**KATHFORD INTERNATIONAL COLLEGE OF ENGINEERING AND MANAGEMENT**

**Affiliated to Tribhuvan University**

**Institute of Science and Technology**

**Project on**

**“STUDENT MANAGEMENT SYSTEM”**

**A PROJECT REPORT**

**Submitted to**

**Department of Computer Science and Information Technology**

**Kathford International College of Engineering and Management**

***In Partial fulfilment for the first semester course “Fundamental of Computer Programming (CSC-102)” of Bachelor of Science in Computer Science and Information Technology***

***(B.Sc. CSIT)***

***Under the Supervision of***

***Miss Deni Shahi***

Submitted by

Kushal Kc (Roll no. 17)

Mala Thapa Magar (Roll no. 19)

Namkong Hang Kirat (Roll no. 23)

Neha Adhikari (Roll no. 24)

April, 2016

**Kathford International College of Engineering and Management Balkumari, Lalitpur**

# **LETTER OF APPROVAL**

This to certify that this project prepared by Kushal Kc, Mala Thapa Magar, Namkong Hang Kirat, Neha adhikari entitled “**STUDENT MANAGEMENT SYSTEM”** in partial fulfilment for the first semester course **Fundamental of Computer Programming (CSC-102)** of B.Sc. in Computer Science and Information Technology has been well studied. In our opinion it is satisfactory in the scope and quality as a project for the required subject.

Ms. Deni Shahi Mr. Sushant Poudel

CSIT department HoD, CSIT Department

Kathford International College Kathford International College

Of Engineering and Management of Engineering and Management

Balkumari, Lalitpur Balkumari, Lalitpur

External examiner Internal examiner

**Kathford International College of Engineering and Management Balkumari, Lalitpur**

# **Supervisor’s Recommendation**

I hereby recommend that this project prepared under my supervision by Kushal Kc, Mala Thapa Magar, Namkong Hang Kirat, Neha Adhikari entitled “**STUDENT MANAGEMENT SYSTEM”** in partial fulfilment for the first semester course **Fundamental of Computer Programming (CSC-102)** of B.Sc. in Computer Science and Information Technology be processed for evaluation.

Miss Deni Shahi

CSIT department

Kathford International College of

Engineering and Management Balkumari, Lalitpur

# **ABSTRACT**

This project **“STUDENT MANAGEMENT SYSTEM”** allows authorized members to access the record of academically registered students. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming. All these problems are solved using this project.

It can be used in various educational institutes across the globe and simplifies working of institutes. This project helps in maintaining the database of the students in any educational organization. We can easily access any student’s information any time and can be kept safely for long period of time without damage.

**ACKNOWLEDGEMENT**

We take this opportunity to express our sincere gratitude to all those who helped us in various capacities in undertaking this project and devising the report. We are privileged to express our sense of gratitude to our respected teacher **Miss Deni Sahi** whose unparalleled knowledge and judgement along with his know-how, was an immense support in completing the project. We are also grateful to **Mr. Sushant Paudel**, the Head of Department, Information Technology, for the brainwave and encouragement given. We take this opportunity also to thank our friends and contemporaries for their co-operation and compliance.

We would also like to thank our C Programming Teacher **Mr. Mekh Raj Jaishi** for his unfailing cooperation and sparing his valuable time to assist us in our project.

**CONTENTS**

Introduction…………………………………………………………1

Problem statement…………………………………………………..1

Objectives…………………………………………………………...1

Scopes and Limitations……………………………………………...2

Implementation………………………………………………...…....2-6

Conclusion and enhancement………………………………………..7

Appendices………………..…………………………………............8-30

1. **INTRODUCTION**

This project **Student management system** is an information management system for education establishments to manage student data. It provide capabilities for registering students and manage student-related data needs in a school.

# **PROBLEM STATEMENT**

Managing students record is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming. All these problems are solved using this project. Throughout the project the focus has been on presenting information in an easy and intelligible manner. The project is very useful for those who want to know about Student Management Systems and want to develop software based on the same concept. Only authorized users ae allowed to enter through the system. So security is also maintained in the system. It reduces paperwork in an educational institution.

1. **OBJECTIVES**

* To allow the administrator to find and edit the personal details of a student and to also allow the student to update his/her profile.
* To keep all the records of students, such as their id, name, mailing address, phone number, DOB etc.
* To help any organization to maintain its student's personal data.

**1**

1. **SCOPES AND LIMITATIONS**

The Student Management System can be enhanced to include some other functionality like marks, attendance management. Talent management of students based on their performance evaluation can be added. Social networking can also be added wherein students can interact with each other. Online class functionality can be added. Can evolve as an online institution. Functionality of chat and messages can be added. Online exam functionality can be added. Online resume builder functionality can also be added.

1. **IMPLEMENTATION**

Student Management System is a simple console application without graphics, developed using C programming language. This mini project utilizes various aspects of the C language such as functions, arrays, pointers, file handling, and data structure.

1. **Algorithm**

Step 1: Start

Step 2: Input the choice (1-6)

Step 3: Switch choice

* choice==1?

Print add record

Open a file \*fp="record.txt"

Is fp==NULL?

* if yes, print error to open file
* if no, input stu.id, stu.name, stu.address, stu.parname, stu.phone\_no

**2**

* choice==2?

Print search record

Input s\_id

Open "record.txt"

* Is s\_id==stu.id?

If yes, print record found

Print stu.id, stu.name, stu.address, stu.parname, stu.phone\_no

If no, print no record found

* choice==3?

If yes, print modify record

Open "record.txt"

Input s\_id

* Is s\_id==stu.id?

If yes, input stu.id, stu.name, stu.address, stu.parname, stu.phone\_no

If no, print no record found

* choice==4?

Print delete record

Open "record.txt “

Input s\_id

* Is s\_id==stu.id?

If yes, delete record

* choice==5?

goto step 5

Step 4: go to step 2

Step 5: Stop

**3**

1. **Flowchart**

choice ==2

choice ==1

Start

Input choice (1-6)

Switch choice

Add record

Search record

choice ==3

Modify record

Delete record

choice ==4

choice ==5

Stop

default

**4**

1. **Header file used**

A header file is generally used to define all of the functions, variables and constants contained in any function library might be used. Various header files are to be used in order to make this program usable.

* #include <stdio.h>: We have used printf(), scanf(), fflush(stdin) and putchar() predefined functions under this header file.
* #include <conio.h>: We have used getch() and gets() predefined functions under this header file.
* #include <string.h>: We have used strcmp() and strlen() predefined functions under this header file.
* #include <stdlib.h>: We have used exit() and system(“cls”) predefined functions under this header file.
* #include <ctype.h>: We have used isdigit() and isalpha() predefined functions under this header file.

1. **Control statements used**

* if statement: We have used if statement in adding, searching, modifying and deleting records.
* Switch statement: We have used switch statement to make our program menu driven.

1. **Loops used**

* For loop: We have used this loop in adding and modifying records.
* While loop: We have used this loop in searching, modifying and deleting records.
* Goto: We have used this loop in adding and modifying records.

**5**

1. **Arrays used**

iD[20]: To allocate the space for students’ id.

fname[20]: To allocate the space for students’ first name.

mname[20]: To allocate the space for students’ middle name.

lname[20]: To allocate the space for students’ last name.

add[20]: To allocate the space for students’ address.

fatname[20]: To allocate the space for students’ father name.

motname[20]: To allocate the space for students’ mother name.

mphone[15]: To allocate the space for mobile number.

hphoneac[15]: To allocate the space for area code of home telephone.

hphonen[15]: To allocate the space for telephone number.

1. **User defined functions used**

* add\_student(): To add records of student.
* search\_student():To search records of student.
* mod\_student():To modify records of student.
* delete\_student():To delete records of student.
* printChar(): To print the character for required number of times.
* printHead(): To print the heading of the program.

1. **File handling functions**

* fopen(): To open the file “record.txt” in ab+, rb and rb+ mode and another file “temp.txt” in wb mode.
* fclose(): To close the files “record.txt” and “temp.txt”.
* fread(): To read data from the files “record.txt” and “temp.txt”.
* fwrite(): To write data in the files “record.txt” and “temp.txt”.
* fseek(): We have used SEEK\_CUR position under this function to move the cursor in the file to the desired position from current position.
* remove(): To delete file “record.txt” in case of deleting record.
* rename(): To rename the file “temp.txt” to “record.txt” in case of deleting record.

**6**

1. **CONCLUSION AND ENHANCEMENT**

**Conclusion**

* Helpful to perform paperless work and manage all data.
* Provides easy, accurate, unambiguous and faster data access.

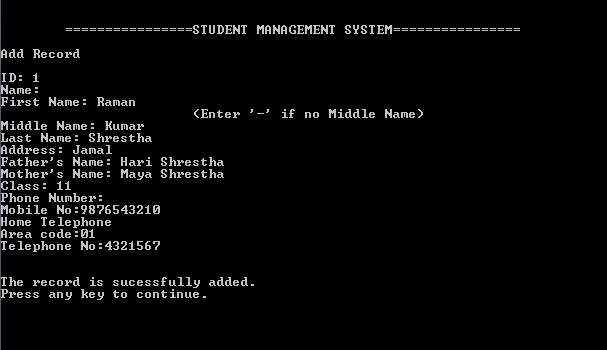
**Enhancement**

Student management system can be further enhanced to add other functionality also. Some of them are listed below.

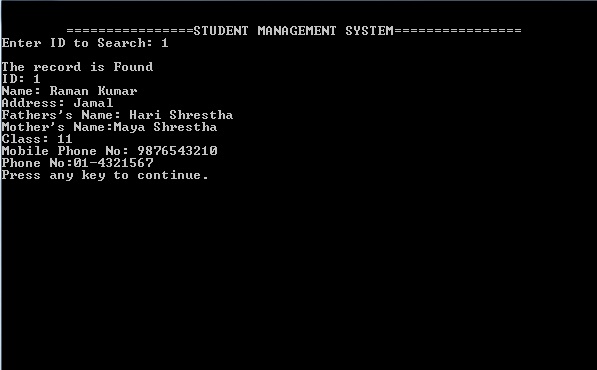
* Mark management
* Attendance management
* Assignment record

**7**

1. **APPENDICES**

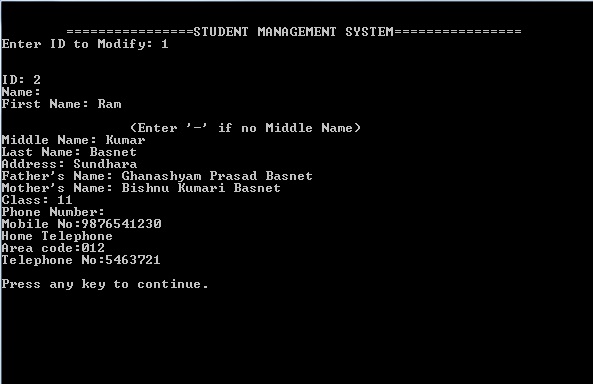
****

**Figure: Adding record**

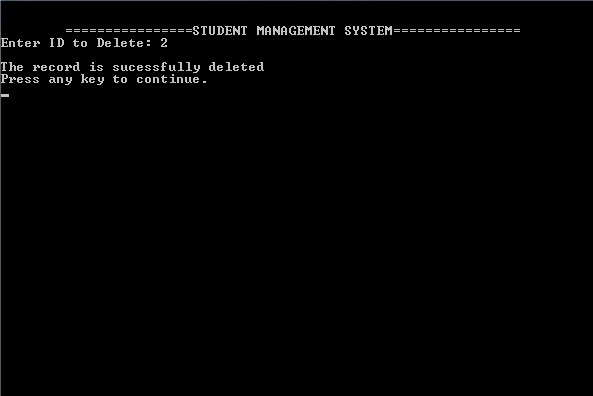
****

**Figure: Searching record**

**8**

****

**Figure: Modifying record**

****

**Figure: Deleting record**

**9**

**Source code**

1. **Adding records:**

void add\_student()

{

printHead();

int i;

printf("\nAdd Record\n");

FILE \*fp;

fp = fopen("record.txt","ab+");

if(fp == NULL)

{

printf("Error in Opening file\nMake sure your file is not write protected");

exit(1);

}

fflush(stdin);

printf("\nID: ");

gets(stu.ID);

printf("Name:\n");

fname:

printf("First Name: ");gets(stu.fname);

for(i=0;i<strlen(stu.fname);i++)

{

if(isdigit(stu.fname[i]))

**10**

{

printf("\n\t\tEnter valid name with characters.\n");

goto fname;

}

else if(isalpha(stu.fname[i]))

{

}

else

{

printf("\n\t\tEnter valid First Name with characters.\n");

goto fname;

}

}

mname:

printf("\t\t\t(Enter '-' if no Middle Name)\nMiddle Name: ");gets(stu.mname);

if(stu.mname[0]!='-')

{

for(i=0;i<strlen(stu.mname);i++)

{

if(isdigit(stu.mname[i]))

{

printf("\n\t\tEnter valid name with characters.\n");

goto mname;}

**11**

else if(isalpha(stu.mname[i]))

{

}

else

{

printf("\n\t\tEnter valid Middle Name with characters.\n");

goto mname;

}

}

}

else if(stu.mname[0]=='-')

{

}

lname:

printf("Last Name: ");gets(stu.lname);

for(i=0;i<strlen(stu.lname);i++)

{

if(isdigit(stu.lname[i]))

{

printf("\n\t\tEnter valid Last Name with characters.\n");

goto lname;

}

**12**

else if(isalpha(stu.lname[i]))

{

}

else

{

printf("\n\t\tEnter valid Last Name with characters.\n");

goto lname;

}

}

printf("Address: ");gets(stu.add);

printf("Father's Name: ");gets(stu.fatname);

printf("Mother's Name: ");gets(stu.motname);

grade:

printf("Class: ");scanf("%d",&stu.Class);

if(stu.Class>12||stu.Class<1)

{

printf("Enter Valid Class");

goto grade;

}

else

{

}

printf("Phone Number: ");

**13**

mobileno:

printf("\nMobile No:");

fflush(stdin);

gets(stu.mphone);

if(strlen(stu.mphone)==10 && stu.mphone[0]=='9')

{

for(i=0;i<strlen(stu.mphone);i++)

{

if(isalpha(stu.mphone[i]))

{

printf("\n\t\tEnter valid Mobile Number.\n");

goto mobileno;

}

else if(isdigit(stu.mphone[i]))

{

}

else

{

printf("\n\t\tEnter valid Mobile Number.\n");

goto mobileno;

}

}

}

**14**

else

{

printf("\n\t\tEnter valid Mobile Number.\n");

goto mobileno;

}

printf("Home Telephone\n");

areacode:

printf("Area code:");

fflush(stdin);

scanf("%s",&stu.hphoneac);

if(strlen(stu.hphoneac)==3 && stu.hphoneac[0]=='0')

{

for(i=0;i<strlen(stu.hphoneac);i++)

{

if(isalpha(stu.hphoneac[i]))

{

printf("\n\t\tEnter valid Area Code.\n");

goto areacode;

}

else if(isdigit(stu.hphoneac[i]))

{

}

**15**

else

{

printf("\n\t\tEnter valid Area Code.\n");

goto areacode;

}

}

}

else if(strlen(stu.hphoneac)==2)

{

if(stu.hphoneac[0]=='0' && stu.hphoneac[1]=='1')

{

}

else

{

printf("\n\t\tEnter valid Area Code.\n");

goto areacode;

}

}

else if(strlen(stu.hphoneac)!=3 || strlen(stu.hphoneac)!=2)

{

printf("\n\t\tEnter valid Area Code.\n");

goto areacode;

}

**16**

ghphonen:

printf("Telephone No:");

fflush(stdin);

scanf("%s",&stu.hphonen);

if(strlen(stu.hphonen)==7)

{

for(i=0;i<strlen(stu.hphonen);i++)

{

if(isalpha(stu.hphonen[i]))

{

printf("\n\t\tEnter valid Telephone Number.\n");

goto ghphonen;

}

else if(isdigit(stu.hphonen[i]))

{

}

else

{

printf("\n\t\tEnter valid Telephone Numbers.\n");

goto ghphonen;

}

}

}

**17**

else if(strlen(stu.hphonen)!=7)

{

printf("\n\t\tEnter valid Telephone Numbers.\n");

goto ghphonen;

}

fwrite(&stu, sizeof(stu), 1, fp);

printf("\n\nThe record is sucessfully added.");

printf("\nPress any key to continue.\n");

getch();

printHead();

fclose(fp);

}

1. **Searching records:**

void search\_student(){

printHead();

char s\_id[15];

int isFound = 0;

printf("Enter ID to Search: ");

fflush(stdin);

gets(s\_id);

FILE \*fp;

fp = fopen("record.txt","rb");

while(fread(&stu,sizeof(stu),1,fp) == 1){

18

if(strcmp(s\_id,stu.ID) == 0){

isFound = 1;

break;

}

}

if(isFound == 1)

{

printf("\nThe record is Found");

printf("\nID: %s",stu.ID);

printf("\nName: %s %s %S",stu.fname,stu.mname,stu.lname);

printf("\nAddress: %s",stu.add);

printf("\nFathers's Name: %s",stu.fatname);

printf("\nMother's Name:%s",stu.motname);

printf("\nClass: %d",stu.Class);

printf("\nMobile Phone No: %s",stu.mphone);

printf("\nPhone No:%s-%s",stu.hphoneac,stu.hphonen);

}

else

{

printf("\nSory, No record found in the database.\n");

}

fclose(fp);

printf("\nPress any key to continue.\n");

**19**

getch();

printHead();

return;

}

1. **Modifying records:**

void mod\_student(){

printHead();

char s\_id[15];

int i;

int isFound = 0, print = 37;

printf("Enter ID to Modify: ");

fflush(stdin);

gets(s\_id);

FILE \*fp;

fp = fopen("record.txt","rb+");

while(fread(&stu, sizeof(stu),1,fp) == 1){

if(strcmp(s\_id, stu.ID) == 0){

fflush(stdin);

printf("\n\nID: ");

gets(stu.ID);

printf("Name:\n");

fname:

**20**

printf("First Name: ");gets(stu.fname);

for(i=0;i<strlen(stu.fname);i++)

{

if(isdigit(stu.fname[i]))

{

printf("\n\t\tEnter valid name with characters.\n");

goto fname;

}

else if(isalpha(stu.fname[i]))

{

}

else

{

printf("\n\t\tEnter valid First Name with characters.\n");

goto fname;

}

}

mname:

printf("\n\t\t(Enter '-' if no Middle Name)\nMiddle Name: ");gets(stu.mname);

if(stu.mname[0]!='-')

{

for(i=0;i<strlen(stu.mname);i++)

{

**21**

if(isdigit(stu.mname[i]))

{

printf("\n\t\tEnter valid name with characters.\n");

goto mname;

}

else if(isalpha(stu.mname[i]))

{

}

else

{

printf("\n\t\tEnter valid Middle Name with characters.\n");

goto mname;

}

}

}

else if(stu.mname[0]=='-')

{

}

lname:

printf("Last Name: ");gets(stu.lname);

for(i=0;i<strlen(stu.lname);i++)

{

if(isdigit(stu.lname[i]))

**22**

{

printf("\n\t\tEnter valid Last Name with characters.\n");

goto lname;

}

else if(isalpha(stu.lname[i]))

{

}

else

{

printf("\n\t\tEnter valid Last Name with characters.\n");

goto lname;

}

}

printf("Address: ");gets(stu.add);

printf("Father's Name: ");gets(stu.fatname);

printf("Mother's Name: ");gets(stu.motname);

grade:

printf("Class: ");scanf("%d",&stu.Class);

if(stu.Class>12||stu.Class<1)

{

printf("Enter Valid Class");

goto grade;

}

**23**

else

{

}

printf("Phone Number: ");

mobileno:

printf("\nMobile No:");

fflush(stdin);

gets(stu.mphone);

if(strlen(stu.mphone)==10 && stu.mphone[0]=='9')

{

for(i=0;i<strlen(stu.mphone);i++)

{

if(isalpha(stu.mphone[i]))

{

printf("\n\t\tEnter valid Mobile Number.\n");

goto mobileno;

}

else if(isdigit(stu.mphone[i]))

{

}

else

{

printf("\n\t\tEnter valid Mobile Number.\n");

**24**

goto mobileno;

}

}

}

else

{

printf("\n\t\tEnter valid Mobile Number.\n");

goto mobileno;

}

printf("Home Telephone\n");

areacode:

printf("Area code:");

fflush(stdin);

scanf("%s",&stu.hphoneac);

if(strlen(stu.hphoneac)==3 && stu.hphoneac[0]=='0')

{

for(i=0;i<strlen(stu.hphoneac);i++)

{

if(isalpha(stu.hphoneac[i]))

{

printf("\n\t\tEnter valid Area Code.\n");

goto areacode;

}

**25**

else if(isdigit(stu.hphoneac[i]))

{

}

else

{

printf("\n\t\tEnter valid Area Code.\n");

goto areacode;

}

}

}

else if(strlen(stu.hphoneac)==2)

{

if(stu.hphoneac[0]=='0' && stu.hphoneac[1]=='1')

{

}

else

{

printf("\n\t\tEnter valid Area Code.\n");

goto areacode;

}

}

else if(strlen(stu.hphoneac)!=3 || strlen(stu.hphoneac)!=2)

{

**26**

printf("\n\t\tEnter valid Area Code.\n");

goto areacode;

}

ghphonen:

printf("Telephone No:");

fflush(stdin);

scanf("%s",&stu.hphonen);

if(strlen(stu.hphonen)==7)

{

for(i=0;i<strlen(stu.hphonen);i++)

{

if(isalpha(stu.hphonen[i]))

{

printf("\n\t\tEnter valid Telephone Number.\n");

goto ghphonen;

}

else if(isdigit(stu.hphonen[i]))

{

}

else

{

printf("\n\t\tEnter valid Telephone Numbers.\n");

goto ghphonen;

**27**

}

}

}

else if(strlen(stu.hphonen)!=7)

{

printf("\n\t\tEnter valid Telephone Numbers.\n");

goto ghphonen;

}

fseek(fp,-sizeof(stu), SEEK\_CUR);

fwrite(&stu,sizeof(stu), 1, fp);

isFound = 1;

break;

}

printf("Student Record has been modified.");

}

if(!isFound){

printf("\nNo Record Found\n");

}

printf("\nPress any key to continue.\n");

getch();

fclose(fp);

printHead();

return;}

**28**

1. **Deleting record:**

void delete\_student()

{

printHead();

char s\_id[15];

int isFound = 0, print = 37;

printf("Enter ID to Delete: ");

fflush(stdin);

gets(s\_id);

FILE \*fp, \*temp;

fp = fopen("record.txt","rb");

temp = fopen("temp.txt", "wb");

while(fread(&stu, sizeof(stu),1,fp) == 1)

{

if(strcmp(s\_id, stu.ID) == 0)

{

fwrite(&stu,sizeof(stu),1,temp);

}

}

fclose(fp);

fclose(temp);

remove("record.txt");

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

**29**

printf("\nThe record is sucessfully deleted");

printf("\nPress any key to continue.\n");

getch();

printHead();

return;

}

**30**

**REFERENCE AND BIBLIOGRAPHY**

**Reference**

* E Balaguruswamy, Programming in ANSI C, 5e, Tata McGraw-Hill Education Private Limited, 2011.
* Sah Satyendra, C programming, Om Publications, first edition, 2005.

**Bibliography**

* <http://www.slideshare.net/rajsharma528/projectreportonstudentinformationmanagementsystemphpmysql>
* [www.slideshare.net](http://www.slideshare.net)
* [www.google.com](http://www.google.com)
* [www.wikipedia.com](http://www.wikipedia.com)