PROJECT REPORT

CONTACT MANAGEMENT SYSTEM

Team Members: -

M.VISHNU VARDHAN (RA2111026010437)

M.ROHAN REDDY (RA2111026010445)

LALIT KRISHNAN S. R (RA2111026010453)

PARATALA PRANAV MURARI (RA2111026010454)

AVULA CHAKRI GOVIND YADAV (RA2111026010460)

Submitted To :-

PRABAKARAN SIR.

Contact Management System In C

Index:-

❖ Aim:-

Using this project, we can easily add, view, edit, search and delete contacts. We can list contacts by name, phone number, address, and email.

❖ Algorithm:-

Step 1: Start

Step 2: Accept choices from main menu below

```
       MAIN MENU
=====================
[1] Add a New Contact
[2] List all Contacts
[3] Search for Contact
[4] Edit a Contact
[5] Delete a Contact
[0] Exit
=================
```

Step 3: If the choice is 1, accept and store name, phone, address , email address. After accepting details choose [1] to go back to Main menu or [0] to exit.

Step 4: If the choice is 2, display the stored data. After displaying details choose [1] to go back to Main menu or [0] to exit.

Step 5: If the choice is 3, search for contact name if match found display the matched data if no match found display no match found

and try again. Choose [1]Yes or [0]No to try again .If [0]No is chosen display [1] to go back to Main menu or [0] to exit.
Step 6: If the choice is 4, accept name of contact to be edited. Replace the changed data to that of the previous data. After accepting details choose [1] to go back to Main menu or [0] to exit.
Step 7: If the choice is 5, accept name of the contact to be deleted. After accepting details choose [1] to go back to Main menu or [0] to exit.
Step 8: If the choice is [0]Exit , choose [1] to go back to Main menu or [0] to exit.
Step 9: Stop.

❖ Procedure (Module description):-

File handling, data structure, functions, and pointers are the main things which make up this simple c mini project. The key features of contact management system are listed below:

- Add new contacts: with information such as name, phone number, address, and email.
- List all contacts: lists all the contacts stored in file with their respective contact details.
- Search contacts: based on name and phone number.
- Edit contacts: edit information given while adding the contacts – name, phone number, address, and email.
- Delete contacts: deletes contacts from file.

❖ Code:-

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#include<process.h>
#include<stdlib.h>
#include<dos.h>
```

```
struct contact

{
   long ph;
   char name[20],add[20],email[30];
} list;
char query[20],name[20];
FILE *fp, *ft;
int i,n,ch,l,found;
int main()                                                  //Start of main
{
main:
   system("cls");    /* ************Main menu
*********************** */
   system("color 3f");
   printf("\n\t **** Welcome to Contact Management System
****");
   printf("\n\n\n\t\t\tMAIN
MENU\n\t\t=====================\n\t\t[1] Add a New
Contact\n\t\t[2] List all Contacts\n\t\t[3] Search for Contact\n\t\t[4]
Edit a Contact\n\t\t[5] Delete a Contact\n\t\t[0]
Exit\n\t\t=================\n\t\t");
   printf("Enter Your Choice: ");
   scanf("%d",&ch);
   switch(ch)
   {
   case 0:                                      //This case is used for Exit
      printf("\n\n\t\tAre You Sure To Exit?");
      break;
      /* ***********************Add new contacts*************
*/
   case 1:                    //This case is used for Add a new Contact
      system("cls");
      fp=fopen("contact.dll","a");
      for (;;)
```

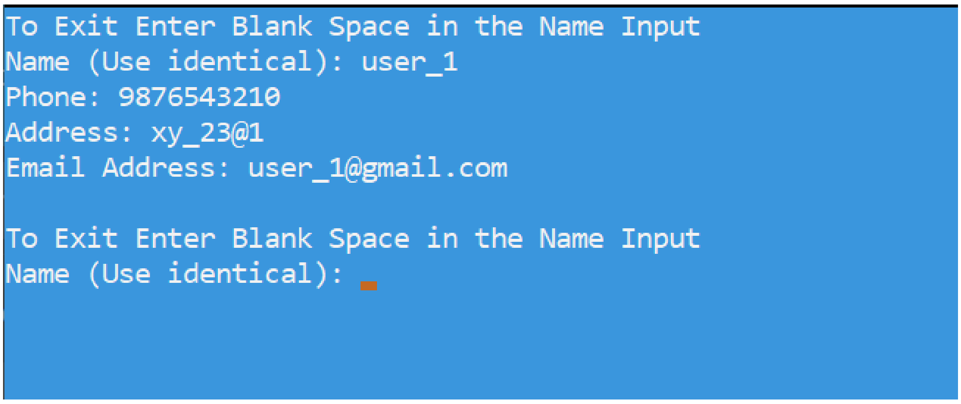
```
        {
            fflush(stdin);
            printf("To Exit Enter Blank Space in the Name
Input\nName (Use identical): ");
            scanf("%[^\n]",&list.name);
            if(stricmp(list.name,"")==0 || stricmp(list.name," ")==0)
                break;
            fflush(stdin);
            printf("Phone: ");
            scanf("%ld",&list.ph);
            fflush(stdin);
            printf("Address: ");
            scanf("%[^\n]",&list.add);
            fflush(stdin);
            printf("Email Address: ");
            gets(list.email);
            printf("\n");
            fwrite(&list,sizeof(list),1,fp);
        }
        fclose(fp);
        break;
        /* *********************list of
contacts*************************  */
    case 2:                             //This case is used for List all Contact
        system("cls");
printf("\n\t\t=================================\n\t\t\tLIS
T OF
CONTACTS\n\t\t=================================\n\n
Name\t\tPhone No\t    Address\t\tE-
mail\n============================================
======================\n\n");
        for(i=97; i<=122; i=i+1)
        {
            fp=fopen("contact.dll","r");
            fflush(stdin);
        found=0;
        while(fread(&list,sizeof(list),1,fp)==1)
        {
          if(list.name[0]==i || list.name[0]==i-32)
          {
            printf("\nName\t: %s\nPhone\t: %ld\nAddress\t:
  %s\nEmail\t: %s\n",list.name,
               list.ph,list.add,list.email);
            found++;
          }
        }
        if(found!=0)
        {
  printf("==========================================
================= [%c]-(%d)\n\n",i-32,found);
          getch();
        }
        fclose(fp);
      }
      break;
      /* *******************search
  contacts**********************  */
    case 3:           //This case is used search for Contact

system("cls");

      do
      {
        found=0;
        printf("\n\n\t..::CONTACT
  SEARCH\n\t===========================\n\t..::Name of
  contact to search: ");
        fflush(stdin);
        scanf("%[^\n]",&query);
        l=strlen(query);
        fp=fopen("contact.dll","r");
```

```
            system("cls");
            printf("\n\n..::Search result for '%s'
\n============================================
=====\n",query);
            while(fread(&list,sizeof(list),1,fp)==1)
            {
                for(i=0; i<=l; i++)
                    name[i]=list.name[i];
                name[l]='\0';
                if(stricmp(name,query)==0)
                {
                    printf("\n..::Name\t: %s\n..::Phone\t:
%ld\n..::Address\t: %s\n..::Email\t:
%s\n",list.name,list.ph,list.add,list.email);
                    found++;
                    if (found%4==0)
                    {
                        printf("..::Press any key to continue...");
                        getch();
                    }
                }
            }
            if(found==0)
                printf("\n..::No match found!");
            else
                printf("\n..::%d match(s) found!",found);
            fclose(fp);
            printf("\n ..::Try again?\n\n\t[1] Yes\t\t[0] No\n\t");
            scanf("%d",&ch);
        }
        while(ch==1);
        break;
        /* *********************edit
contacts************************/
```

```
   case 4:                        //This case is used for Edit a Contact
system("cls");
        fp=fopen("contact.dll","r");
        ft=fopen("temp.dat","w");
        fflush(stdin);
        printf("..::Edit
contact\n===============================\n\n\t..::Enter
the name of contact to edit:");
        scanf("%[^\n]",name);
        while(fread(&list,sizeof(list),1,fp)==1)
        {
            if(stricmp(name,list.name)!=0)
                fwrite(&list,sizeof(list),1,ft);
        }
        fflush(stdin);
        printf("\n\n..::Editing '%s'\n\n",name);
        printf("..::Name(Use identical):");
        scanf("%[^\n]",&list.name);
        fflush(stdin);
        printf("..::Phone:");
        scanf("%ld",&list.ph);
        fflush(stdin);
        printf("..::address:");
        scanf("%[^\n]",&list.add);
        fflush(stdin);
        printf("..::email address:");
        gets(list.email);
        printf("\n");
        fwrite(&list,sizeof(list),1,ft);
        fclose(fp);
        fclose(ft);
        remove("contact.dll");
        rename("temp.dat","contact.dll");
        break;
        /* ********************delete
contacts**********************/
    case 5:                        //This case is used for Delete a Contact
        system("cls");
        fflush(stdin);
        printf("\n\n\t..::DELETE A
CONTACT\n\t==========================\n\t..::Enter the
name of contact to delete:");
        scanf("%[^\n]",&name);
        fp=fopen("contact.dll","r");
        ft=fopen("temp.dat","w");
        while(fread(&list,sizeof(list),1,fp)!=0)
            if (stricmp(name,list.name)!=0)
                fwrite(&list,sizeof(list),1,ft);
        fclose(fp);
        fclose(ft);
        remove("contact.dll");
        rename("temp.dat","contact.dll");
        break;
    default:
        printf("Invalid choice");
        break;
    }
    printf("\n\n\n..::Enter the Choice:\n\n\t[1] Main Menu\t\t[0]
Exit\n");
    scanf("%d",&ch);
    switch (ch)
    {
    case 1:
        goto main;
    case 0:
        break;
    default:
        printf("Invalid choice");
        break;
    }
    return 0;
}                                                   //end of main
```

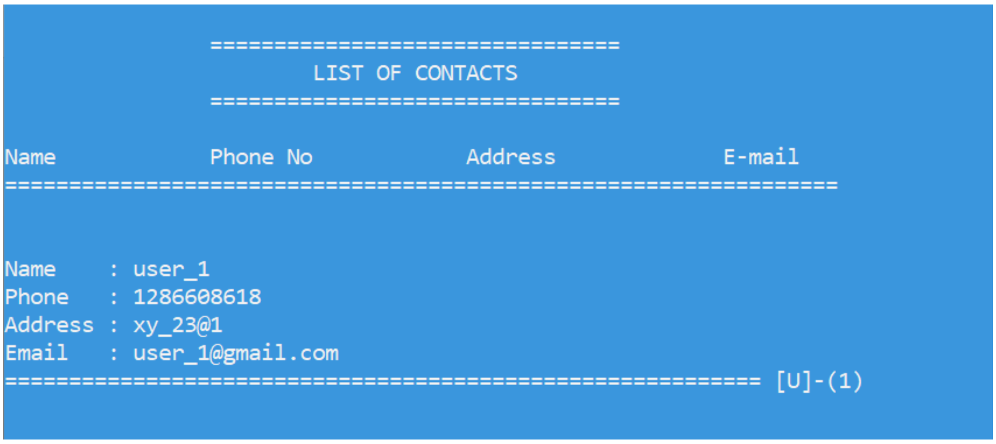
❖ Output Screenshot:-

Adding new contact-



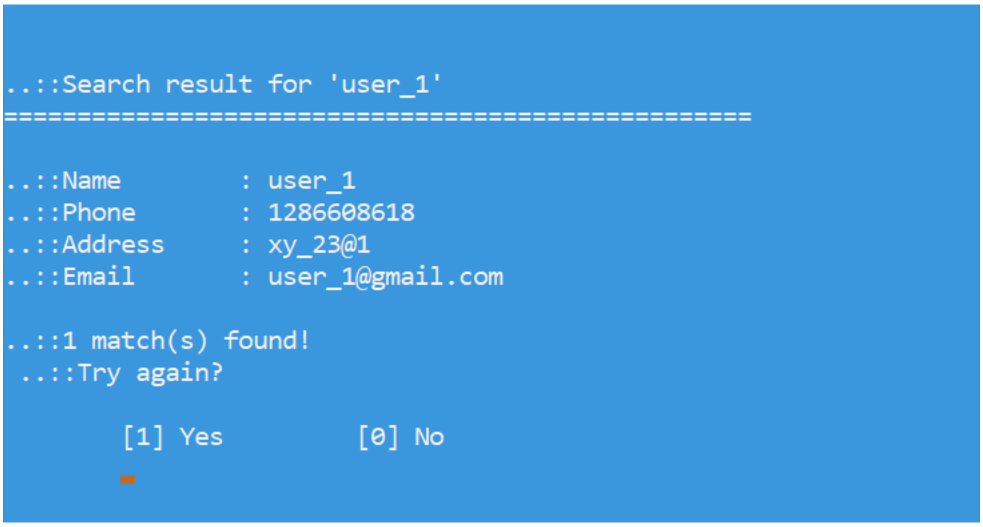
```
To Exit Enter Blank Space in the Name Input
Name (Use identical): user_1
Phone: 9876543210
Address: xy_23@1
Email Address: user_1@gmail.com

To Exit Enter Blank Space in the Name Input
Name (Use identical): _
```

List of all contacts-



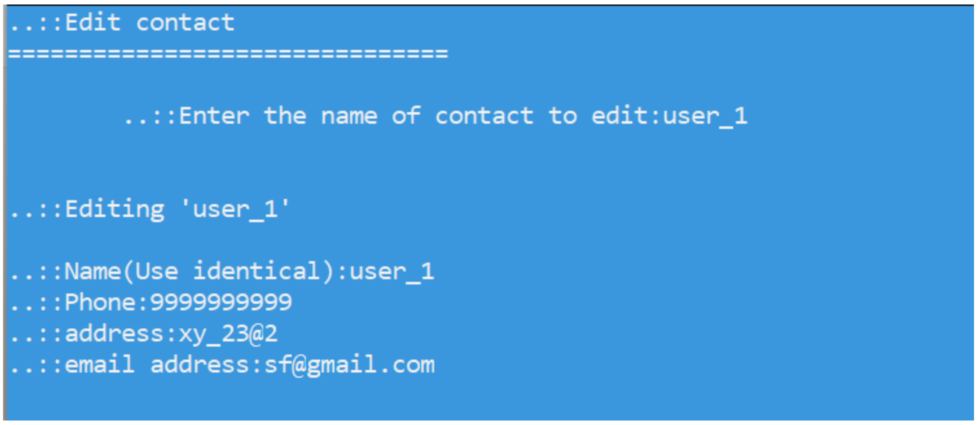
```
                ================================
                        LIST OF CONTACTS
                ================================

Name            Phone No            Address             E-mail
=================================================================


Name    : user_1
Phone   : 1286608618
Address : xy_23@1
Email   : user_1@gmail.com
=========================================================== [U]-(1)
```

Search for contact-

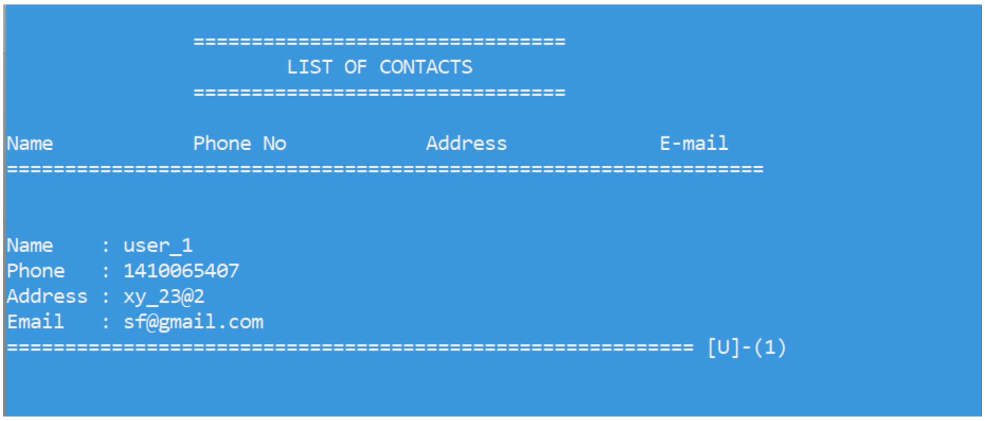


```
..::Search result for 'user_1'
====================================================

..::Name        : user_1
..::Phone       : 1286608618
..::Address     : xy_23@1
..::Email       : user_1@gmail.com

..::1 match(s) found!
 ..::Try again?

        [1] Yes         [0] No
```

Edit contact-

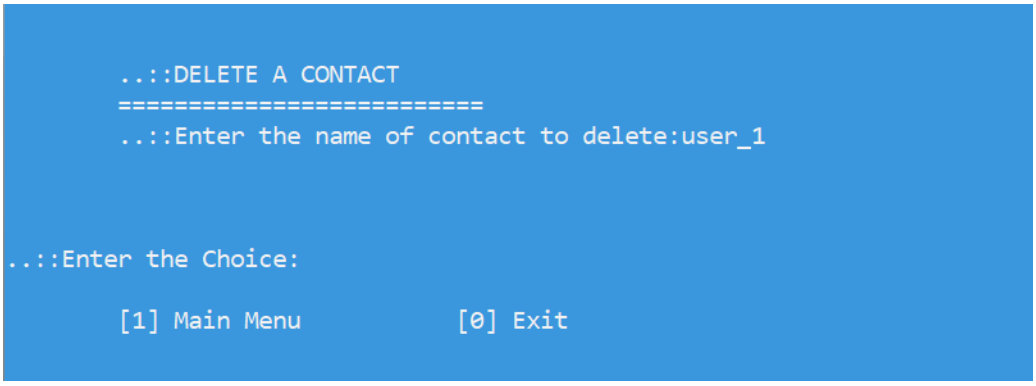


```
..::Edit contact
===============================

        ..::Enter the name of contact to edit:user_1


..::Editing 'user_1'

..::Name(Use identical):user_1
..::Phone:9999999999
..::address:xy_23@2
..::email address:sf@gmail.com
```

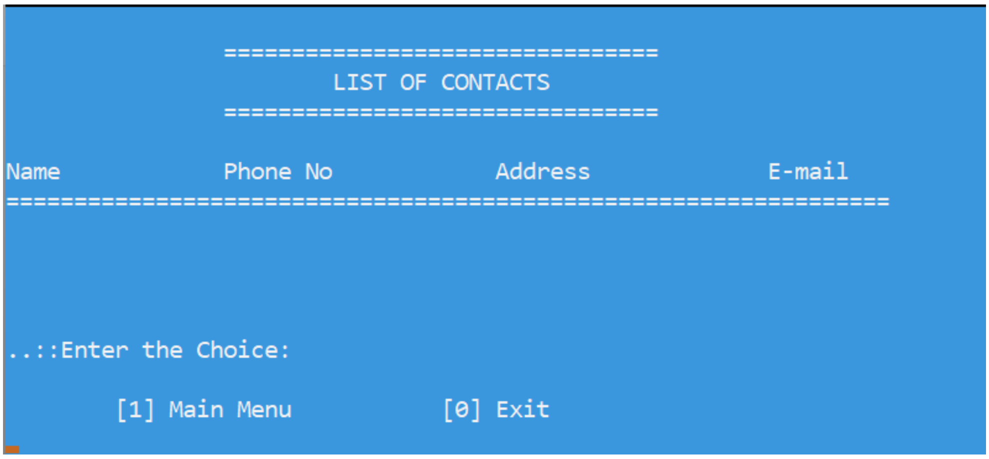


```
                ================================
                        LIST OF CONTACTS
                ================================

Name            Phone No            Address             E-mail
=================================================================


Name    : user_1
Phone   : 1410065407
Address : xy_23@2
Email   : sf@gmail.com
=========================================================== [U]-(1)
```

Deleting contact –



```
        ..::DELETE A CONTACT
        ==========================
        ..::Enter the name of contact to delete:user_1



..::Enter the Choice:

        [1] Main Menu               [0] Exit
```



```
                ================================
                        LIST OF CONTACTS
                ================================

Name            Phone No            Address             E-mail
=================================================================



..::Enter the Choice:

        [1] Main Menu               [0] Exit
```