PROJECT REPORT

Submitted by

Parvathy Ullas-RA2211003010098
Sudarshan Kannan-RA2211003010107

Under the Guidance of

Mr. S. Iniyan

Assistant Professor, Department of Computing Technologies

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR - 603203

MAY 2023

SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this Project Report titled **"Student Report Card System"** is the bonafide work done by Parvathy Ullas-RA2211003010098 and Sudarshan Kannan-RA2211003010107 who completed the project under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE	**SIGNATURE**
Dr. S. Iniyan	**Dr. M. Pushpalatha**
OODP – Course Faculty	Professor & Head
Associate Professor	Department of Computing Technologies
Department of Computing Technologies	School of Computing
SRMIST	SRMIST

TABLE OF CONTENTS

PROBLEM STATEMENT

A student report card is a document that summarizes a student's academic performance over a specific period. It typically includes the student's name, roll number, class, and grades obtained in different subjects. The report card provides valuable information to parents, teachers, and students regarding the student's academic progress and helps identify areas where the student needs improvement. C++ can be used to make a student report card. The program should take input from the user regarding the student's personal details, such as name, roll number, and class. The program should then allow the user to input the student's marks for different subjects.

The program should then calculate the student's overall percentage and display the report card with the following information:

- Personal Details: Name, Roll Number, Class
- Subject-wise Marks: Display marks obtained by the student in each subject.
- Total Marks: Display the total marks obtained by the student out of the total marks for all subjects.
- Percentage: Calculate and display the student's overall percentage.

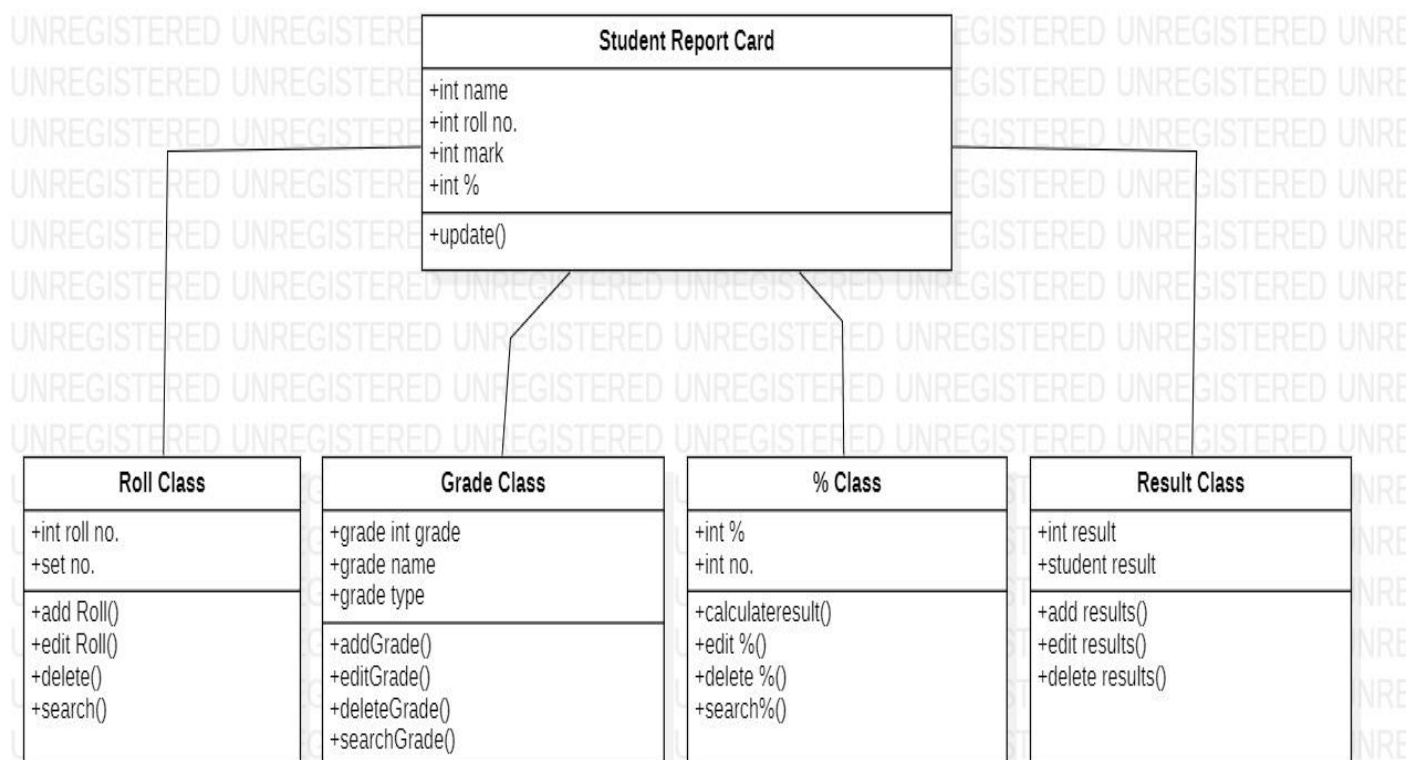The program should also allow the user to save the report card as a file on the computer. Additionally, the program should have the option to generate multiple report cards for different students.
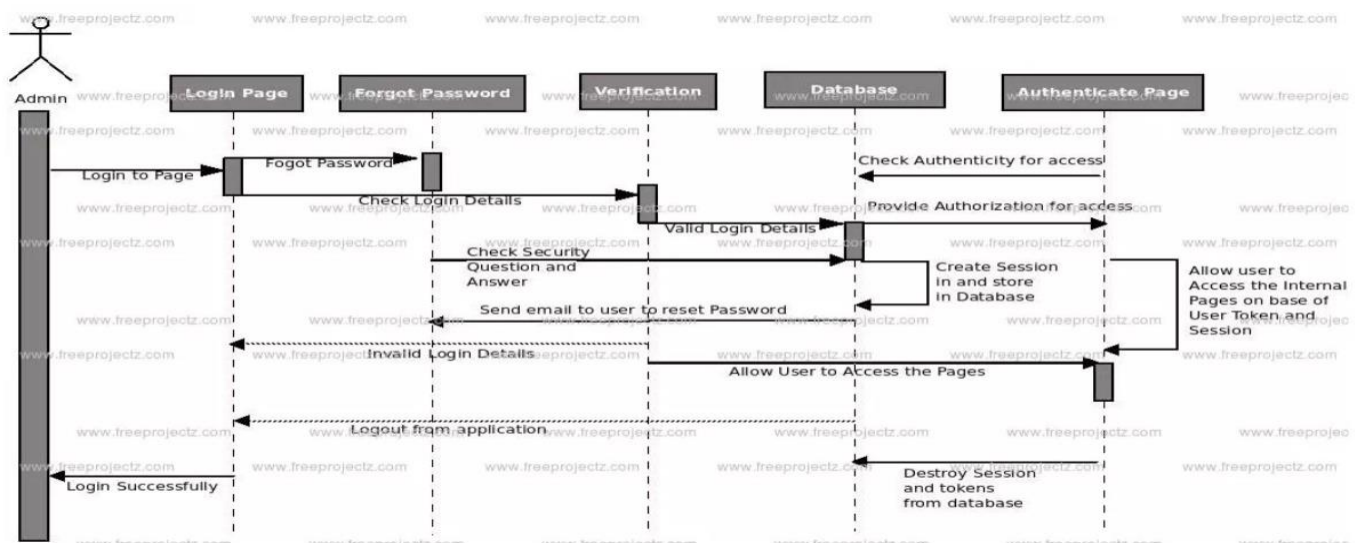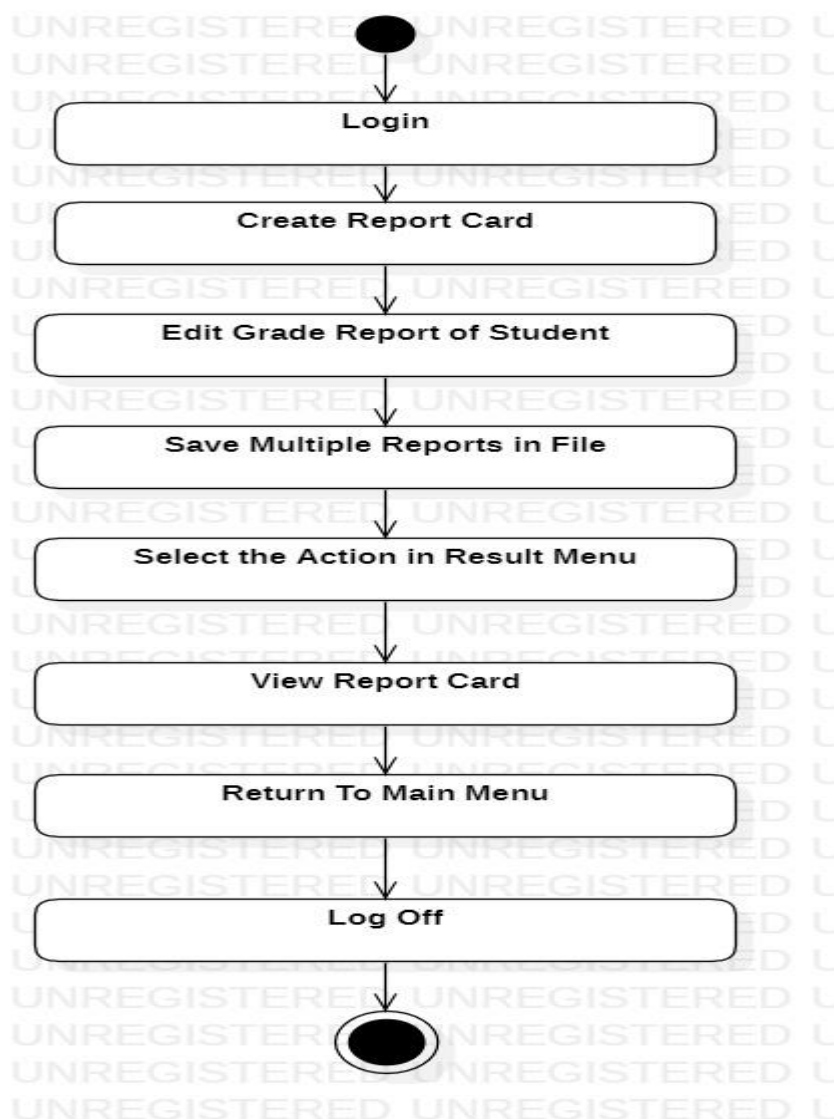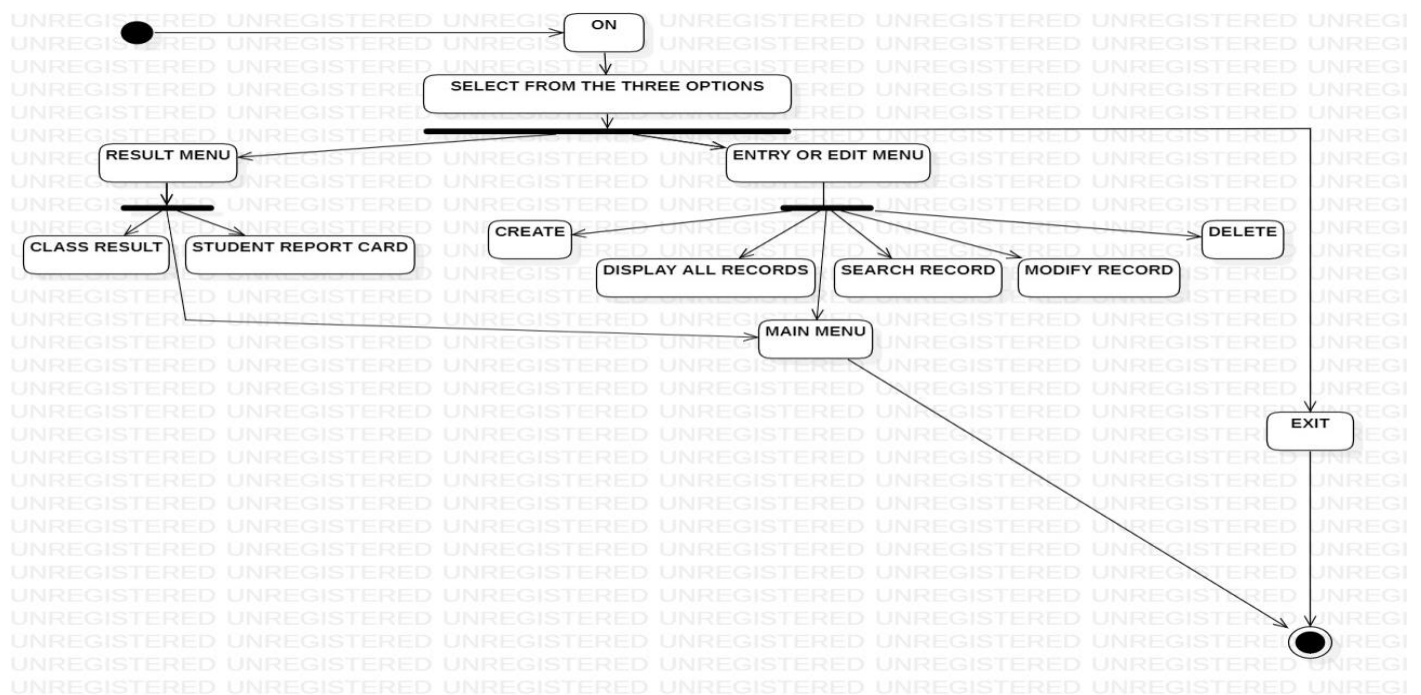
DIAGRAMS:-

1.USE CASE DIAGRAM:-
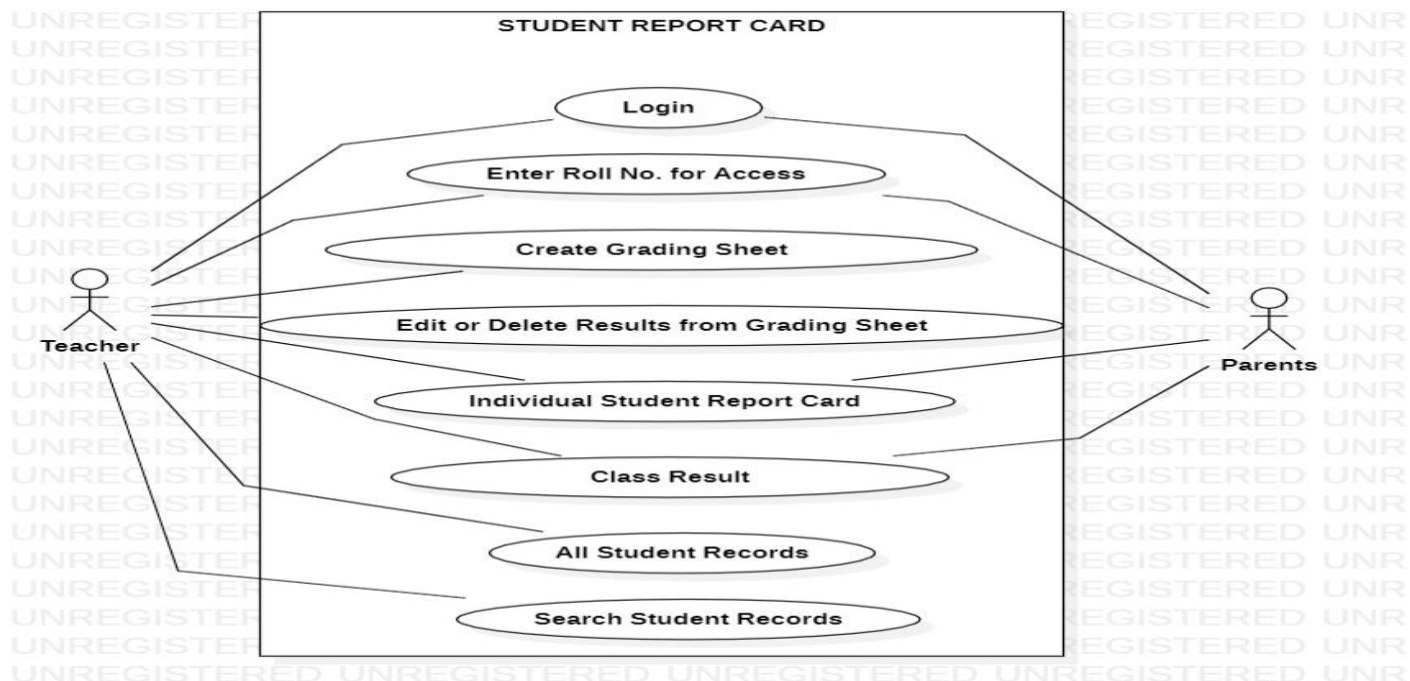
2.CLASS DIAGRAM:-
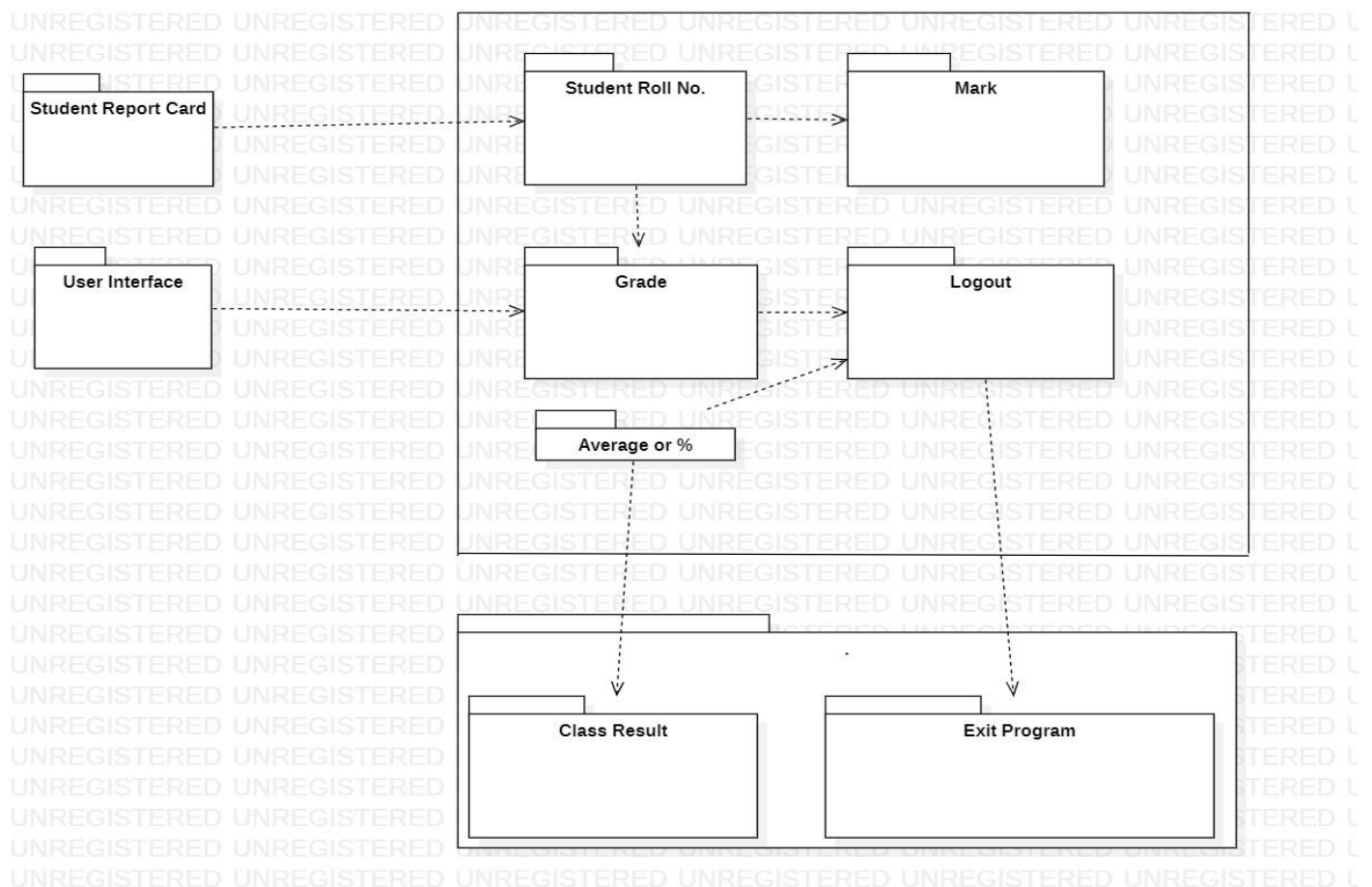
3.SEQUENCE DIAGRAM:-

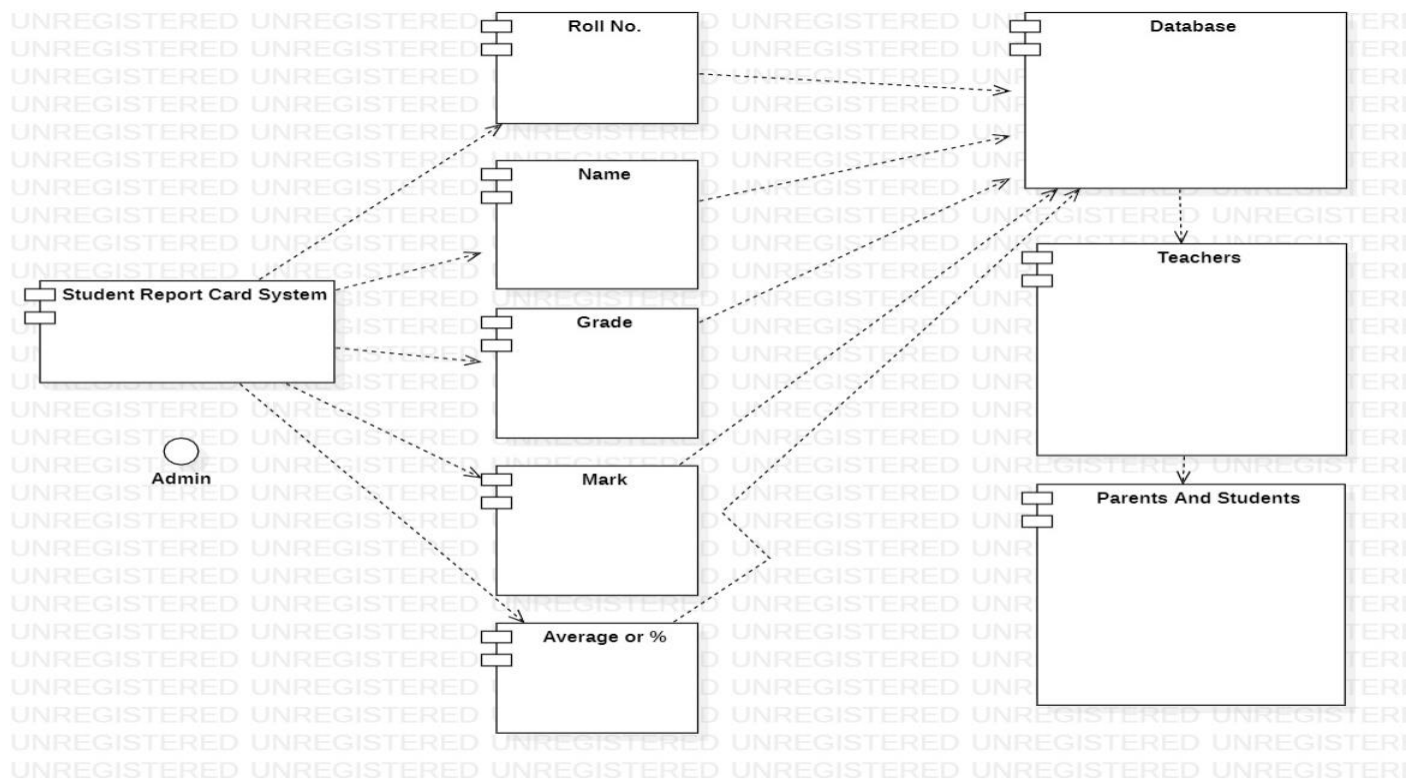# 4.STATE CHART DIAGRAM:-

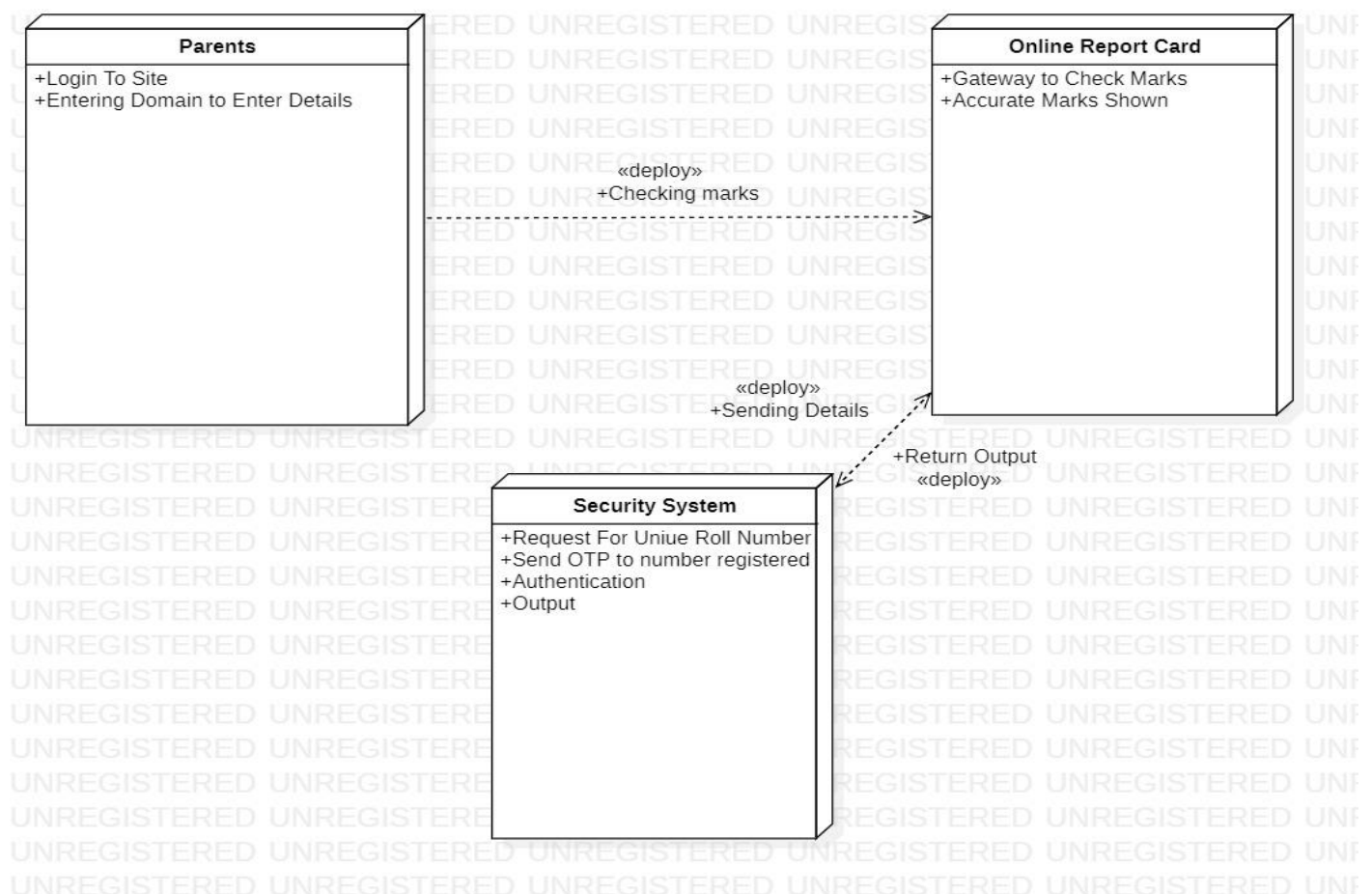# 5.ACTIVITY DIAGRAM:-

6.PACKAGE DIAGRAM:-

7.COMPONENT DIAGRAM:-

8.DEPLOYMENT DIAGRAM:-

Modules Of Project:-

This is a C++ program that manages student records, including writing records to a binary file, displaying all records, displaying a specific record based on the roll number, modifying a record, and deleting a record.

The program starts with a menu, which offers two options - a result menu and an entry/edit menu.

The program includes a class named "student" that contains private variables such as rollno, name, p_marks, c_marks, m_marks, e_marks, cs_marks, per, and grade. The public member functions of the class are getdata, showdata, show_tabular, and retrollno.

The getdata function is used to accept data from the user, while the showdata function displays the student's information on the screen. The show_tabular function displays the student's information in tabular format. The calculate function calculates the student's percentage and grade based on the marks obtained in each subject.

The program also includes several functions to manage student records. The write_student function writes the student record to a binary file, while the display_all function reads all records from the binary file and displays them on the screen. The display_sp function accepts a roll number and reads the corresponding record from the binary file. The modify_student function accepts a roll number and updates the corresponding record in the binary file. The delete_student function accepts a roll number and deletes the corresponding record from the binary file. The class_result function displays all records in tabular format from the binary file.

The main function displays a welcome screen and a menu that allows the user to select an option. The program exits when the user chooses to exit the program.

CODE:-

```cpp
#include<iostream>
#include<fstream>
#include<iomanip>
using namespace std;

class student
{
    int rollno;
    char name[50];
    int p_marks, c_marks, m_marks, e_marks, cs_marks;
    double per;
    char grade;
    void calculate();    //function to calculate grade
public:
    void getdata();      //function to accept data from user
    void showdata() const;  //function to show data on screen
    void show_tabular() const;
    int retrollno() const;
}; //class ends here

void student::calculate()
{
    per=(p_marks+c_marks+m_marks+e_marks+cs_marks)/5.0;
    if(per>=60)
        grade='A';
    else if(per>=50)
        grade='B';
    else if(per>=33)
        grade='C';
    else
        grade='F';
}

void student::getdata()
{
    cout<<"\nEnter The roll number of student ";
    cin>>rollno;
    cout<<"\n\nEnter The Name of student ";
    cin.ignore();
    cin.getline(name,50);
    cout<<"\nEnter The marks in physics out of 100 : ";
```

```cpp
    cout<<"\n\nEnter The Name of student ";
    cin.ignore();
    cin.getline(name,50);
    cout<<"\nEnter The marks in physics out of 100 : ";
    cin>>p_marks;
    cout<<"\nEnter The marks in chemistry out of 100 : ";
    cin>>c_marks;
    cout<<"\nEnter The marks in maths out of 100 : ";
    cin>>m_marks;
    cout<<"\nEnter The marks in english out of 100 : ";
    cin>>e_marks;
    cout<<"\nEnter The marks in computer science out of 100 : ";
    cin>>cs_marks;
    calculate();
}

void student::showdata() const
{
    cout<<"\nRoll number of student : "<<rollno;
    cout<<"\nName of student : "<<name;
    cout<<"\nMarks in Physics : "<<p_marks;
    cout<<"\nMarks in Chemistry : "<<c_marks;
    cout<<"\nMarks in Maths : "<<m_marks;
    cout<<"\nMarks in English : "<<e_marks;
    cout<<"\nMarks in Computer Science :"<<cs_marks;
    cout<<"\nPercentage of student is :"<<per;
    cout<<"\nGrade of student is :"<<grade;
}

void student::show_tabular() const
{
    cout<<rollno<<setw(6)<<" "<<name<<setw(10)<<p_marks<<setw(4)<<c_marks<<setw(4)<<m_marks<<setw(4)
        <<e_marks<<setw(4)<<cs_marks<<setw(8)<<per<<setw(6)<<grade<<endl;
}

int student::retrollno() const
{
    return rollno;
}

void write_student();   //write the record in binary file
```
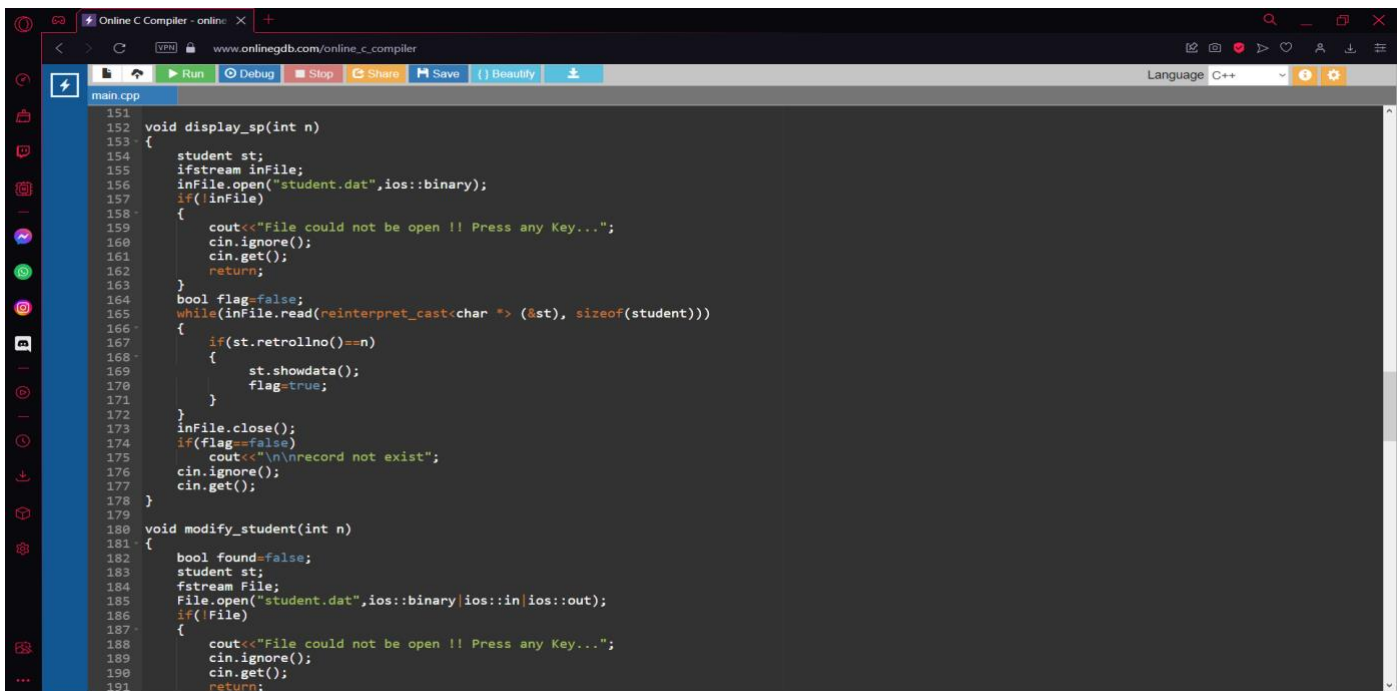
```cpp
76  }
77
78  void write_student();   //write the record in binary file
79  void display_all(); //read all records from binary file
80  void display_sp(int);   //accept rollno and read record from binary file
81  void modify_student(int);   //accept rollno and update record of binary file
82  void delete_student(int);   //accept rollno and delete selected records from binary file
83  void class_result();    //display all records in tabular format from binary file
84  void result();          //display result menu
85  void intro();           //display welcome screen
86  void entry_menu(); //display entry menu on screen
87
88  int main()
89  {
90      char ch;
91      cout.setf(ios::fixed|ios::showpoint);
92      cout<<setprecision(2); // program outputs decimal number to two decimal places
93      intro();
94      do
95      {
96          system("cls");
97          cout<<"\n\n\n\tMAIN MENU";
98          cout<<"\n\n\t01. RESULT MENU";
99          cout<<"\n\n\t02. ENTRY/EDIT MENU";
100         cout<<"\n\n\t03. EXIT";
101         cout<<"\n\n\tPlease Select Your Option (1-3) ";
102         cin>>ch;
103         switch(ch)
104         {
105             case '1': result();
106                 break;
107             case '2': entry_menu();
108                 break;
109             case '3':
110                 break;
111             default :cout<<"\a";
112         }
113     }while(ch!='3');
114     return 0;
115 }
116
```

```cpp
113     }while(ch!='3');
114     return 0;
115 }
116
117 void write_student()
118 {
119     student st;
120     ofstream outFile;
121     outFile.open("student.dat",ios::binary|ios::app);
122     st.getdata();
123     outFile.write(reinterpret_cast<char *> (&st), sizeof(student));
124     outFile.close();
125         cout<<"\n\nStudent record Has Been Created ";
126     cin.ignore();
127     cin.get();
128 }
129 void display_all()
130 {
131     student st;
132     ifstream inFile;
133     inFile.open("student.dat",ios::binary);
134     if(!inFile)
135     {
136         cout<<"File could not be open !! Press any Key...";
137         cin.ignore();
138         cin.get();
139         return;
140     }
141     cout<<"\n\n\n\t\tDISPLAY ALL RECORD !!!\n\n";
142     while(inFile.read(reinterpret_cast<char *> (&st), sizeof(student)))
143     {
144         st.showdata();
145         cout<<"\n\n====================================\n";
146     }
147     inFile.close();
148     cin.ignore();
149     cin.get();
150 }
151
152 void display_sp(int n)
153 {
```
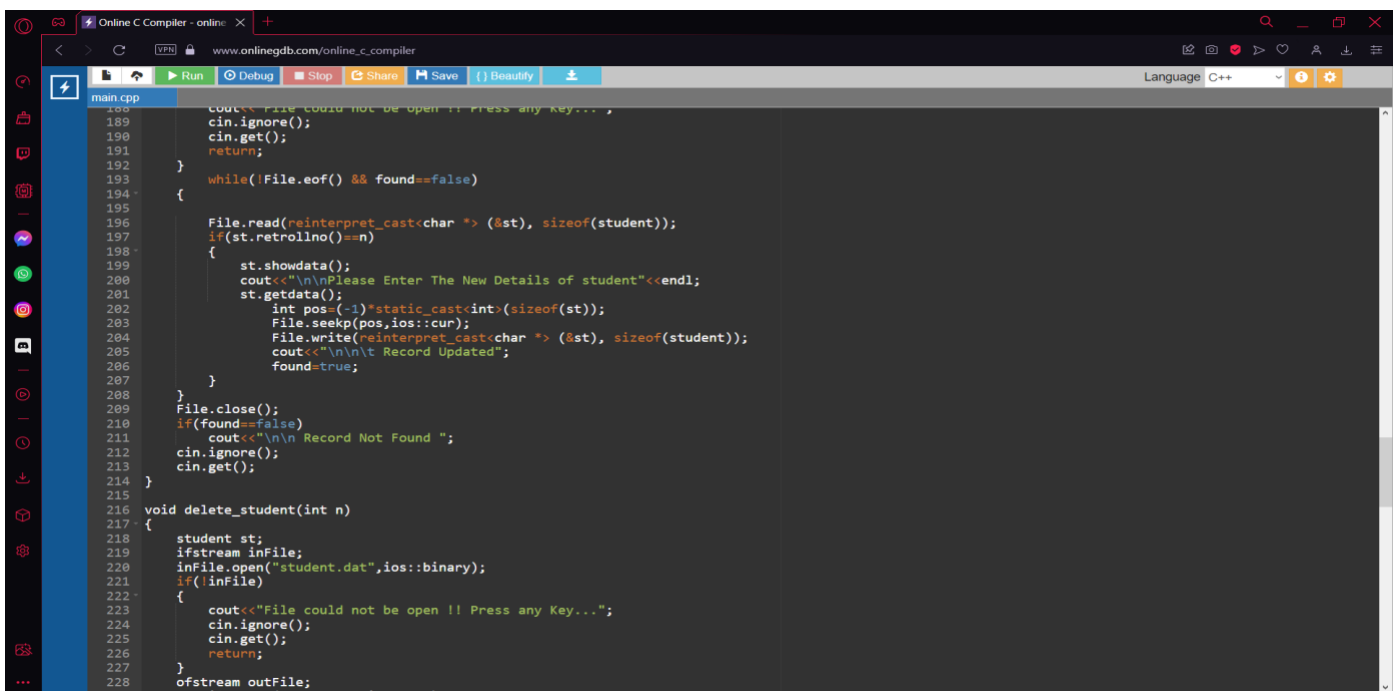
```cpp
151
152  void display_sp(int n)
153  {
154      student st;
155      ifstream inFile;
156      inFile.open("student.dat",ios::binary);
157      if(!inFile)
158      {
159          cout<<"File could not be open !! Press any Key...";
160          cin.ignore();
161          cin.get();
162          return;
163      }
164      bool flag=false;
165      while(inFile.read(reinterpret_cast<char *> (&st), sizeof(student)))
166      {
167          if(st.retrollno()==n)
168          {
169              st.showdata();
170              flag=true;
171          }
172      }
173      inFile.close();
174      if(flag==false)
175          cout<<"\n\nrecord not exist";
176      cin.ignore();
177      cin.get();
178  }
179
180  void modify_student(int n)
181  {
182      bool found=false;
183      student st;
184      fstream File;
185      File.open("student.dat",ios::binary|ios::in|ios::out);
186      if(!File)
187      {
188          cout<<"File could not be open !! Press any Key...";
189          cin.ignore();
190          cin.get();
191          return;
```
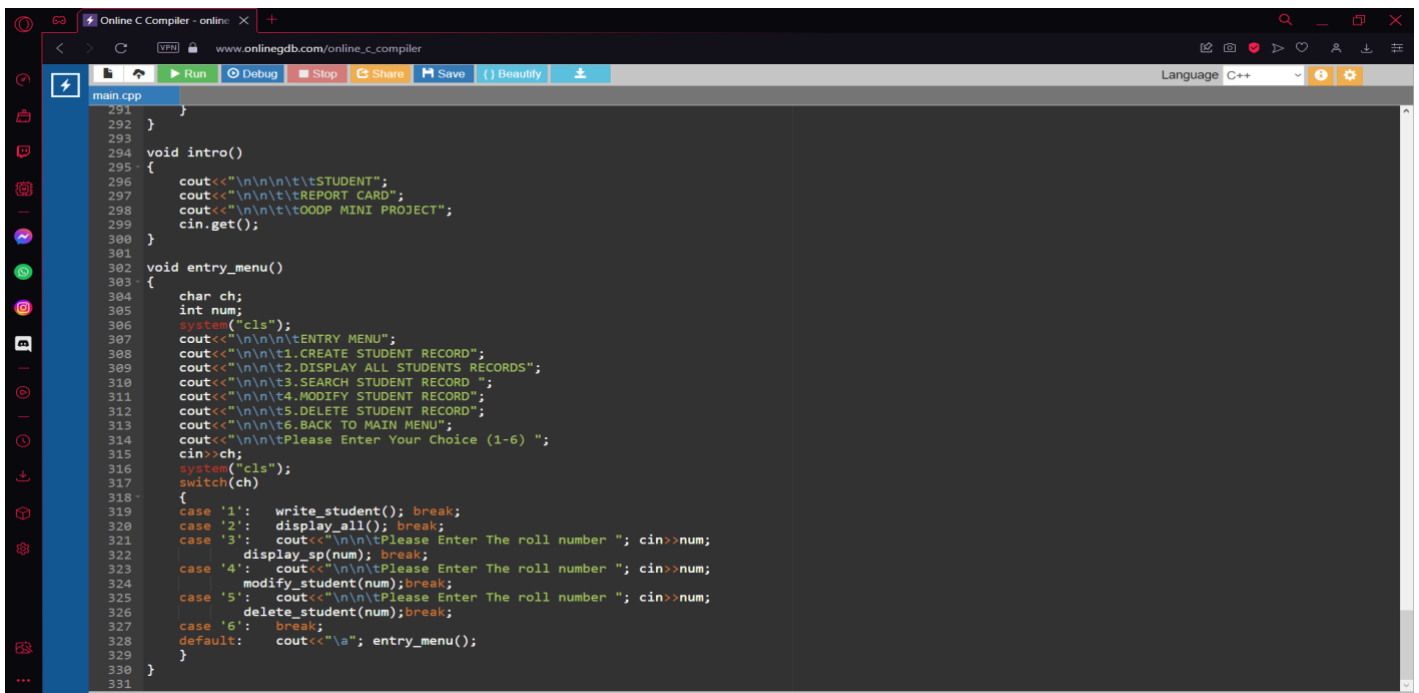
```cpp
188          cout<<"File could not be open !! Press any Key...";
189          cin.ignore();
190          cin.get();
191          return;
192      }
193      while(!File.eof() && found==false)
194      {
195
196          File.read(reinterpret_cast<char *> (&st), sizeof(student));
197          if(st.retrollno()==n)
198          {
199              st.showdata();
200              cout<<"\n\nPlease Enter The New Details of student"<<endl;
201              st.getdata();
202                  int pos=(-1)*static_cast<int>(sizeof(st));
203                  File.seekp(pos,ios::cur);
204                  File.write(reinterpret_cast<char *> (&st), sizeof(student));
205                  cout<<"\n\n\t Record Updated";
206                  found=true;
207          }
208      }
209      File.close();
210      if(found==false)
211          cout<<"\n\n Record Not Found ";
212      cin.ignore();
213      cin.get();
214  }
215
216  void delete_student(int n)
217  {
218      student st;
219      ifstream inFile;
220      inFile.open("student.dat",ios::binary);
221      if(!inFile)
222      {
223          cout<<"File could not be open !! Press any Key...";
224          cin.ignore();
225          cin.get();
226          return;
227      }
228      ofstream outFile;
```

```cpp
                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;
    }
    cout<<"\n\n\t\tALL STUDENTS RESULT \n\n";
    cout<<"=======================================================\n";
    cout<<"R.No      Name         P   C   M   E   CS   %age   Grade"<<endl;
    cout<<"=======================================================\n";
    while(inFile.read(reinterpret_cast<char *> (&st), sizeof(student)))
    {
        st.show_tabular();
    }
```

```cpp
    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\tSTUDENT";
    cout<<"\n\n\t\tREPORT CARD";
    cout<<"\n\n\t\tOODP MINI PROJECT";
    cin.get();
}

void entry_menu()
{
```

```cpp
291          }
292     }
293
294     void intro()
295   ⌐ {
296          cout<<"\n\n\n\t\tSTUDENT";
297          cout<<"\n\n\t\tREPORT CARD";
298          cout<<"\n\n\t\tOODP MINI PROJECT";
299          cin.get();
300     }
301
302     void entry_menu()
303   ⌐ {
304          char ch;
305          int num;
306          system("cls");
307          cout<<"\n\n\n\tENTRY MENU";
308          cout<<"\n\n\t1.CREATE STUDENT RECORD";
309          cout<<"\n\n\t2.DISPLAY ALL STUDENTS RECORDS";
310          cout<<"\n\n\t3.SEARCH STUDENT RECORD ";
311          cout<<"\n\n\t4.MODIFY STUDENT RECORD";
312          cout<<"\n\n\t5.DELETE STUDENT RECORD";
313          cout<<"\n\n\t6.BACK TO MAIN MENU";
314          cout<<"\n\n\tPlease Enter Your Choice (1-6) ";
315          cin>>ch;
316          system("cls");
317          switch(ch)
318   ⌐     {
319          case '1':   write_student(); break;
320          case '2':   display_all(); break;
321          case '3':   cout<<"\n\n\tPlease Enter The roll number "; cin>>num;
322                  display_sp(num); break;
323          case '4':   cout<<"\n\n\tPlease Enter The roll number "; cin>>num;
324                  modify_student(num);break;
325          case '5':   cout<<"\n\n\tPlease Enter The roll number "; cin>>num;
326                  delete_student(num);break;
327          case '6':   break;
328          default:    cout<<"\a"; entry_menu();
329          }
330     }
331
```

OUTPUT:-


```
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)?
```

```
                 ALL STUDENTS RESULT
==========================================================
R.No       Name        P   C   M   E   CS   %age    Grade
==========================================================
1       A          90  80  70  80  90   82.00     A
2       B          100 90  80  70  60   80.00     A
3       C          40  50  60  70  80   60.00     A
```

```
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)
```

Conclusion And Result:-

This program allows users to input and store student information such as name, roll number, and marks obtained in various subjects. The program then calculates the percentage and assigns a grade to the student based on their performance. The program provides several options such as displaying all student records, displaying a specific student record, modifying a student record, deleting a student record, and displaying the class result in a tabular format.

The program uses a class called 'student' to store and manipulate student data. It includes member functions to input data, calculate the percentage and grade, display the data, and return the roll number. The program also includes functions to read and write data to a binary file.

The main function of the program includes a menu that allows users to select various options to perform operations on student records. The program uses switch case statements to execute different functions based on user input.

In conclusion, this program provides a basic framework for storing and manipulating student data in a file.

References:-

- https://www.codewithc.com/student-report-card-system-project-in-c/
- https://www.sourcecodester.com/cc/15442/student-report-card-management-system-c-free-source-code.html
- YouTube