE 16](#_Toc75172200)

# ABSTRACT

The purpose of **“Contact Management System”** is to automate the existing manual system by the help of computerized equipments and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

**Contact Management System,** as declared above, can lead to error free, secure, reliable and fastmanagement system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

The aim is to automate its existing manual system by the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. Basically the project describes how to manage for good performance and better services for the clients.

**According’s:** CMS refers to Contact management system.

# INTRODUCTION

The CMS has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some case reduce the hardships faced by this existing system. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user friendly.

CMS, as described above can lead to error free, secure, reliable and fastmanagement system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

Every organization, whether big or small, has challenges to overcome and managing the information of Contact, Profile, Mobile, Emails. Every CMS has different Contact need, therefore we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

# OBJECTIVES

The main objective of the Project on CMS is to manage the details of Contact, Telephone, Profile, Emails. It manages all the information about Contact, Mobile, Emails, Contact. The Project is totally built at administrative end and thus only the administrator is guaranteed the access. The Purpose of the project is to build an application program to reduce the manual work for managing the Contact, Mobile, Telephone. It tracks all the details about the Telephone, Profile, Emails.

**Functionalities provided by Contact Management System are as follows:**

* Provides the searching facilities based on various factor. Such as Contact, Telephone, Profile, Emails.
* CMSalso manage the mobile details online for profiledetails, Emails details, Contact.
* It tracks all the information of Mobile, Profile etc.
* Shows the information and description of the Contact, Telephone.
* To increase efficiency of managing the Contact.
* It deals with monitoring the information and translation of profile.
* Manage the information of Contact.
* Editing, Adding and Updating of records is improve which results in proper resource management of Contact data.
* Manage the information of Profile.
* Integration of all records of Emails.

# FEATURES

A CMS usually tracks and stores contact information in a database with universal access. Companies can group the entire information of their potential clients, employees, suppliers. Free CMS is a kind of customer relationship management tool. But, CMS has specific purposes. Whereas, It is a modern tool, much more advance.

* Accuracy in work.
* Easy and Fast retrieval of information.
* Well Designed reports.
* Decrease the loads of the person involve in existing manual system.
* Access of any information individually.
* Work becomes very speedy.
* Easy to update information.

# SYSTEM REQUIREMENTS

Following hardware and software requirement should be met for flawless running of this system:

**Hardware:** Hardware is the collection of physical parts of a computer system. This includes the

computer case, monitor, keyboard, and mouse. It also includes all the parts inside the computer

case, such as the hard disk drive, motherboard, video card, and many others. Computer hardware

is what you can physically touch.

RAM: 256 MB

Hard Disk: 200MB or above free space available

Processor: Intel Pentium III or higher

**Software:** Software is a set of instructions, data or programs used to operate computers and

execute specific tasks. It is the opposite of hardware, which describes the physical aspects of a

Computer.

OS: Windows 7 or Higher

Applications: Dev C++ or other C++ compiler.

# ALGORITHM

Step 1: Start.

Step 2: Shows Main menu.

Step 3: For condition of menu program further shows option as:

* + 1. Add contacts
    2. Show available contacts
    3. Search contact
    4. Edit contact
    5. Delete contact
    6. Exit

Step 4: The program provides functions to run on the basis of option chosen by user.

Step 5: If user chooses option 6, user is asked for confirmation.

Then if chosen to exit go to step 6.

Else go to step 2.

Step 6: Stop.

# FLOWCHART

Flowchart is a diagram that represents workflow or process.

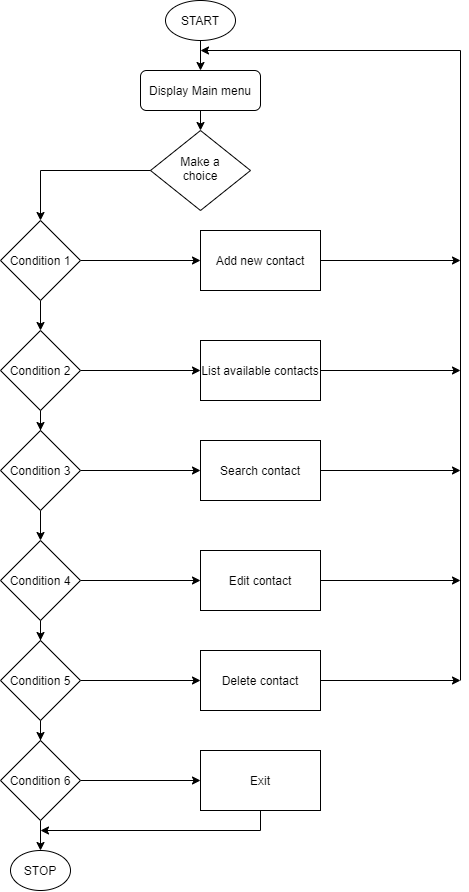


Figure 1: Flow chart

# FUNCTIONAL REQUIREMENT

In software and system engineering, a functional requirement defines a function of a system or its component, where a function is described as a specification of behavior between input and outputs.

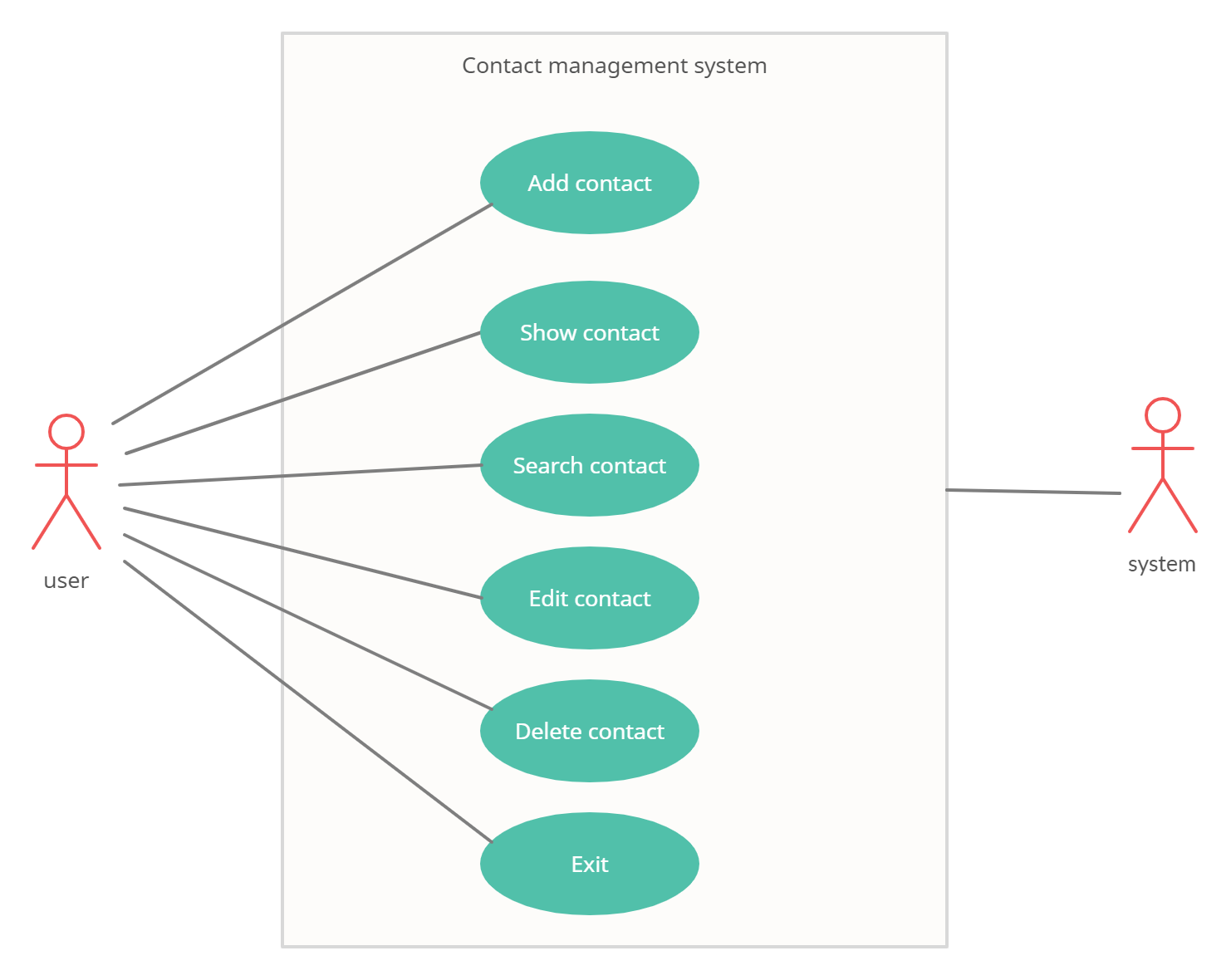
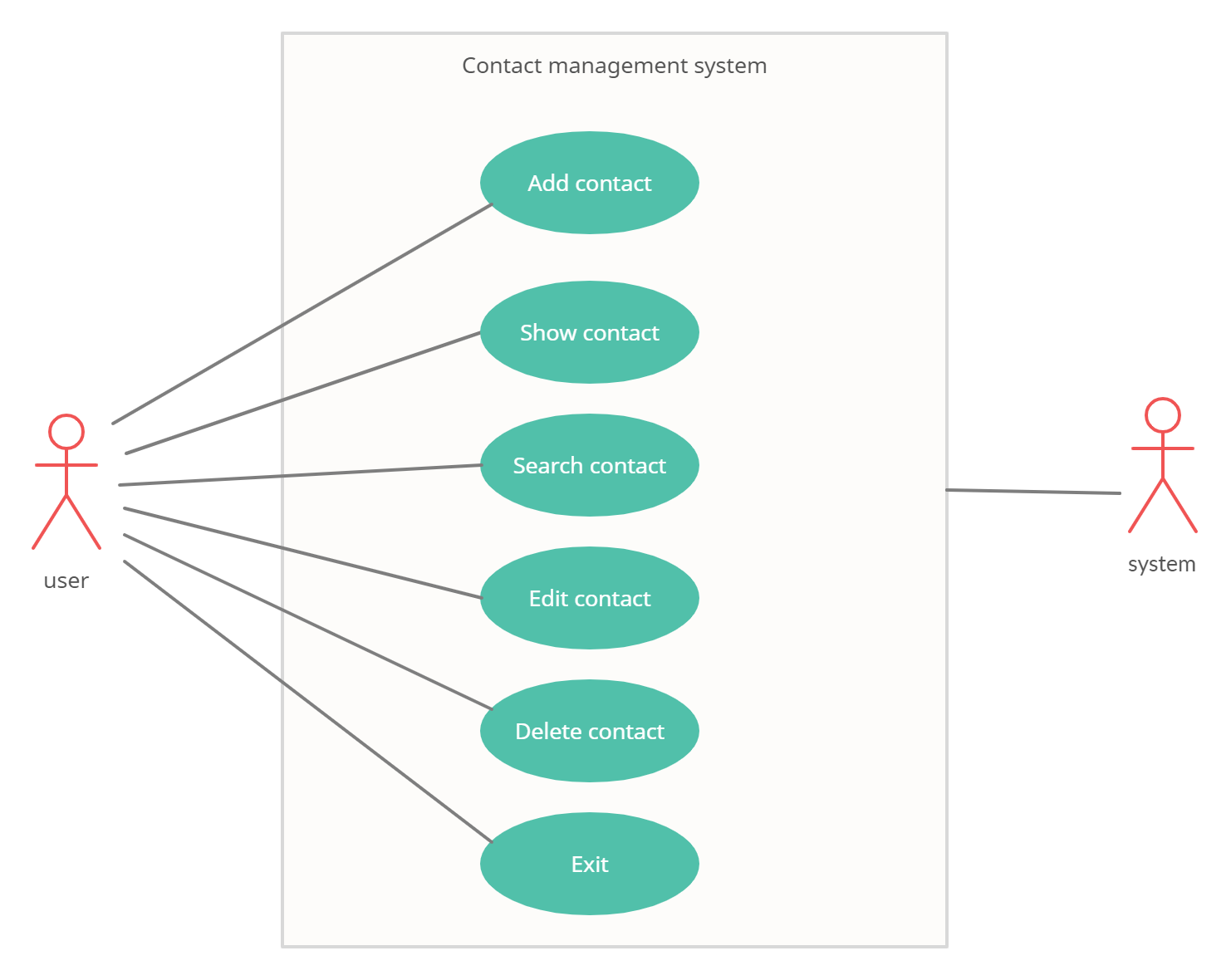
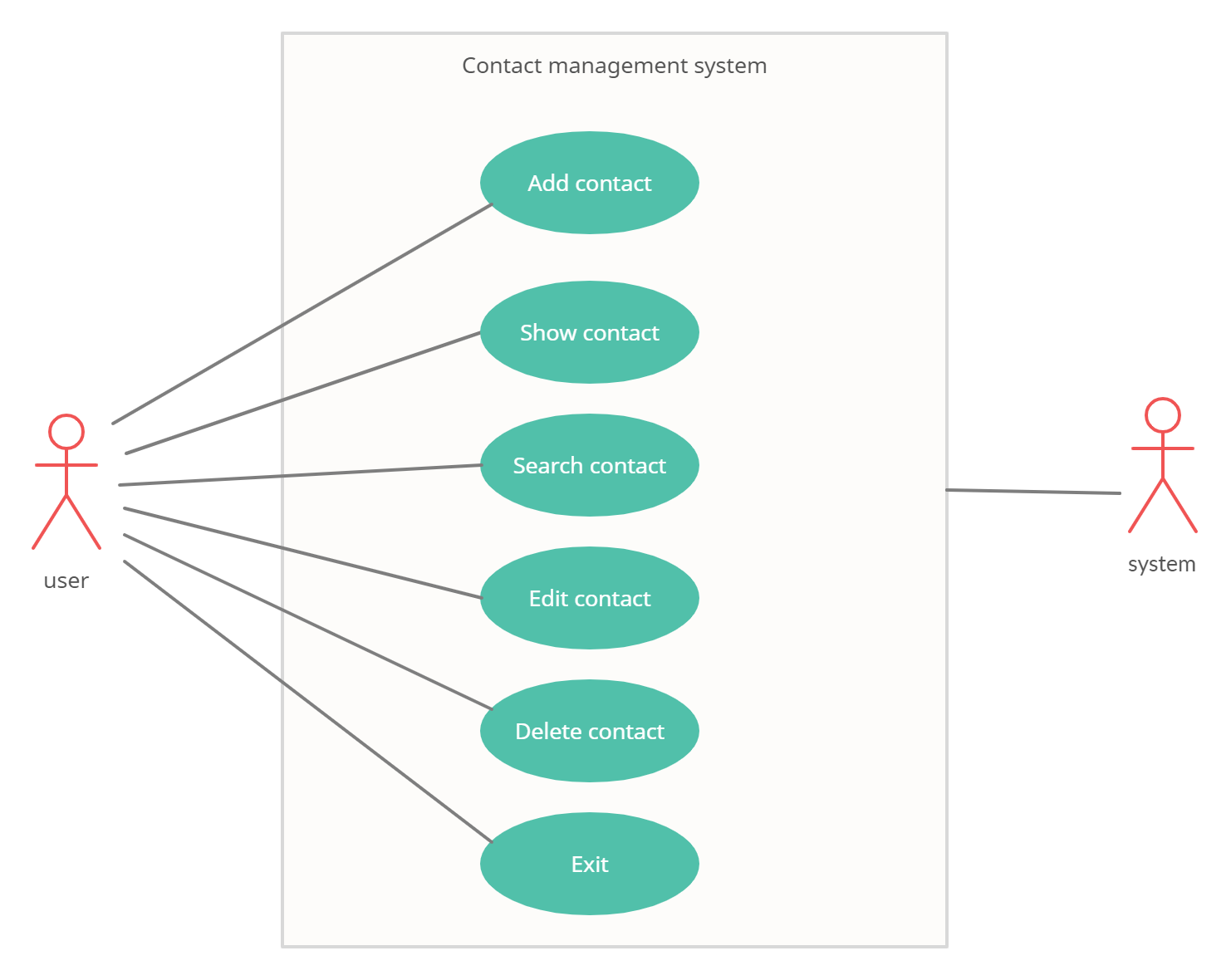
****

Figure 2: Use case diagram





# LIMITATIONS

Although, We have put our best efforts to make the software flexible, easy to operate but limitations cannot be ruled out even by us. Though the software presents a broad range of options to its users some intricate options could not be covered into it; partly because of logistic and partly due to lack of sophistication. Paucity of time was also major constraint, thus it was not possible to make the software foolproof and dynamic. Lack of time also compelled me to ignore some part such as storing old result of the candidate etc.

Considerable efforts have made the software easy to operate even for the people not related to the field of computers but it is acknowledged that a layman may find it a bit problematic at the first instance. The user is provided help at each step for his convenience in working with the software.

**List of limitations which is available in the Contact Management System**

* Excel export has not been developed for Contact due to some critically.
* The transactions are executed in off-line mode, hence on-line data for Telephone, Profile capture and modification is not possible.
* Off-line reports of Contact, Emails and Telephone cannot be generated due to batch mode execution.

# CONCLUSION

Our project is only a humble venture to satisfy the needs to manage the project work. Several user friendly coding have also adopted. This package shall prove to be a powerful package in satisfying all the requirements of the school. The objective of software planning is to provide a frame work that enables the manager to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

**At the end it is concluded that we have made effort on following points**

* A description of the background and context of the project and its relation to work already done in the area.
* The description of Purpose, Scope, and applicability.
* We define the problem on which we are working in the project.
* We describe the requirement Specifications of the system and the actions that can be done on these things.
* We included features and operations in detail, including screen layouts.
* Finally the system is implemented and tested according to test cases.

## FUTURE SCOPE

It may help collecting perfect management in details. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of Passed years perfectly and vividly. It also helps in current all works relative toCMS. It will be also reduced the cost of collecting the management and collection procedure will go on smoothly.

Our project aims at process automation, i.e. we have tried to computerize various processes of CMS

* In computer system the person has to fill the various forms & number of copies of the forms can be easily generated at a time.
* In computer system, it is not necessary to create the manifest but we can directly print it, which saves our time.
* To assist the staff in capturing the effort spent on their respective working areas.
* To utilize resources in an efficient manner by increasing their productivity through automation.
* It satisfy the user requirement.
* Be easy to understand by the user and operator.
* Be easy to operate.
* Have a good user interface
* Be expandable
* Delivered on schedule within the budget.

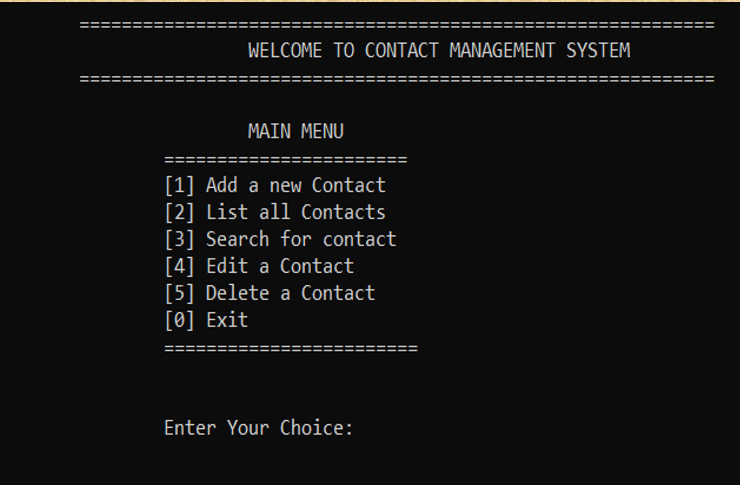
# REFERENCES

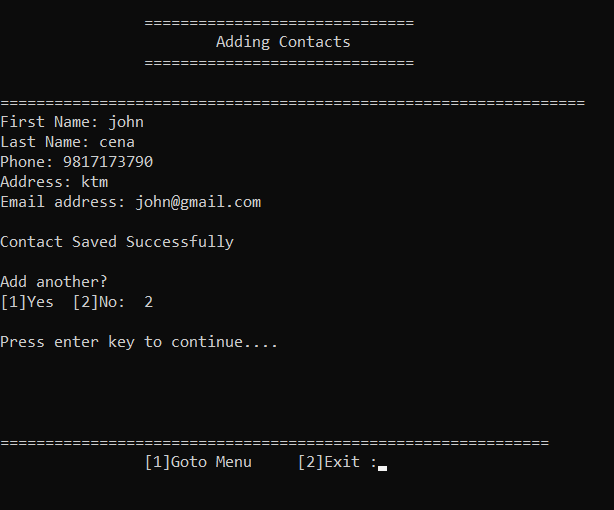
* [www.w3schools.com](http://www.w3schools.com/)
* [www.sites.google.com/site/completelearning.com](http://www.sites.google.com/site/completelearning.com)
* [www.sourcecodesworld.com](http://www.sourcecodesworld.com/)
* www.cprogramming.com
* www.draw.io

# APPENDIX

## SNAPSHOTS

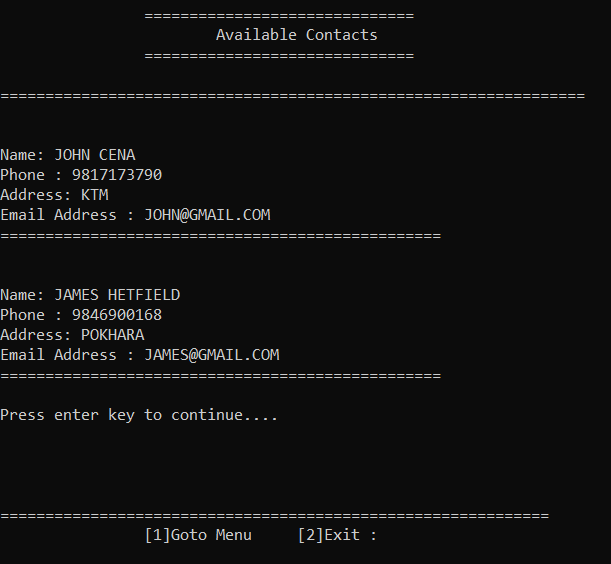
**MAIN MENU**

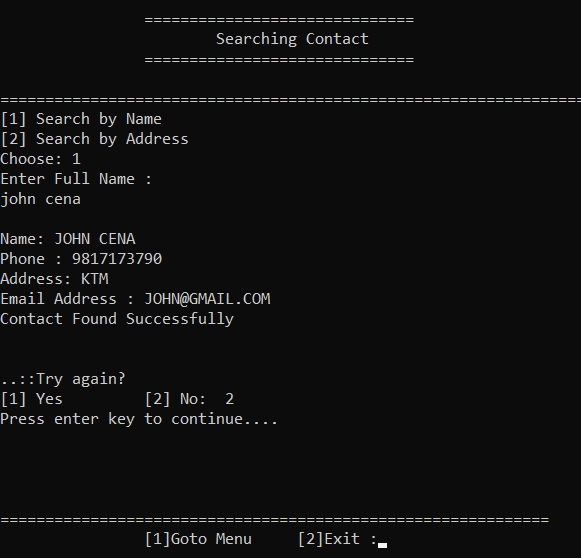
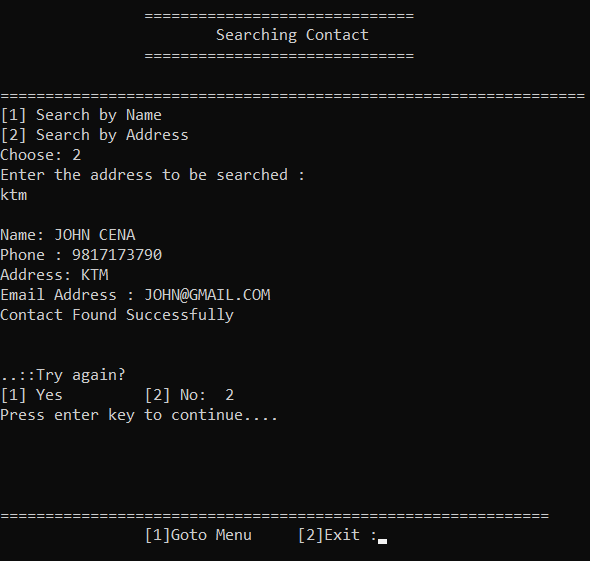




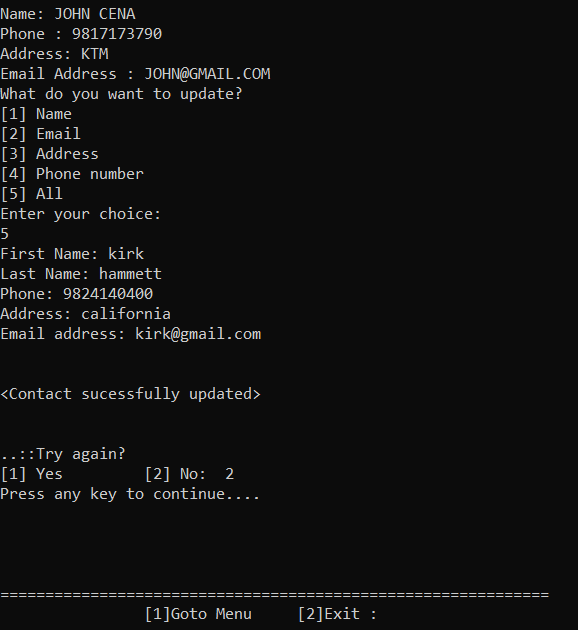
**ADDING CONTACTS**

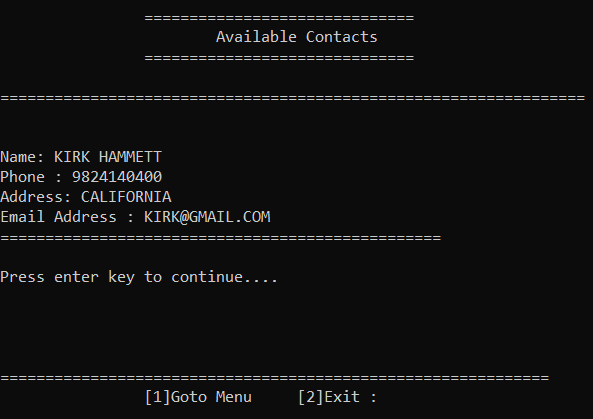
**DISPLAY CONTACTS**

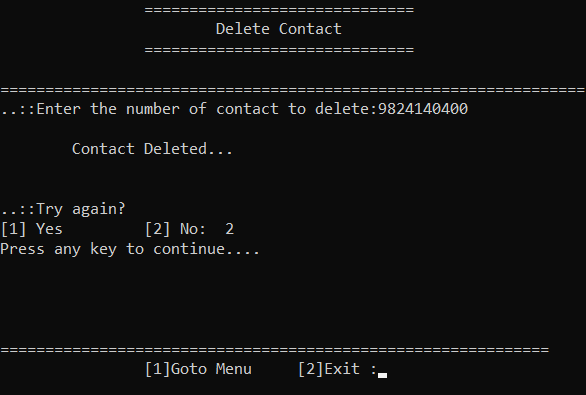


**SEARCHING CONTACT**

**EDITING CONTACT**







**DELETING CONTACT**



## SOURCE CODE

//Contact Management System

#include<fstream>

#include<windows.h>

#include<string.h>

#include<iostream>

#include<conio.h>

using namespace std;

//START OF CLASS

class contact

{

private:

unsigned long long int pno;

char name[30],lname[10],email[20],address[50];

public:

//FUNCTION FOR CREATING A CONTACT

void create\_contact()

{

fstream fp;

fp.open("contacts.txt");

contact c1;

cout<<"First Name: ";

cin.ignore();

cin>>name;

cout<<"Last Name: ";

cin.ignore();

cin>>lname;

strcat(name," ");

strcat(name,lname);

strupr(name);

phone:

cout<<"Phone: ";

cin.ignore();

cin>>pno;

if(pno<=9800000000 || pno>=9899999999)

{

cout<<"\n<Phone number should start from '98' and must have 10 digit>\nEnter again ";

goto phone;

}

while(fp.read((char\*)&c1,sizeof(c1)))

{

while(c1.getPhone()==pno)

{

cout<<"\n\t<Already registered!>\nEnter another num: ";

cin>>pno;

}

}

cout<<"Address: ";

cin.ignore();

cin>>address;

strupr(address);

cout<<"Email address: ";

cin.ignore();

cin>>email;

strupr(email);

cout<<"\n";

//END OF CREATING FUNCTION

}

void setName()

{

cout<<"First Name: ";

cin.ignore();

cin>>name;

cout<<"Last Name: ";

cin.ignore();

cin>>lname;

strcat(name," ");

strcat(name,lname);

strupr(name);

}

void setAddress()

{

cout<<"Address: ";

cin.ignore();

cin>>address;

strupr(address);

}

void setEmail()

{

cout<<"Email address: ";

cin.ignore();

cin>>email;

strupr(email);

}

void setNumber()

{

fstream fp;

fp.open("contacts.txt");

contact c1;

phone:

cout<<"Phone: ";

cin.ignore();

cin>>pno;

if(pno<=9800000000 || pno>=9899999999)

{

cout<<"\n<Phone number should start from '98' and must have 10 digit>\nEnter again ";

goto phone;

}

while(fp.read((char\*)&c1,sizeof(c1)))

{

while(c1.getPhone()==pno)

{

cout<<"\n\t<Already registered!>\nEnter another num: ";

cin>>pno;

}

}

fp.close();

}

void show\_contact()

{

cout<<endl<<"Name: "<<name;

cout<<endl<<"Phone : "<<pno;

cout<<endl<<"Address: "<<address;

cout<<endl<<"Email Address : "<<email;

}

unsigned long long int getPhone()

{

return pno;

}

char\* getName()

{

return name;

}

char\* getAddress()

{

return address;

}

char\* getEmail()

{

return email;

}

};

contact c1,c;

fstream fp;

//FUNCTION FOR ADDING A CONTACT

void addcontact()

{

system("cls");

int choice=0;

cout<<"\n\t\t==============================\n\t\t\tAdding Contacts\n\t\t==============================\n";

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

fp.open("contacts.txt",ios::out|ios::app);

c1.create\_contact();

fp.write((char\*)&c1,sizeof(c1));

fp.close();

cout<<"Contact Saved Successfully"<<endl;

flag1:

cout<<"\nAdd another? \n";

cout<<"[1]Yes \t[2]No: ";

cin>>choice;

cout<<"\n";

if(choice!=1 && choice!=2)

{

cout<<"\n<<<Invalid input>>>";

}

else if(choice==1)

{

addcontact();

}

else

{

return;

}

//END OF ADDING FUNCTION

}

//FUNCTION FOR DIAPLAYING ALL SAVING CONTACTS

void allcontacts()

{

bool found = false;

system("cls");

cout<<"\n\t\t==============================\n\t\t\tAvailable Contacts\n\t\t==============================\n";

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

fp.open("contacts.txt",ios::in);

while(fp.read((char\*)&c1,sizeof(c1)))

{

found= true;

c1.show\_contact();

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

}

if(found==false)

cout<<"\n<NO contacts found...>"<<endl;

fp.close();

//END OF DISPLAY FUNCTION

}

//FUNCTION FOR SEARCHING A CONTACT

void searchcontact()

{

system("cls");

int counter=0,choice=0,key;

do{

cout<<"\n\t\t==============================\n\t\t\tSearching Contact\n\t\t==============================\n";

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

char user[30],user1[10],addr[20];

fp.open("contacts.txt",ios::in);

cout<<"[1] Search by Name"<<endl;

cout<<"[2] Search by Address"<<endl;

cout<<"Choose: ";

cin>>key;

switch(key)

{

case 1:

cout<<"Enter Full Name :";

cin>>user>>user1;

strcat(user," ");

strcat(user,user1);

strupr(user);

while(fp.read((char\*)&c1,sizeof(c1)))

{

if(strcmp(c1.getName(),user)==0)

{

c1.show\_contact();

counter++;

}

}

key=3;

break;

case 2:

cout<<"Enter the address to be searched :";

cin>>addr;

strupr(addr);

while(fp.read((char\*)&c1,sizeof(c1)))

{

if(strcmp(c1.getAddress(),addr)==0)

{

c1.show\_contact();

counter++;

}

}

key=3;

break;

default:

break;

}

}while(key!=3);

fp.close();

if(counter==0)

cout<<" Contact Not Found "<<endl;

else

cout<<"\nContact Found Successfully"<<endl;

fp.close();

flag2:

cout<<"\n\n..::Try again?\n[1] Yes\t\t[2] No: ";

cin>>choice;

if(choice!=1 && choice!=2)

{

cout<<"\n<<<Invalid input>>>";

}

else if(choice==1)

{

searchcontact();

}

else

{

return;

}

//END OF SEARCHING FUNCTION

}

void chooseUpdate(contact& c2)

{

system("cls");

int choice;

again:

c1.show\_contact();

cout<<endl<<"What do you want to update?"<<endl;

cout<<"[1] Name"<<endl;

cout<<"[2] Email"<<endl;

cout<<"[3] Address"<<endl;

cout<<"[4] Phone number"<<endl;

cout<<"[5] All"<<endl;

cout<<"Enter your choice:"<<endl;

cin>>choice;

switch(choice)

{

case 1:

c2.setName();

break;

case 2:

c2.setEmail();

break;

case 3:

c2.setAddress();

break;

case 4:

c2.setNumber();

break;

case 5:

c2.create\_contact();

break;

default:

goto again;

}

}

//FUNCTION FOR UPDATING CONTACT

void updatecontact()

{

system("cls");

int choice=0;

bool found = false;

unsigned long long int phn;

cout<<"\n\t\t==============================\n\t\t\tEdit Contact\n\t\t==============================\n";

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

cout<<"..::Enter the number of contact to edit: ";

cin.ignore();

cin>>phn;

fp.open("contacts.txt",ios::in|ios::out);

while(fp.read((char\*)&c1,sizeof(c1)))

{

if(c1.getPhone()==phn)

{

chooseUpdate(c1);

int pos=(-1)\*sizeof(c1);

fp.seekp(pos,ios::cur);

fp.write((char\*)&c1,sizeof(c1));

found = true;

cout<<"\n<Contact sucessfully updated>"<<endl;

break;

}

}

if(found==false)

cout<<"\nContact Not Found..."<<endl;

fp.close();

flag3:

cout<<"\n\n..::Try again?\n[1] Yes\t\t[2] No: ";

cin>>choice;

if(choice!=1 && choice!=2)

{

cout<<"\n<<<Invalid input>>>";

}

else if(choice==1)

{

updatecontact();

}

else

{

return;

}

//END OF UPDATING FUNCTION

}

//FUNCTION FOR DELETING A CONTACT

void deletecontact()

{

system("cls");

int counter=0,choice=0;

unsigned long long int phn;

cout<<"\n\t\t==============================\n\t\t\tDelete Contact\n\t\t==============================\n";

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

cout<<"..::Enter the number of contact to delete:";

cin.ignore();

cin>>phn;

fp.open("contacts.txt",ios::in|ios::out);

fstream ft;

ft.open("temp.txt",ios::out);

fp.seekg(0,ios::beg);

while(fp.read((char\*)&c1,sizeof(c1)))

{

if(c1.getPhone()==phn)

{

counter++;

}

else

{

ft.write((char\*)&c1,sizeof(c1));

}

}

ft.close();

fp.close();

remove("contacts.txt");

rename("temp.txt","contacts.txt");

if(counter==0)

{

cout<<"Contact not found...."<<endl;

}

else

{

cout<<"\n\tContact Deleted..."<<endl;

//END OF DELETING FUNCTION

}

flag4:

cout<<"\n\n..::Try again?\n[1] Yes\t\t[2] No: ";

cin>>choice;

if(choice!=1 && choice!=2)

{

cout<<"\n<<<Invalid input>>>";

}

else if(choice==1)

{

deletecontact();

}

else

{

return;

}

}

//FUNCTION FOR DISPLAYING MAINMENU

void mainmenu()

{

int ch,i=0;

system("cls"); //loading screen

cout<<"\n\n\n\n\n\n\t\t\t\t\tSYSTEM LOADING \n\t\t\t\t\t";

for(i=1;i<6;i++)

{

Sleep(500);

cout<<"...";

}

cout<<"\n\n\t\t\t\t\t<System Sucessfully Loaded>\n\n";

cout<<"\t\t\t\t\t<Enter any key to continue>";

getch();

system("cls");

flag5:

system("cls");

//\*\*\*\*\*\*\*\*\*\*\*\*Main menu \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

cout<<"\n\t============================================================\n\t\t\tWELCOME TO CONTACT MANAGEMENT SYSTEM\n\t============================================================\n";

cout<<"\n\t\t\tMAIN MENU\n\t\t=======================\n\t\t";

cout<<"[1] Add a new Contact\n\t\t";

cout<<"[2] List all Contacts\n\t\t";

cout<<"[3] Search for contact\n\t\t";

cout<<"[4] Edit a Contact\n\t\t";

cout<<"[5] Delete a Contact\n\t\t";

cout<<"[0] Exit\n\t\t========================\n\t\t";

cout<<"\n\n\t\tEnter Your Choice: ";

cin>>ch;

if(ch<0 && ch >5)

{

cout<<"\n\n\*\*Invalid input\*\*";

}

switch(ch)

{

case 0:

exit:

cout<<"\n\n\t\tEXit for sure ?\n\t\t[1]Yes [2]No: ";

cin>>ch;

if(ch==1)

{

cout<<"\n\n\n\t\t\t<<<<<<<<<< Thank you for using the system >>>>>>>>>>>>>>>\n\n\n\n";

cout<<"\n\n\n\n\n\n\t\t\t\t\tSystem Closing \n\t\t\t\t\t";

for( i=1;i<6;i++)

{

Sleep(500);

cout<<"...";

}

cout<<"\n\t\t\t\t\t<System Sucessfully closed>";

getch();

exit (0);

}

else if(ch ==2)

{

goto flag5;

}

else

{

cout<<"\n\t\t<Invalid data provided>";

goto exit;

}

case 1:

{

addcontact();

cout<<"Press enter key to continue...."<<endl;

cin.get();

cin.ignore();

break;

}

case 2:

{

allcontacts();

cout<<"Press enter key to continue...."<<endl;

cin.get();

cin.ignore();

break;

}

case 3:

{

searchcontact();

cout<<"Press enter key to continue...."<<endl;

cin.get();

cin.ignore();

break;

}

case 4:

{

updatecontact();

cout<<"Press any key to continue...."<<endl;

cin.get();

cin.ignore();

break;

}

case 5:

{

deletecontact();

cout<<"Press any key to continue...."<<endl;

cin.get();

cin.ignore();

break;

}

case 6:

exit (0);

default:

cout<<"\n\n\t\t\t\t\t<Enter valid option>"<<endl;

cout<<"\n\n\t\t\t\t\tpress Enter key to continue";

getchar();

break;

}

flag6:

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

cout<<"\n\t\t[1]Goto Menu\t [2]Exit :";

cin>>ch;

if(ch==1)

{

goto flag5;

}

else if(ch==2)

{

goto exit;

}

else

{

cout<<"\n\t\t<Invalid data provided>";

goto exit;

}

}

int main()

{

mainmenu();

return 0;

} //END OF PROGRAM