Fee Maintenance System

**PROJECT REPORT**

Submittedby

**S.Balasubramanian(21CS020)**

*inpartialfulfillmentforthe award ofthe degree* of

**BACHELOROFENGINEERING**

**in**

COMPUTERSCIENCEANDENGINEERING

**P.S.R.ENGINEERINGCOLLEGE,SIVAKASI–626140**

(AnAutonomousinstitution,AffiliatedtoAnnaUniversity,Chennai)

**ANNAUNIVERSITY:CHENNAI600025**

# ABSTRACT

TheFeeMaintenancesystemimplementedinc language is a user friendly Interface for Maintaing Fee details.It reads user inputs and stores them whenever displaymethodisinvokedItdisplaysthefeedetailsand expenditure further it alerts for fee collection.

**ACKNOWLEDEMENT**

We take this opportunity to all those who helped towards successful completion of this mini project. At the very outset we thank the almighty for his profuse blessings showered on us. We thank our beloved parents whose encouragement and support help us to complete our project successfully.

It is our greatest pleasure to convey our thanks to **Thiru R. Solaisamy, Correspondent** and **Director Er. S. vigneswari Arunkumar B. Tech., PSR engineering college, Sivakasi** for providing required facilities and suitable infrastructure to complete our project.

It is our greatest privilege to convey our thanks to **Dr. J. S. Senthilkumaar, M.E., Ph.D., Principal** for continuous support to complete our project without hurdles.

WeproudprofoundgratitudetoourbelovedHeadoftheDepartment **Dr. A. Ramathilagam, M.E., Ph.D., Professor** for providing ample facilities to complete our project successfully.

We also bound to thanks to all Faulty and Non-teaching staff members of The **Department of Computer Science and Engineering** whose support and cooperation also contributed much to complete this project work.

TABLEOFCONTENTS

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TOPIC** | **PAGENO** |
| 1 | INTRODUCTION |  |
| 1.1 | OBJECTIVES |  |
| 2 | PROJECTOVERVIEW |  |
| 3 | METHODOLOGY |  |
| 3.1 | SYSTEMARCHITECTURE |  |
| 3.2 | TECHNOLOGIESUSED |  |
| 4 | SYSTEMFEATURES |  |
| 4.1 | ADDSTUDENT |  |
| 4.2 | DELETESTUDENT |  |
| 4.3 | DISPLAYBILLINGINFORMATION |  |
| 4.4 | EXPENDITURE |  |
| 4.5 | FEEPAYMENT |  |
| 5 | IMPLEMENTATION |  |
| 6 | RESULTSANDOUTCOMES |  |
| 6.1 | USERINTERFACE SCREENSHOTS |  |
| 6.2 | PERFORMANCEMETRICS |  |
| 7 | CONCLUSION |  |
| 8 | FUTUREENHANCEMENT |  |
| 9 | ACKNOWLEDGEMENT |  |
| 10 | REFERENCES |  |
| 11 | APPENDIX |  |

# Introduction:

**1.2.Objective:**

TheObjectiveofFeeMaintenancesystemisto provide a user friendly Interface for Maintaing Fee details.It reads user inputs and stores them whenever display method is invoked It displays the fee details and expenditure further it alerts for fee collection.

# ProjectOverview:

With a focus of providing simplicity in Fee maintenanceFee maintenance systemtakes informations like name,rollnoandfeeandstoresit.FurtherItdisplaythestored data and also summarize the expenditure.

# Methodology

* 1. **SystemArchitecture:**

The system architecture encompasses the structure and organization of theFee Maintenance system. It includes the design of the structure and each of the functions. Architecture ensures a clear and logical framework for effective Fee maintenance.

# 3.2.TechnologiesUsed:

The project leverages C for implementation, utilizing the Visual studio codeenvironment. The choice of C ensures a balance between performance and simplicity, while Visual studio code provides a familiar and accessible platform for users. These technologies collectively contribute to the project's efficiency and user- friendly design.

# SystemFeatures:

* 1. **Addstudent:**

Itreadsname,rollno,examfeesandtutionfeesfromthe operator as input.

# Deletestudent:

Itreadstherollnumberofstudentasinputanddeletes the particular student’s records.

# DisplayBillingInformation:

ItDisplaysnumberofstudentsFeedatabasedonpast inputs.

# Expenditure:

ItdisplaystheyearlyExpenditureoftheorganizationand calculates the revenue generated.

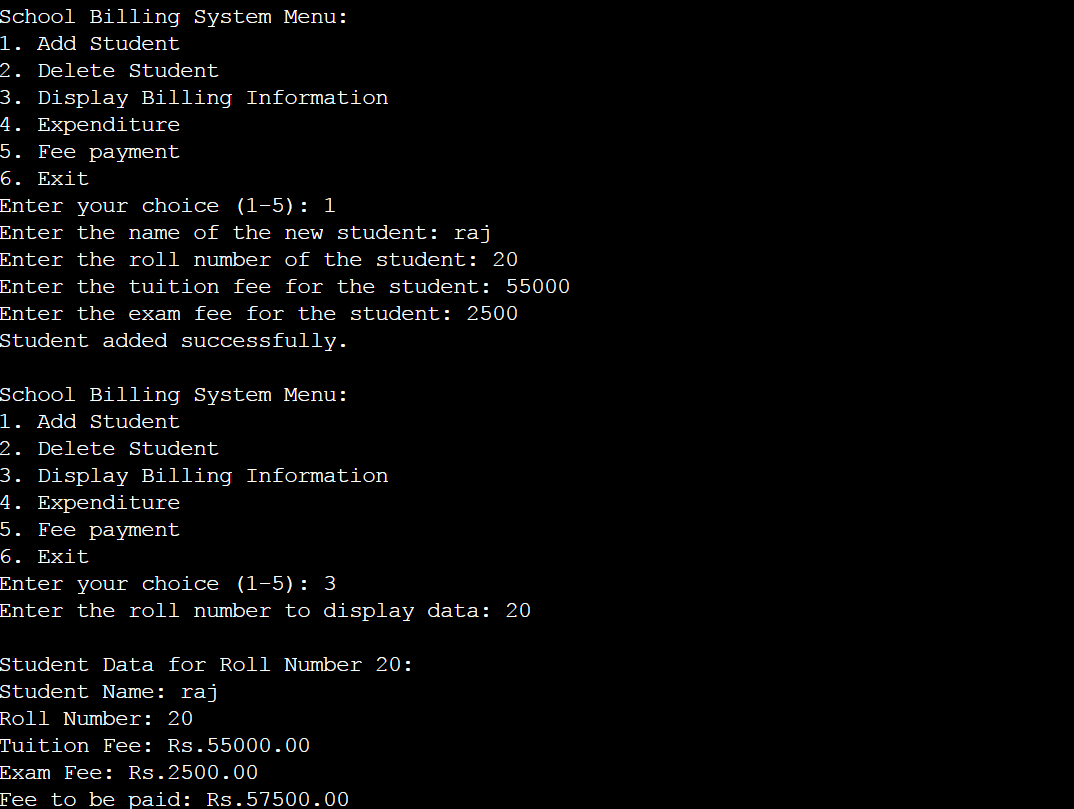
# Implementation:

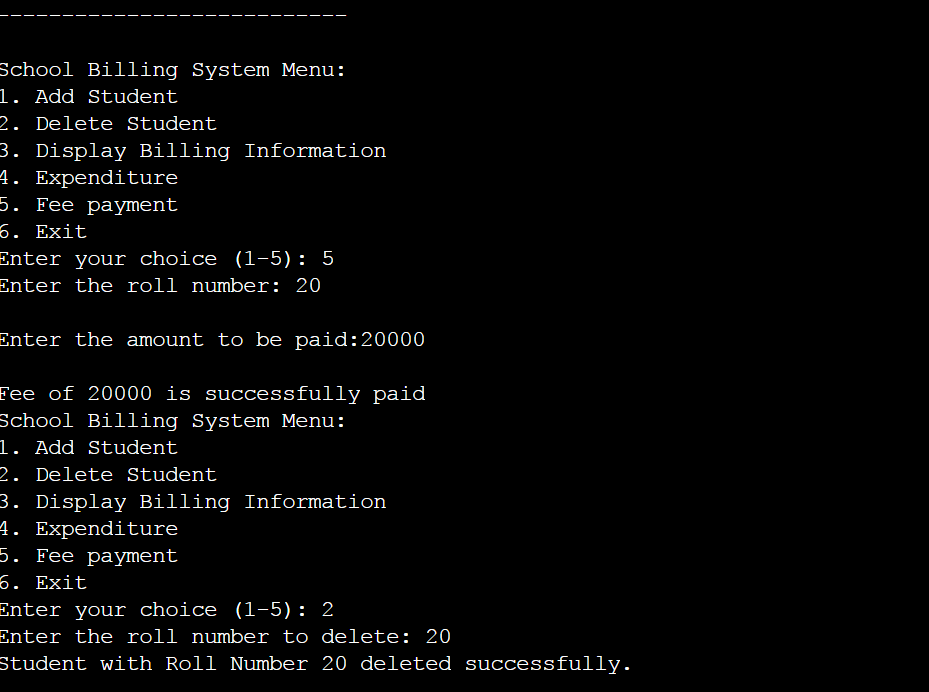
The implementation phase involves coding the project in C++ within the Visual studio code environment. The development process adheres to the established methodology, focusing on simplicity, efficiency, and user- friendly design.

# ResultsandOutcomes:

* 1. **UserInterfaceScreenshots:**

Screenshots showcase the system's user interface, providing a visual representation of the project's design and usability. These images demonstrate the simplicity and accessibility of the system.





# PerformanceMetrics:

Performancemetricsdefinestheefficiencyand responsivenessoftheEquitystrength analyzer. Metrics may include execution time for key operations, system resource utilization, and user feedback regarding system responsiveness.

# Conclusion:

In conclusion, The Fee Management system achieves its objectives by providing a user-friendly, efficient, and accessible solution for Fee collection. The system's simplicity, coupled with key features, ensures a practical tool for maintaining Fee details within the Visual studio code environment.

# FutureEnhancements:

Infuturethissystemcanbeintegratedaswebbased system with database and further the digital Application form can be attached to get data directly.

# Acknowledgments:

We extend our gratitude to all those who contributed to the development of the Equity strength analyzer. Special thanks to our project mentors for their guidance and insights throughout the developmentprocess. We also acknowledge the support from our peers and colleagues who provided valuable feedback and suggestions. Additionally, our appreciation goes to the open-sourcecommunityforthewealthofresourcesthat

facilitated the project. This collaborative effort has played acrucial role in the successful realization of the Equity strength analyzer.

# References:

* Cprogramming.com
* Geeksforgreeks.com

# Appendix:

* 1. **Codesnippets:**

## Addstudent:

voidaddStudent(){

if (numStudents < MAX\_STUDENTS) { printf("Enterthenameofthenewstudent:"); scanf("%s", students[numStudents].name);

printf("Enter the roll number of the student: "); scanf("%d",&students[numStudents].rollNumber);

printf("Enterthetuitionfeeforthestudent:");

scanf("%f",&students[numStudents].tuitionFee);

printf("Enter the exam fee for the student: "); scanf("%f", &students[numStudents].examFee);

estimated\_reven = students[numStudents].examFee + students[numStudents].tuitionFee;

numStudents++;

printf("Studentaddedsuccessfully.\n");

}else{

printf("Maximumnumberofstudentsreached.\n");

}

}

## Deletestudent:

voiddeleteStudent(introllNumber){ int i, found = 0;

for(i=0;i<numStudents;i++){

if(students[i].rollNumber==rollNumber){ found = 1;

break;

}

}

if(found){

//Shiftremainingelementstofillthegap for (int j = i; j < numStudents - 1; j++) {

students[j]=students[j+1];

}

numStudents--;

printf("Student with Roll Number %d deleted successfully.\n", rollNumber);

}else{

printf("Student with Roll Number %d not found.\n", rollNumber);

}

}

## Displaystudent data:

voiddisplayStudentData(introllNumber){ int i, found = 0;

for(i=0;i<numStudents;i++){

if(students[i].rollNumber==rollNumber){ found = 1;

break;

}

}

if(found){

float totalFee = students[i].tuitionFee + students[i].examFee; printf("\nStudent Data for Roll Number %d:\n",

rollNumber);

printf("Student Name: %s\n", students[i].name); printf("Roll Number: %d\n", students[i].rollNumber); printf("TuitionFee:Rs.%.2f\n",students[i].tuitionFee); printf("Exam Fee: Rs.%.2f\n", students[i].examFee);

students[i].totalfee=students[i].examFee+students[i].tuitionFee; printf("Fee to be paid: Rs.%.2f\n", students[i].totalfee); printf(" \n");

}else{

printf("Student with Roll Number %d not found.\n", rollNumber);

}

}

## Expenditure:

voidexpenditure(){ int

celeb=10000,maintenance=50000,Sfortech=120000,sfornontech=

60000;

inttotal=celeb+maintenance+Sfortech+sfornontech; printf("\n EXPENDITURE

");

printf("\nYearly maintenance Costs:%d",maintenance); printf("\nYearlycelebrationActivitiesspends:%d",celeb); printf("\nSalary for teaching staffs:%d",Sfortech); printf("\nSalaryforNonteachingstaffs:%d",sfornontech); printf("\nRevenue Generated:%d",estimated\_reven); if(estimated\_reven==0)

{

printf("\nRaiseAnnouncementforfeecollection");

}

if(estimated\_reven>0&&estimated\_reven<total)

{

printf("\nStillfeescollectionlags");

}

if(estimated\_reven==total)

{

printf("\nFeescollectionisover");

}

}

## Fee payment:

voidFeepayment(introllNumber,intamount)

{

inti,found=0; int temp;

for(i=0;i<numStudents;i++){

if(students[i].rollNumber==rollNumber){ found = 1;

break;

}

}

temp=(students[i].totalfee)-amount;

students[i].totalfee=temp; if (found) {

printf("\nFeeof%dissuccessfullypaid",amount);

}else{

printf("Student with Roll Number %d not found.\n", rollNumber);

}

}

## Main function:

intmain(){ int choice;

do {

printf("\nSchoolBillingSystemMenu:\n"); printf("1. Add Student\n");

printf("2.DeleteStudent\n");

printf("3.DisplayBillingInformation\n"); printf("4. Expenditure\n");

printf("5.Feepayment\n"); printf("6. Exit\n");

printf("Enteryourchoice(1-5):"); scanf("%d", &choice);

introllNumber,amount; switch (choice) {

case1:

addStudent(); break;

case2:

printf("Entertherollnumbertodelete:"); scanf("%d", &rollNumber); deleteStudent(rollNumber);

break; case 3:

printf("Entertherollnumbertodisplaydata:"); scanf("%d", &rollNumber); displayStudentData(rollNumber);

break; case 4:

expenditure();

break;

case5:

printf("Enter the roll number: "); scanf("%d", &rollNumber); printf("\nEntertheamounttobepaid:"); scanf("%d",&amount); Feepayment(rollNumber,amount); break;

case6:

printf("Exitingtheprogram.Goodbye!\n"); break;

default:

printf("Invalid choice. Please enter a number between 1 and 5.\n");

}

}while(choice!=6);

return0;

}

# UserManual:

## GettingStarted:

* + - InstallVisualstudioCode.
    - Installand createanewproject.

## Addstudent:

* Choosecase1.
* Addstudentdetailslikename,rollnumber,tuitionfeeand exam fee.

## Deletestudent:

* Choosecase2.
* Deletestudent record byenteringrollnumber asreference.

## Displaybillinginformation:

* Choosecase3.
* Itdisplays feeinformationofastudentbyreadingtheroll number as input.

## Expenditure:

* Choosecase4.
* ItdisplaysExpenditureandalsoalertsforfeescollection based on revenue generated.

## Fee payment:

* Choosecase5.
* Itreadstherollnumberand amountto bepaid.

## Exit:

* Choosecase5.
* Itterminatestheprogramandexitswith 0.