



SRM University AP
OBJECT ORIENTED PROGRAMMING
with C++
Group - 9

SRM STUDENT REPORT MANAGEMENT SYSTEM”

submitted in partial fulfilment for the award of the degree in

BACHELOR OF TECHNOLOGY
IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

<u>K.Ajay Kumar</u>	<u>AP21110010198</u>
<u>Sk.Abdul Basheer</u>	<u>AP21110010203</u>
<u>Y.Baladithya</u>	<u>AP21110010210</u>
<u>T.Gnana Kartheek</u>	<u>AP21110010238</u>
<u>P.Sri Veda Aryan</u>	<u>AP21110010251</u>
<u>Ch.Bhargav</u>	<u>AP21110010252</u>

Under the guidance of: L.Srinivasa Rao

Project Mentor and Professor

ACKNOWLEDGEMENT

We would like to thank the SRM AP faculty for giving us the opportunity to work on a project. A group project can never be done by a single person. So, we appreciate our whole hearted help and support from our group members in completing our project. Successful completion of a project requires guidance and help from a number of people. Hence, we give our sincere thanks to L.Srinivasa Rao our mentor and professor for his special concern and for providing sufficient information related to the topic, which helped us in completing the project work on time.

CONTENTS

Chapter	<i>Title</i>	<i>Page No.</i>
Chapter 1	Objectives	4
Chapter 2	Technologies Used: - ▪ Hardware Requirement ▪ Software Requirement	5
Chapter 3	Abstract	6
Chapter 4	Introduction	7
Chapter 5	Project Modules	8
Chapter 6	Algorithm	9
Chapter 7	Sample Input/Output	10-18
Chapter 8	Conclusion	19

CHAPTER 1:

OBJECTIVE

The main objective of our project on “SRM STUDENT REPORT MANAGEMENT SYSTEM” is to

- >store marks of various students
- >display and manage their reports

The project aims to depict an actual Student Report Management System. The program reduces all the manual work, time & energy required to manage reports for each student at any educational institution.

CHAPTER 2:

TECHNOLOGIES USED

HARDWARE REQUIREMENTS:

Hard Disk: 1 GB or above

RAM: 512 MB

Processor: Intel Pentium or above

SOFTWARE REQUIREMENTS:

Language used: C++

Operating System: Windows XP or above

Compiler Compatibility: GDB Online Compiler
/CodeChef/HackerRank/Code Blocks/Visual Studios

CHAPTER 3:

ABSTRACT

Student report Management System is a C++ project which can be used to perform various operations and store the mark records of students without using SQL(database) i.e. only using C++.

We can perform the following 5 main operations:

1. Addition of New student record.
2. Modify a student record
3. Delete a student record
4. Display a particular student record
5. Display all student records

CHAPTER 4

INTRODUCTION

Student Management System is software which is helpful for students as well as the school authorities. In the current system all the activities are done manually. It is very time consuming and costly. Our Student Management System deals with the various activities related to the students.

In general, in a student report management system software we can register as a user and user has of two types, student and administrator. Administrator has the power to add new user and can edit and delete a user. A student can register as user and can add edit and delete his profile. The administrator can add edit and delete marks for the student. All the users can see the marks. But in our project we don't have two types of users. Every user can perform all the operations. This is a limitation.

CHAPTER 5:

PROJECT MODULES

Our project mainly consists of 2 modules which further contain sub modules.

They are as follows:

1*RESULT MENU :

- >Class Result
- >Student Report Card

2*ENTRY/EDIT MENU:

- >CREATE STUDENT RECORD
- >DISPLAY ALL STUDENTS RECORDS
- >SEARCH STUDENT RECORD
- >MODIFY STUDENT RECORD
- >DELETE STUDENT RECORD

CHAPTER 6:

ALGORITHM

Step1.Start

Step2.Show the welcome screen and ask the user to press Enter to continue.

Step 3. Ask the user to select any option

1*Result Menu 2*Entry/Edit Menu

3*Exit

Step 3. If the user opts for “Result Menu” ask the user to select between

1.Class Result 2. Student Report Card 3. Back to Main Menu

Step 4. If user opts for Class Result display the results of all the students in the class. If the user opts for Student Report card, ask the user for the student roll number and display his report card.

Step 5. If the user opts for “Entry/Edit Menu” ask the user to select between

1.CREATE STUDENT RECORD 2.DISPLAY ALL STUDENTS RECORDS

3.SEARCH STUDENT RECORD 4.MODIFY STUDENT RECORD

5.DELETE STUDENT RECORD 6.BACK TO MAIN MENU

Step 6. If the user inputs either 1 or 3 or 4 or 5, ask the user for roll number and perform the selected operation. If the user opts for 2, display all the student records.

Step 7. If the user opts for Exit save the information and exit.

Step 8. End.

CHAPTER 7:

CODE & SAMPLE INPUT/OUTPUT

```
#include<bits/stdc++.h>
using namespace std;

class Student
{
    char Name[50];
    int Roll_no;
    int py_marks, iscp_marks, cpp_marks, cpplab_marks, eco_marks, de_marks, delab_marks,
    daa_marks, daalab_marks, mat_marks, ises_marks;
    double Percentage;
    char Grade;
    void Calculate_Grade();
public:
    void getdata();
    void showdata() const;
    void show_tabular() const;
    int retrollno() const;
};

void Student::Calculate_Grade()
{
    Percentage=(py_marks+iscp_marks+cpp_marks+cpplab_marks+eco_marks+de_marks+delab_
marks+daa_marks+daalab_marks+mat_marks+ises_marks)/ 11.0;
    if(Percentage>=90)
        Grade='A';
    else if(Percentage>=80 && Percentage<90)
        Grade='B';
    else if(Percentage>=70 && Percentage<80)
        Grade='C';
    else if(Percentage>=60 && Percentage<70)
        Grade='D';
    else if(Percentage>=50 && Percentage<60)
        Grade='E';
    else
        Grade='F';
}

void Student::getdata()
{
    cout<<"/tSTUDENT DETAILS"<<endl<<endl;
    cout<<"Roll Number :"<<endl;
    cin>>Roll_no;
    cout<<"\nName :";
    cin.ignore();
    cin.getline(Name,50);
```

```

cout<<"\nPython Marks(out of 100) :";
cin>>py_marks;
cout<<"\nISCP Marks :";
cin>>iscp_marks;
cout<<"\nCPP Marks: ";
cin>>cpp_marks;
cout<<"\nCPP Lab Marks :";
cin>>cpplab_marks;
cout<<"\nEconomics Marks :";
cin>>eco_marks;
cout<<"\nDigital Electronics Marks :";
cin>>de_marks;
cout<<"\nDigital Electronics Lab Marks :";
cin>>delab_marks;
cout<<"\nDesign and Analysis of Algorithms Marks :";
cin>>daa_marks;
cout<<"\nDesign and Analysis of Algorithms Lab Marks :";
cin>>daalab_marks;
cout<<"\nMaths Marks :";
cin>>mat_marks;
cout<<"\nISES Marks :";
cin>>ises_marks;
Calculate_Grade();
}

void Student::showdata() const
{
    cout<<"/t Student Details"<<endl;
    cout<<"Roll Number :"<<Roll_no<<endl;
    cout<<"\nName :"<<Name;
    cout<<"\nPython Marks(out of 100) :"<<py_marks<<endl;
    cout<<"\nISCP Marks :"<<iscp_marks<<endl;
    cout<<"\nCPP Marks: "<<cpp_marks<<endl;
    cout<<"\nCPP Lab Marks :"<<cpplab_marks<<endl;
    cout<<"\nEconomics Marks :"<<eco_marks<<endl;
    cout<<"\nDigital Electronics Marks :"<<de_marks<<endl;
    cout<<"\nDigital Electronics Lab Marks :"<<delab_marks<<endl;;
    cout<<"\nDesign and Analysis of Algorithms Marks :"<<daa_marks<<endl;
    cout<<"\nDesign and Analysis of Algorithms Lab Marks :"<<daalab_marks<<endl;
    cout<<"\nMaths Marks :"<<mat_marks<<endl;
    cout<<"\nISES Marks :"<<ises_marks<<endl;
    cout<<"\nPercentage :"<<Percentage<<endl;
    cout<<"\nGrade :"<<Grade<<endl;
}

void Student::show_tabular() const
{
    cout<<Roll_no<<setw(6)<<"
    "<<Name<<setw(8)<<py_marks<<setw(5)<<iscp_marks<<setw(7)<<cpp_marks<<setw(6)

```

```
<<cpplab_marks<<setw(9)<<eco_marks<<setw(6)<<de_marks<<setw(5)<<delab_marks<<setw(8)<<daa_marks<<setw(10)<<daalab_marks<<setw(9)
<<mat_marks<<setw(6)<<ises_marks<<setw(11)<<Percentage<<setw(10)<<Grade<<endl;
}
```

```
int Student::retrollno() const
{
    return Roll_no;
}
```

```
void write_student();
void display_all();
void display_sp(int);
void modify_student(int);
void delete_student(int);
void class_result();
void result();
void intro();
void entry_menu();
```

```
int main()
{
    char ch;
    cout.setf(ios::fixed|ios::showpoint);
    cout<<setprecision(2);
    intro();
    do
    {
        system("cls");
        cout<<"\n\n\n\tMAIN MENU";
        cout<<"\n\n\t01. RESULT MENU";
        cout<<"\n\n\t02. ENTRY/EDIT MENU";
        cout<<"\n\n\t03. EXIT";
        cout<<"\n\n\tPlease Select Your Option (1-3) ";
        cin>>ch;
        switch(ch)
        {
            case '1': result();
                       break;
            case '2': entry_menu();
                       break;
            case '3':
                       break;
            default :cout<<"\a";
        }
    }while(ch!='3');
    return 0;
}
```

```
void write_student()
{
    Student st;
    ofstream outFile;
    outFile.open("student.dat",ios::binary|ios::app);
    st.getdata();
    outFile.write(reinterpret_cast<char *> (&st), sizeof(Student));
    outFile.close();
    cout<<"\n\nStudent record Has Been Created ";
    cin.ignore();
    cin.get();
}

void display_all()
{
    Student st;
    ifstream inFile;
    inFile.open("student.dat",ios::binary);
    if(!inFile)
    {
        cout<<"File could not be open !! Press any Key...";
        cin.ignore();
        cin.get();
        return;
    }
    cout<<"\n\n\n\t\tDISPLAY ALL RECORD !!!\n\n";
    while(inFile.read(reinterpret_cast<char *> (&st), sizeof(Student)))
    {
        st.showdata();
        cout<<"\n\n===== \n";
    }
    inFile.close();
    cin.ignore();
    cin.get();
}

void display_sp(int n)
{
    Student st;
    ifstream inFile;
    inFile.open("student.dat",ios::binary);
    if(!inFile)
    {
        cout<<"File could not be open !! Press any Key...";
        cin.ignore();
        cin.get();
        return;
    }
    bool flag=false;
    while(inFile.read(reinterpret_cast<char *> (&st), sizeof(Student)))
```

```
{
if(st.retrollno()==n)
{
    st.showdata();
    flag=true;
}
}
inFile.close();
if(flag==false)
cout<<"\n\nrecord not exist";
cin.ignore();
cin.get();
}

void modify_student(int n)
{
    bool found=false;
    Student st;
    fstream File;
    File.open("student.dat",ios::binary|ios::in|ios::out);
    if(!File)
    {
        cout<<"File could not be open !! Press any Key...";
        cin.ignore();
        cin.get();
        return;
    }
    while(!File.eof() && found==false)
    {
        File.read(reinterpret_cast<char *> (&st), sizeof(Student));
        if(st.retrollno()==n)
        {
            st.showdata();
            cout<<"\n\nPlease Enter The New Details of student"<<endl;
            st.getdata();
            int pos=(-1)*static_cast<int>(sizeof(st));
            File.seekp(pos,ios::cur);
            File.write(reinterpret_cast<char *> (&st), sizeof(Student));
            cout<<"\n\n\t Record Updated";
            found=true;
        }
    }
    File.close();
    if(found==false)
        cout<<"\n\n Record Not Found ";
    cin.ignore();
    cin.get();
}
```

```
void delete_student(int n)
{
    Student st;
    ifstream inFile;
    inFile.open("student.dat",ios::binary);
    if(!inFile)
    {
        cout<<"File could not be open !! Press any Key...";
        cin.ignore();
        cin.get();
        return;
    }
    ofstream outFile;
    outFile.open("Temp.dat",ios::out);
    inFile.seekg(0,ios::beg);
    while(inFile.read(reinterpret_cast<char *> (&st), sizeof(Student)))
    {
        if(st.retrollno()!=n)
        {
            outFile.write(reinterpret_cast<char *> (&st), sizeof(Student));
        }
    }
    outFile.close();
    inFile.close();
    remove("student.dat");
    rename("Temp.dat","student.dat");
    cout<<"\n\n\tRecord Deleted ..";
    cin.ignore();
    cin.get();
}
```

```
void class_result()
{
    Student st;
    ifstream inFile;
    inFile.open("student.dat",ios::binary);
    if(!inFile)
    {
        cout<<"File could not be open !! Press any Key...";
        cin.ignore();
        cin.get();
        return;
    }
}
```

```
//py_marks+iscp_marks+cpp_marks+cpplab_marks+eco_marks+de_marks+delab_marks+daa
_marks+daalab_marks+mat_marks+ises_marks)
cout<<"\n\n\t\tALL STUDENTS RESULT \n\n";
cout<<"=====
=====\\n";
```

```

cout<<"R.No    Name    Py  ISCP  CPP  CPPLAB  ECO  DE  DELAB  DAA  DAALAB
MAT  ISES  Percentage  Grade"<<endl;
cout<<"=====
=====\\n";
while(inFile.read(reinterpret_cast<char *> (&st), sizeof(Student)))
{
    st.show_tabular();
}
cin.ignore();
cin.get();
inFile.close();
}

void result()
{
    char ch;
    int rno;
    system("cls");
    cout<<"\\n\\n\\n\\tRESULT MENU";
    cout<<"\\n\\n\\n\\t1. Class Result";
    cout<<"\\n\\n\\n\\t2. Student Report Card";
    cout<<"\\n\\n\\n\\t3. Back to Main Menu";
    cout<<"\\n\\n\\n\\tEnter Choice (1/2/3)? ";
    cin>>ch;
    system("cls");
    switch(ch)
    {
        case '1' :    class_result(); break;
        case '2' :    cout<<"\\n\\n\\tEnter Roll Number Of Student : "; cin>>rno;
                        display_sp(rno); break;
        case '3' :    break;
        default:      cout<<"\\a";
    }
}

void intro()
{
    cout<<"\\n\\n\\n\\t\\t SRM STUDENT REPORT";
    cout<<"\\n\\n\\t\\t MANAGEMENT SYSTEM";
    cout<<"\\n\\n\\t\\t PROJECT";
    cout<<"\\n\\n\\n\\tMADE BY :GROUP 9";
    cout<<"\\n\\tSECTION :CSED ";
    cin.get();
}

void entry_menu()
{
    char ch;
    int num;
    system("cls");

```



```
cout<<"\n\n\n\tENTRY MENU";
cout<<"\n\n\n\t1.CREATE STUDENT RECORD";
cout<<"\n\n\n\t2.DISPLAY ALL STUDENTS RECORDS";
cout<<"\n\n\n\t3.SEARCH STUDENT RECORD ";
cout<<"\n\n\n\t4.MODIFY STUDENT RECORD";
cout<<"\n\n\n\t5.DELETE STUDENT RECORD";
cout<<"\n\n\n\t6.BACK TO MAIN MENU";
cout<<"\n\n\n\tPlease Enter Your Choice (1-6) ";
cin>>ch;
system("cls");
switch(ch)
{
case '1':      write_student(); break;
case '2':      display_all(); break;
case '3':      cout<<"\n\n\n\tPlease Enter The roll number "; cin>>num;
                display_sp(num); break;
case '4':      cout<<"\n\n\n\tPlease Enter The roll number "; cin>>num;
                modify_student(num);break;
case '5':      cout<<"\n\n\n\tPlease Enter The roll number "; cin>>num;
                delete_student(num);break;
case '6':      break;
default:      cout<<"\a"; entry_menu();
}
}
```

SRM STUDENT REPORT
MANAGEMENT SYSTEM
PROJECT

MADE BY :GROUP 9
SECTION :CSED |

MAIN MENU

- 01. RESULT MENU
- 02. ENTRY/EDIT MENU
- 03. EXIT

Please Select Your Option (1-3) |

RESULT MENU

- 1. Class Result
- 2. Student Report Card
- 3. Back to Main Menu

Enter Choice (1/2/3)? |

ENTRY MENU

- 1.CREATE STUDENT RECORD
- 2.DISPLAY ALL STUDENTS RECORDS
- 3.SEARCH STUDENT RECORD
- 4.MODIFY STUDENT RECORD
- 5.DELETE STUDENT RECORD
- 6.BACK TO MAIN MENU

Please Enter Your Choice (1-6) |

Roll Number :
203

Name :Basheer

Python Marks(out of 100) :56

ISCP Marks :78

CPP Marks: 45

CPP Lab Marks :76

Economics Marks :78

Digital Electronics Marks :56

Digital Electronics Lab Marks :34

Design and Analysis of Algorithms Marks :78

Design and Analysis of Algorithms Lab Marks :98

Maths Marks :76

ISES Marks :56

Student record Has Been Created |

ALL STUDENTS RESULT

R.No	Name	Py	ISCP	CPP	CPPLAB	ECO	DE	DELAB	DAA	DAALAB	MAT	ISES	Percentage	Grade
238	Kartheek	78	69	85	73	73	92	91	82	74	83	82	80.18	B
203	Basheer	56	78	45	76	78	56	34	78	98	76	56	66.45	D

CHAPTER 9:

CONCLUSION

This report contains an overview of the project. With our project user can easily see as well as manage all student reports. We also Included options like Modify student record, search student record, Delete student record...etc. Student management systems make faculty jobs easier by giving them an easy way to find and sort information.

Overall, the team feels that the project is a success. A great deal of knowledge had been gained through this project.