KIET Group of Institutions, Ghaziabad

Computer Science and Information Technology

Project Report

on

Patient record system

Project at KIET Group of Institutions

Submitted By:

Vivek kumar

Shubham Agarwal

Shubham Jaiswal

B.Tech /CSIT (3-C)

Roll No. 2100290110164

2100290110163

2100290110194

ACKNOWLEDGEMENT

I've got this golden opportunity to express my kind gratitude and sincere thanks to my subject faculty **"Mr. VINAY KUMAR"**, Computer Science and Information Technology Department, **KIET GROUP OF INSTITUTIONS** for their kind support and necessary counselling in the preparation of this project report. I'm also indebted to each and every person responsible for the making up of this project directly or indirectly.

I must also acknowledge or deep debt of gratitude each one of my colleague who led this project come out in the way it is. It's my hard work and untiring sincere efforts and mutual cooperation to bring out the project work. Last but not the least, I would like to thank my parents for their sound counselling and cheerful support. They have always inspired us and kept our spirit up.

Aim: To make an patient record system and store data of the patient.

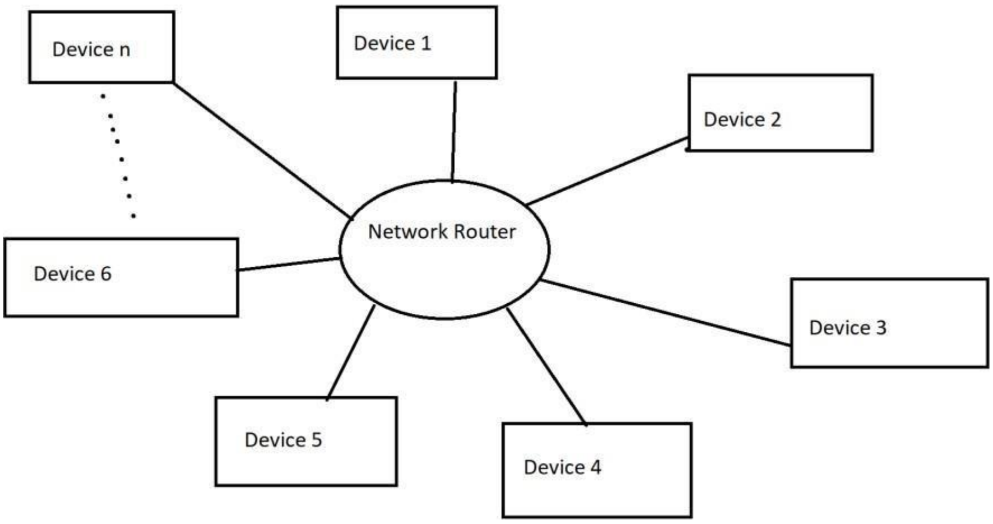
Objective: Records Serve to document the history
of the client.

Abstract: Recent research states that using new and emerging technologies in the areas of telecommunications are widely used in healthcare sector. The system Electornic Patient Record Management System (EPRMS) is a centralized database contains the in-patient record. It was implemented using PHP & MYSQL combination. The database record contains the patient personal info, department lies-in, physician, tours, ,treatment and lab results. Since the patient enters the hospital the workflow starts as the reception user creates new record by entering the personal info and sends the record to assigned department; at this stage the nurse starts update the record by entering the physician comments, required treatment, and sends lab test when it is required. The procedure continues as long as the patient still in the hospital. At last when the patient recovered or died the International Classsification of Diseases(ICD) inserted to the record and out or died date. In addition there are many supported tables that can be updated manually through independent pages by IT administrator. These tables like Physician names, medicines, lab tests, users and ICDs. As the system consists of different users and different user permissions. Also there are advance search that can help to make statistical reports and researches for the physicians. The system is considered time and cost effective to healthcare.

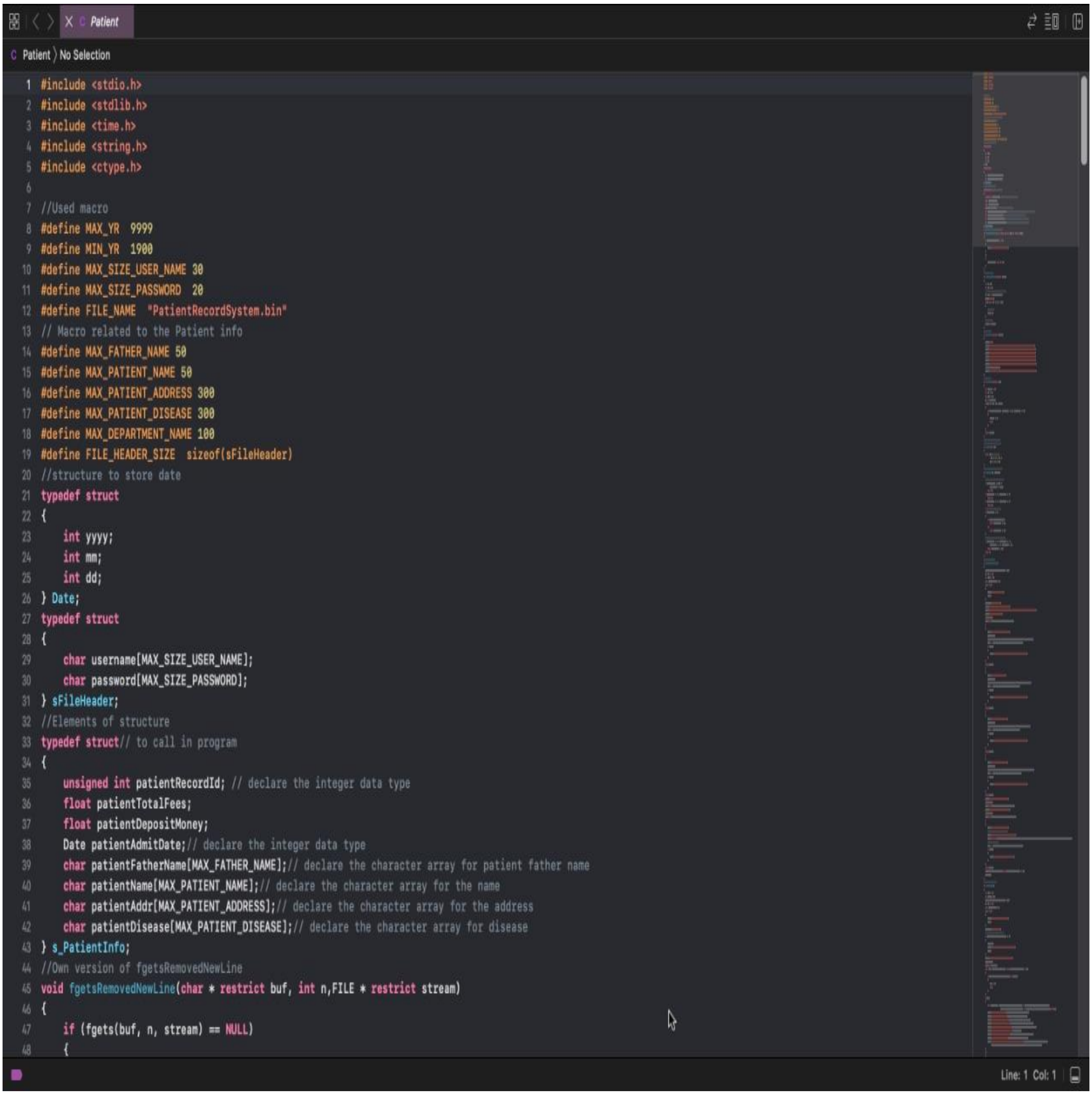
Basic principle: Records serve to recognize the health
needs and can be used as a research
and teaching tool.

Methodology: To arrange the data in structures.

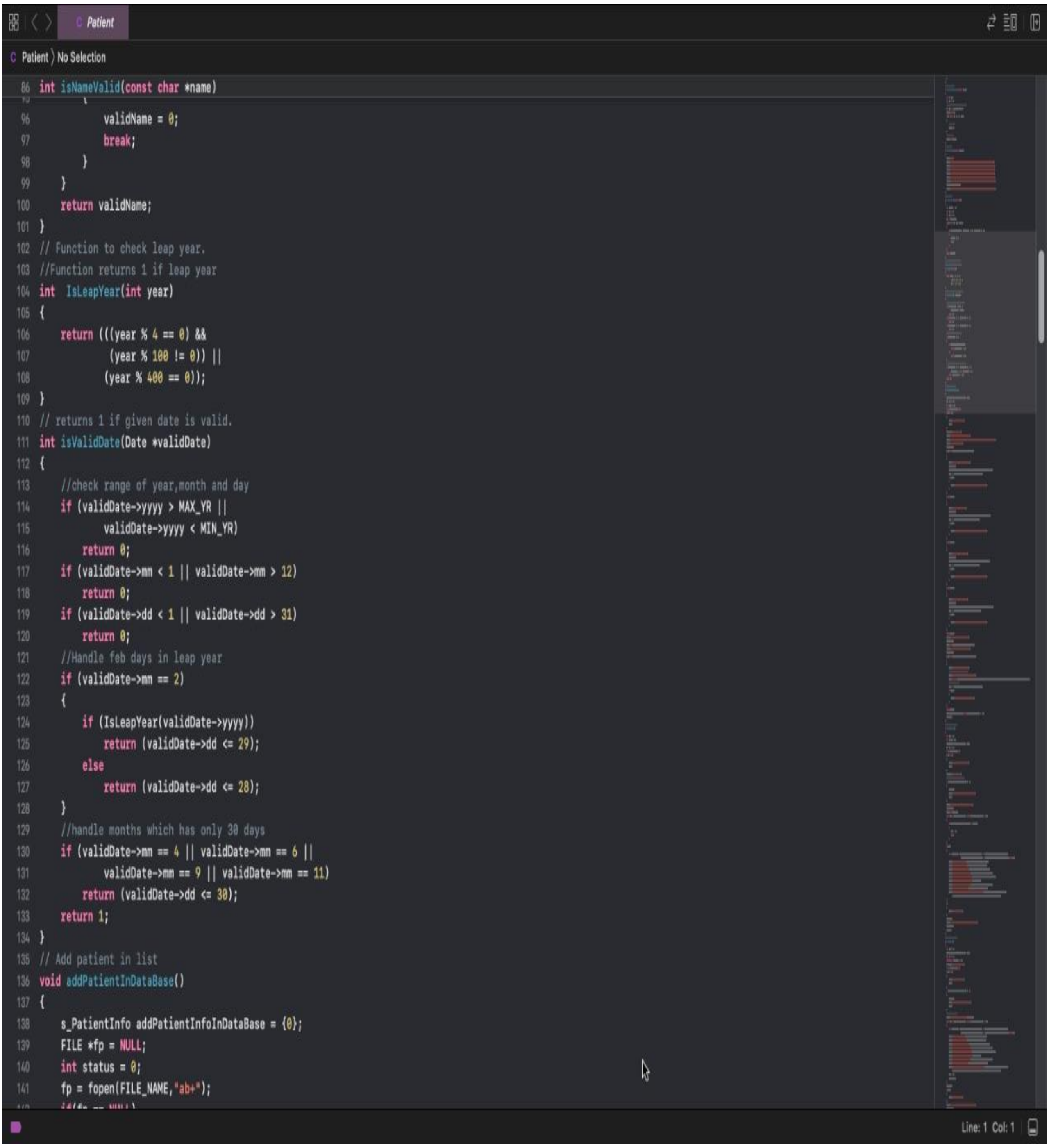
Architecture:



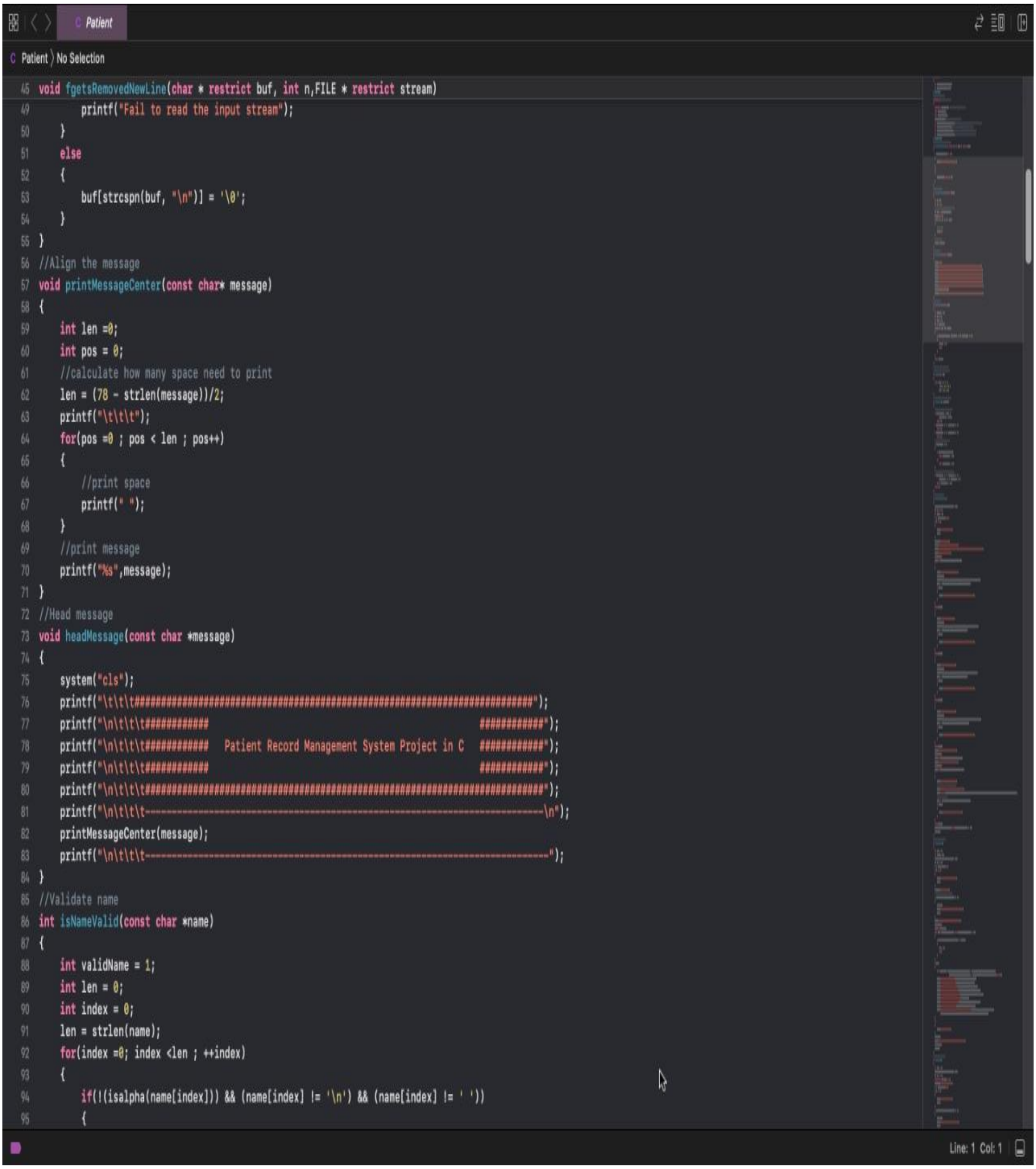
CODING IMPLEMENTATION/ OUTPUT:



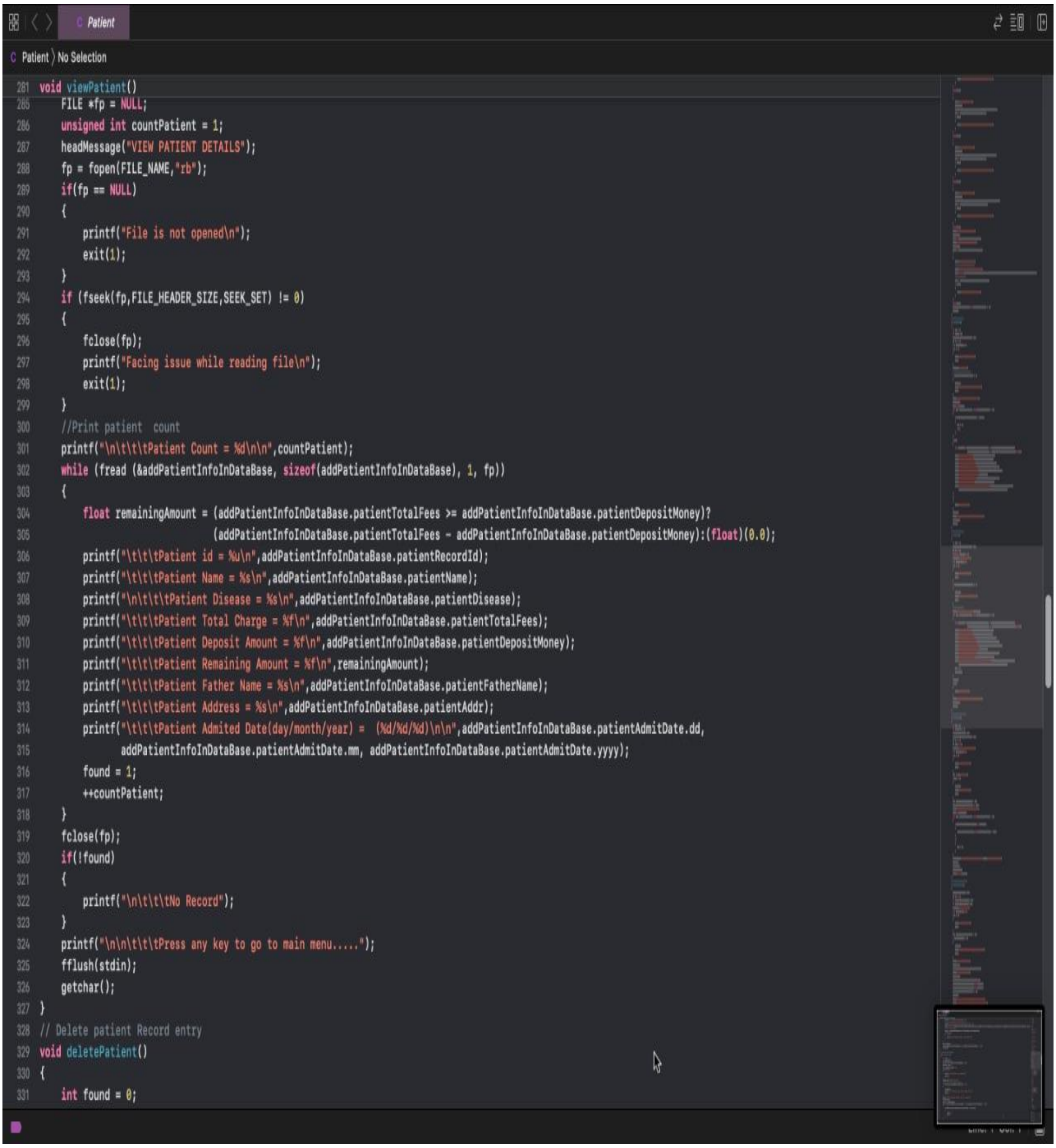
```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <string.h>
#include <ctype.h>

//Used macro
#define MAX_YR  9999
#define MIN_YR  1900
#define MAX_SIZE_USER_NAME 30
#define MAX_SIZE_PASSWORD  20
#define FILE_NAME  "PatientRecordSystem.bin"
// Macro related to the Patient info
#define MAX_FATHER_NAME 50
#define MAX_PATIENT_NAME 50
#define MAX_PATIENT_ADDRESS 300
#define MAX_PATIENT_DISEASE 300
#define MAX_DEPARTMENT_NAME 100
#define FILE_HEADER_SIZE  sizeof(sFileHeader)
//structure to store date
typedef struct
{
    int yyyy;
    int mm;
    int dd;
} Date;
typedef struct
{
    char username[MAX_SIZE_USER_NAME];
    char password[MAX_SIZE_PASSWORD];
} sFileHeader;
//Elements of structure
typedef struct// to call in program
{
    unsigned int patientRecordId; // declare the integer data type
    float patientTotalFees;
    float patientDepositMoney;
    Date patientAdmitDate;// declare the integer data type
    char patientFatherName[MAX_FATHER_NAME];// declare the character array for patient father name
    char patientName[MAX_PATIENT_NAME];// declare the character array for the name
    char patientAddr[MAX_PATIENT_ADDRESS];// declare the character array for the address
    char patientDisease[MAX_PATIENT_DISEASE];// declare the character array for disease
} s_PatientInfo;
//Own version of fgetsRemovedNewLine
void fgetsRemovedNewLine(char * restrict buf, int n,FILE * restrict stream)
{
    if (fgets(buf, n, stream) == NULL)
    {
```



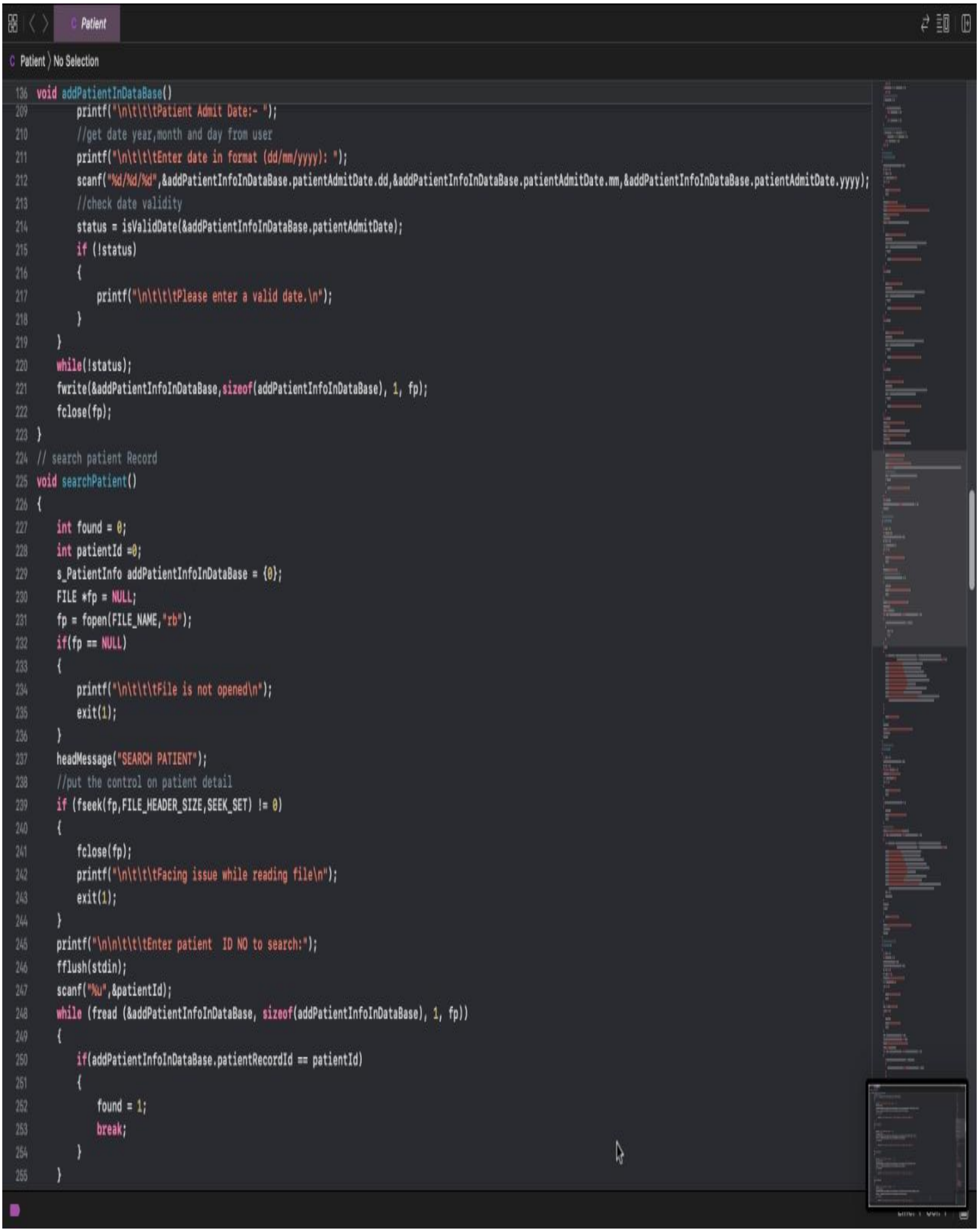
```c
int isNameValid(const char *name)
        {
            validName = 0;
            break;
        }
    }
    return validName;
}
// Function to check leap year.
//Function returns 1 if leap year
int  IsLeapYear(int year)
{
    return (((year % 4 == 0) &&
             (year % 100 != 0)) ||
            (year % 400 == 0));
}
// returns 1 if given date is valid.
int isValidDate(Date *validDate)
{
    //check range of year,month and day
    if (validDate->yyyy > MAX_YR ||
            validDate->yyyy < MIN_YR)
        return 0;
    if (validDate->mm < 1 || validDate->mm > 12)
        return 0;
    if (validDate->dd < 1 || validDate->dd > 31)
        return 0;
    //Handle feb days in leap year
    if (validDate->mm == 2)
    {
        if (IsLeapYear(validDate->yyyy))
            return (validDate->dd <= 29);
        else
            return (validDate->dd <= 28);
    }
    //handle months which has only 30 days
    if (validDate->mm == 4 || validDate->mm == 6 ||
            validDate->mm == 9 || validDate->mm == 11)
        return (validDate->dd <= 30);
    return 1;
}
// Add patient in list
void addPatientInDataBase()
{
    s_PatientInfo addPatientInfoInDataBase = {0};
    FILE *fp = NULL;
    int status = 0;
    fp = fopen(FILE_NAME,"ab+");
    if(fp == NULL)
```



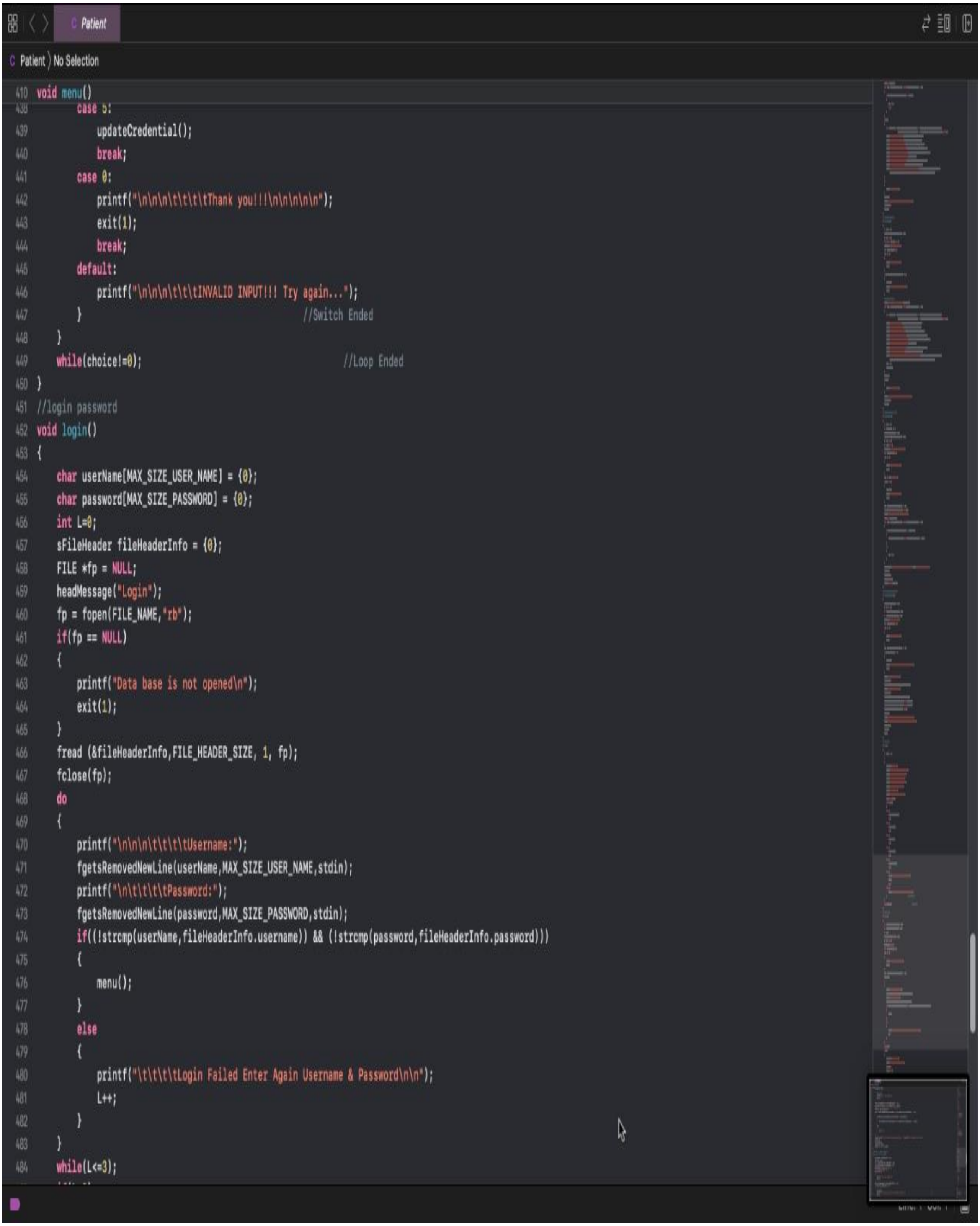
```c
void fgetsRemovedNewLine(char * restrict buf, int n,FILE * restrict stream)
        printf("Fail to read the input stream");
    }
    else
    {
        buf[strcspn(buf, "\n")] = '\0';
    }
}
//Align the message
void printMessageCenter(const char* message)
{
    int len =0;
    int pos = 0;
    //calculate how many space need to print
    len = (78 - strlen(message))/2;
    printf("\t\t\t");
    for(pos =0 ; pos < len ; pos++)
    {
        //print space
        printf(" ");
    }
    //print message
    printf("%s",message);
}
//Head message
void headMessage(const char *message)
{
    system("cls");
    printf("\t\t\t###########################################################################");
    printf("\n\t\t\t############                                                   ############");
    printf("\n\t\t\t############   Patient Record Management System Project in C   ############");
    printf("\n\t\t\t############                                                   ############");
    printf("\n\t\t\t###########################################################################");
    printf("\n\t\t\t---------------------------------------------------------------------------\n");
    printMessageCenter(message);
    printf("\n\t\t\t----------------------------------------------------------------------------");
}
//Validate name
int isNameValid(const char *name)
{
    int validName = 1;
    int len = 0;
    int index = 0;
    len = strlen(name);
    for(index =0; index <len ; ++index)
    {
        if(!(isalpha(name[index])) && (name[index] != '\n') && (name[index] != ' '))
        {
```



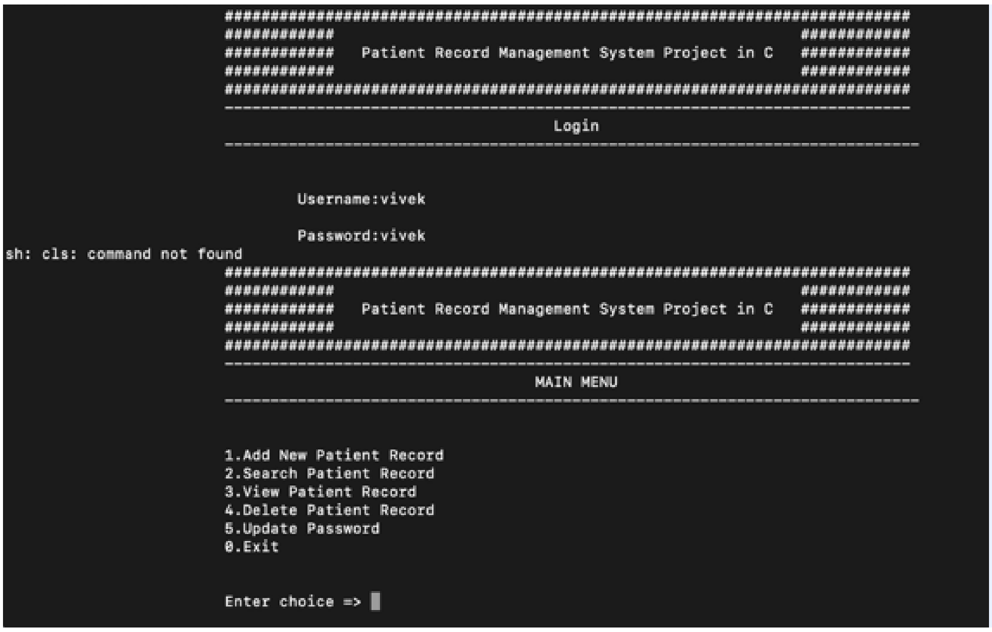
```c
void viewPatient()
    FILE *fp = NULL;
    unsigned int countPatient = 1;
    headMessage("VIEW PATIENT DETAILS");
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("File is not opened\n");
        exit(1);
    }
    if (fseek(fp,FILE_HEADER_SIZE,SEEK_SET) != 0)
    {
        fclose(fp);
        printf("Facing issue while reading file\n");
        exit(1);
    }
    //Print patient  count
    printf("\n\t\t\tPatient Count = %d\n\n",countPatient);
    while (fread (&addPatientInfoInDataBase, sizeof(addPatientInfoInDataBase), 1, fp))
    {
        float remainingAmount = (addPatientInfoInDataBase.patientTotalFees >= addPatientInfoInDataBase.patientDepositMoney)?
                                (addPatientInfoInDataBase.patientTotalFees - addPatientInfoInDataBase.patientDepositMoney):(float)(0.0);
        printf("\t\t\tPatient id = %u\n",addPatientInfoInDataBase.patientRecordId);
        printf("\t\t\tPatient Name = %s\n",addPatientInfoInDataBase.patientName);
        printf("\n\t\t\tPatient Disease = %s\n",addPatientInfoInDataBase.patientDisease);
        printf("\t\t\tPatient Total Charge = %f\n",addPatientInfoInDataBase.patientTotalFees);
        printf("\t\t\tPatient Deposit Amount = %f\n",addPatientInfoInDataBase.patientDepositMoney);
        printf("\t\t\tPatient Remaining Amount = %f\n",remainingAmount);
        printf("\t\t\tPatient Father Name = %s\n",addPatientInfoInDataBase.patientFatherName);
        printf("\t\t\tPatient Address = %s\n",addPatientInfoInDataBase.patientAddr);
        printf("\t\t\tPatient Admited Date(day/month/year) =  (%d/%d/%d)\n\n",addPatientInfoInDataBase.patientAdmitDate.dd,
               addPatientInfoInDataBase.patientAdmitDate.mm, addPatientInfoInDataBase.patientAdmitDate.yyyy);
        found = 1;
        ++countPatient;
    }
    fclose(fp);
    if(!found)
    {
        printf("\n\t\t\tNo Record");
    }
    printf("\n\n\t\t\tPress any key to go to main menu.....");
    fflush(stdin);
    getchar();
}
// Delete patient Record entry
void deletePatient()
{
    int found = 0;
```



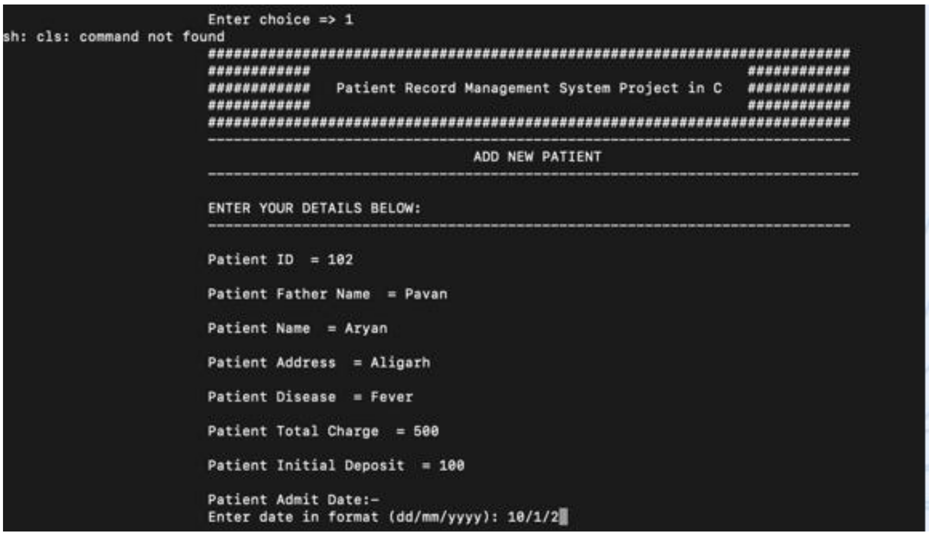
```
void addPatientInDataBase()
        printf("\n\t\t\tPatient Admit Date:- ");
        //get date year,month and day from user
        printf("\n\t\t\tEnter date in format (dd/mm/yyyy): ");
        scanf("%d/%d/%d",&addPatientInfoInDataBase.patientAdmitDate.dd,&addPatientInfoInDataBase.patientAdmitDate.mm,&addPatientInfoInDataBase.patientAdmitDate.yyyy);
        //check date validity
        status = isValidDate(&addPatientInfoInDataBase.patientAdmitDate);
        if (!status)
        {
            printf("\n\t\t\tPlease enter a valid date.\n");
        }
    }
    while(!status);
    fwrite(&addPatientInfoInDataBase,sizeof(addPatientInfoInDataBase), 1, fp);
    fclose(fp);
}
// search patient Record
void searchPatient()
{
    int found = 0;
    int patientId =0;
    s_PatientInfo addPatientInfoInDataBase = {0};
    FILE *fp = NULL;
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("\n\t\t\tFile is not opened\n");
        exit(1);
    }
    headMessage("SEARCH PATIENT");
    //put the control on patient detail
    if (fseek(fp,FILE_HEADER_SIZE,SEEK_SET) != 0)
    {
        fclose(fp);
        printf("\n\t\t\tFacing issue while reading file\n");
        exit(1);
    }
    printf("\n\n\t\t\tEnter patient  ID NO to search:");
    fflush(stdin);
    scanf("%u",&patientId);
    while (fread (&addPatientInfoInDataBase, sizeof(addPatientInfoInDataBase), 1, fp))
    {
        if(addPatientInfoInDataBase.patientRecordId == patientId)
        {
            found = 1;
            break;
        }
    }
```



```c
void menu()
        case 5:
            updateCredential();
            break;
        case 0:
            printf("\n\n\n\t\t\t\tThank you!!!\n\n\n\n\n");
            exit(1);
            break;
        default:
            printf("\n\n\n\t\t\tINVALID INPUT!!! Try again...");
        }                                             //Switch Ended
    }
    while(choice!=0);                                     //Loop Ended
}
//login password
void login()
{
    char userName[MAX_SIZE_USER_NAME] = {0};
    char password[MAX_SIZE_PASSWORD] = {0};
    int L=0;
    sFileHeader fileHeaderInfo = {0};
    FILE *fp = NULL;
    headMessage("Login");
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("Data base is not opened\n");
        exit(1);
    }
    fread (&fileHeaderInfo,FILE_HEADER_SIZE, 1, fp);
    fclose(fp);
    do
    {
        printf("\n\n\n\t\t\t\tUsername:");
        fgetsRemovedNewLine(userName,MAX_SIZE_USER_NAME,stdin);
        printf("\n\t\t\t\tPassword:");
        fgetsRemovedNewLine(password,MAX_SIZE_PASSWORD,stdin);
        if((!strcmp(userName,fileHeaderInfo.username)) && (!strcmp(password,fileHeaderInfo.password)))
        {
            menu();
        }
        else
        {
            printf("\t\t\t\tLogin Failed Enter Again Username & Password\n\n");
            L++;
        }
    }
    while(L<=3);
```



```
                        ###########################################################################
                        ############                                                   ############
                        ############   Patient Record Management System Project in C   ############
                        ############                                                   ############
                        ###########################################################################
                        ---------------------------------------------------------------------------
                                                            Login
                        ----------------------------------------------------------------------------


                                Username:vivek

                                Password:vivek
sh: cls: command not found
                        ###########################################################################
                        ############                                                   ############
                        ############   Patient Record Management System Project in C   ############
                        ############                                                   ############
                        ###########################################################################
                        ---------------------------------------------------------------------------
                                                          MAIN MENU
                        ----------------------------------------------------------------------------


                        1.Add New Patient Record
                        2.Search Patient Record
                        3.View Patient Record
                        4.Delete Patient Record
                        5.Update Password
                        0.Exit


                        Enter choice => 
```



```
                        Enter choice => 1
sh: cls: command not found
                        ###########################################################################
                        ############                                                   ############
                        ############   Patient Record Management System Project in C   ############
                        ############                                                   ############
                        ###########################################################################
                        ---------------------------------------------------------------------------
                                                       ADD NEW PATIENT
                        ----------------------------------------------------------------------------

                        ENTER YOUR DETAILS BELOW:
                        ---------------------------------------------------------------------------

                        Patient ID  = 102

                        Patient Father Name  = Pavan

                        Patient Name  = Aryan

                        Patient Address  = Aligarh

                        Patient Disease  = Fever

                        Patient Total Charge  = 500

                        Patient Initial Deposit  = 100

                        Patient Admit Date:-
                        Enter date in format (dd/mm/yyyy): 10/1/2
```

REFRENCE:

1. https://www.youtube.com/watch?v=udA9p-WFE1k