



Visvesvaraya Technological University
Belagavi, Karnataka

ವಿಶ್ವೇಶ್ವರಯ್ಯ ತಾಂತ್ರಿಕ ವಿಶ್ವವಿದ್ಯಾಲಯ
ಬೆಳಗಾವಿ, ಕರ್ನಾಟಕ

A DBMS MINI- PROJECT REPORT
ON

“SCHOOL DATABASE MANAGEMENT SYSTEM”

Submitted to Visvesvaraya Technological University in partial fulfillment of the requirement for the award of Bachelor of Engineering degree in Computer Science and Engineering.

Submitted by

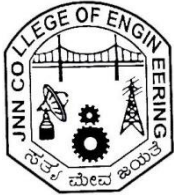
Name	USN
AISHWARYA S	4JN20CS004
CHANDANA C S	4JN20CS025
JEEVITHA V	4JN20CS040
MAHATHI KASHYAP	4JN20CS052

Under the guidance of

Mrs. Sowmya D. M.Tech, (Ph.D)
Asst. Professor, Dept. of CS&E,
JNNCE, Shivamogga.

Mr. Mallesh Kumar K S B.E, M.Tech
Project Manager
Centre for Innovation & Entrepreneurship
JNNCE, Shivamogga.

Department of Computer Science & Engineering
Jawaharlal Nehru New College of Engineering
Shivamogga - 577 20



January 2023

National Education Society ®



Jawaharlal Nehru New College of Engineering

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that the DBMS - MINI PROJECT entitled

“SCHOOL DATABASE MANAGEMENT SYSTEM”

Submitted by

Name	USN
AISHWARYA S	4JN20CS004
CHANDANA C S	4JN20CS025
JEEVITHA V	4JN20CS040
MAHATHI KASHYAP	4JN20CS052

Students of 5th semester B.E. CS&E, in partial fulfillment of the requirement for the award of degree of Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University, Belagavi during the year 2022-23.

Signature of Guides

Mrs. Sowmya D. M.Tech, (Ph.D)
Asst. Professor, Dept. of CS&E,
JNNCE, Shivamogga.

Mr. Mallesh Kumar K S B.E, M.Tech
Project Manager
Centre for Innovation & Entrepreneurship
JNNCE, Shivamogga.

Signature of HOD

Dr. Poornima K M M.Tech, Ph.D.
Prof. & Head, Dept. of CS&E

Examiners: 1. ----- 2. -----

ABSTRACT

The project titled **“SCHOOL DATABASE MANAGEMENT SYSTEM”** is designed with the motive of maintaining all the database of the concern. The basic aim of the project is developing a system, which is very simple, user friendly, easily retrieval and simple access. The project has been developed using NAETBEANS as front end and MYSQL as back end for the school database Management.

This project is insight into the design and implementation of a School database Management System. The primary aim of is to store the data record of students and efficiency in the easy retrieval of data. Today management is one of the most essential features of all form. Management provides sophistication to perform any kind of task in a particular form. This is school database management system; it is used to manage most school related activities.

ACKNOWLEDGEMENT

On presenting the Database Management Systems Mini – Project report on “**SCHOOL DATABASE MANAGEMENT SYSTEM**”, I feel great to express my humble feelings of thanks to all those who have helped me directly or indirectly in the successful completion of the project work.

We would like to thank our respected guides **Mrs. Sowmya D**, Assistant Professor, Dept. of CS & E, and **Mr. Mallesh Kumar K S**, Project Manager, Centre for Innovation & Entrepreneurship, who has helped us a lot in completing this task, for their continuous encouragement and guidance throughout the project work.

We would like to thank **Dr. Poornima K M**, Professor and Head, Dept. of CSE, JNNCE, Shivamogga and **Dr. K Nagendra Prasad**, Principal, JNNCE, Shivamogga for all their support and encouragement.

We are grateful to Department of Computer Science and Engineering and our institution Jawaharlal Nehru New College of Engineering and for imparting me the knowledge with which we can do my best.

Finally, we would also like to thank the whole teaching and non-teaching staff of the Computer Science and Engineering Dept.

Thanking you all,

Name	USN
Aishwarya S	4JN20CS004
Chandana C S	4JN20CS025
Jeevitha V	4JN20CS040
Mahathi Kashyap	4JN20CS052

TABLE OF CONTENT

Abstract	i
Acknowledgement	ii
List of Figures	iv
Chapter 1. Introduction	1-4
1.1 Overview of DBMS	1
1.2 History	1
1.3 Applications of DBMS	2
1.4 Problem statement	3
1.5 Objectives of the Project	4
1.6 Organization of Report	4
 Chapter 2. Requirement Analysis and Design	 5-6
2.1 Basic definition	5
2.2 Advantage	5
2.3 Requirement analysis	5-6
 Chapter 3. Design and Implementation	 7-16
3.1 ER Diagram	8
3.2 Schema Diagram	9
3.3 Pseudocode	10-16
 Chapter 4. Results and Snapshots	 17-26
 Conclusion and Future scope	 27
 References	 28

List of Figures

Figure No.	Title	Page No.
3.1	ER diagram	8
3.2	Schema diagram	9
4.1	Dashboard	17
4.2	Login	17
4.3	Teacher Dashboard	18
4.4	Teacher profile	18
4.5	Attendance marking	19
4.6	Marks entry	19
4.7	Student Dashboard	20
4.8	Student profile	20
4.9	Student attendance	21
4.10	Student marks	21
4.11	Employee dashboard	22
4.12	Employee profile	22
4.13	Fees	23
4.14	Backend tables	23
4.15	Attendance	24
4.16	Marks	24
4.17	Login successful	25
4.18	Insert	25
4.19	Delete	26
4.20	Update	26

CHAPTER 1

INTRODUCTION

Database management system or DBMS is a software designed to assist in managing and utilizing large collection in data, and the need of such system, as well as their use, is growing rapidly. The Alternative to using a DBMS is used to hoc approaches that do not carry over from one application to another.

1.1 Overview of DBMS

A database management system stores data in such a way that it becomes easier to retrieve, manipulate, and produce information. Database is a collection of related data and data is a collection of facts and figures that can be processed to produce information. The area of the Database Management system is microcosm of computer science in general. The issues addressed and the technique used to span a wide spectrum, including languages, object orientation and other programming paradigm, compilation, operating system, concurrent programming, data structures, algorithms, theory, parallel and distributed systems user interface, expert systems and artificial intelligence, statistical techniques, and dynamic programming.

1.2 History

From the earliest days of computers, storing and manipulating data have been a major application focus. The first general purpose DBMS was designed by Charles Bachman at General Electric in the early 1960s was called The Integrated Data Store. It formed the basis for the network data model, which was standardized by the Conferences on Data Systems Languages (CODASYL) and strongly influenced database systems through the 1960s. Bachman was the first recipient of ACM's Turing Award (the computer science equivalent of a Nobel prize) for work in the database Area; he receives the award in 1973.

In the late 1960's IBM developed the Information Management System (IMS) DBMS, used even today in many major installations. IMS form, the basis for an alternative data representation framework called the hierarchical data model. The SABRE system for making airlines reservation was jointly developed by American

Airlines and IBM around the same time, and it allowed several people to access the same data through.

An interesting phenomenon is the emergence of several enterprise resource planning (ERP) and management resources planning (MRP) packages, which add a substantial layer of application – oriented features on top of a DBMS widely used packages include systems from Bann, Oracle, PeopleSoft, SAP and Siebel.

1.3 Applications of DBMS

Nowadays DBMS are used in almost all the areas ranges from science, engineering, medicine, business, industry, government, art, entertainment, education and training.

DBMS in the field of Library Management System

There are thousands of books in the library so it is very difficult to keep records of all the books in a copy or register. DBMS is used to maintain all the information related to book issue date, name of book, author and availability of book.

DBMS in the field of Banking

Another major application is the banks. Thousands of transactions through daily can do this without going bank. To manage such huge transactions is just because of DBMS that manages bank transactions.

DBMS in the field of universities and colleges

Examinations are done online today and universities and colleges maintain all these records through DBMS. Student's registrations details, results, courses and grades all the information is stored in database.

DBMS in the field of Telecommunications

Any telecommunication company cannot even think about their business without DBMS. DBMS is required for these companies to store the call details and monthly postpaid bills.

DBMS in the field of Online Shopping

Online shopping has become a big trend of these days. No one wants to go to shops and to waste his time. Everyone wants to shop from home. So, all these products are added and sold only with the help of DBMS. Purchase information, invoice and payment all these are done with the help of DB.

DBMS in the field of Military

Military keeps records of millions of soldiers and it has millions of files that should be kept secure and safe. As DBMS provides a big security assurance to military information so it is widely used in militaries. One can easily search for all the information about anyone within seconds with the help of DBMS.

1.4 Problem Statement

School database management is a project that manages and stores school information electronically according to school's need. The system helps teachers to keep a constant track of all student's data. It becomes necessary for school to keep a continuous check on the student's performance and attendance status. This task if carried out manually will be tedious and includes chances of occurrence of mistakes. These errors are avoided by allowing the system to keep track of information such as student ID, student name, DOB, address, parent info, academic performance, attendance. And thus, there is no need to keep manual track of this information which thereby avoids chances of mistakes. Thus, this system reduces manual work to a great extent allows smooth flow of library activities by removing chances of errors in the details

1.5 Objectives of the project

The main objectives are,

- It describes how School database Management System is Developed to manage database of the SCHOOL.
- To Create a database of the student, teaching and non-teaching staff.
- To Evaluate the need for a management perspective to improve outcomes of the school.
- To list all the student and staff information that are available in school.

1.6 Organization of the report

This section deals with the Introduction and organization of the project report. Chapter 2 discusses the Specific to the problem-Requirement Analysis-Design. Chapter 3 discusses the Design and Implementation. Chapter 4 gives information about the snapshot and results Chapter 5 include conclusion and future scope. Chapter 6 gives the references of the project.

Chapter 2

REQUIREMENT ANALYSIS AND DESIGN

This chapter includes the requirements for the development of the project. These requirements describe high level system design, software requirements etc.

2.1 Basic Definition

School database Management System is mainly designed to maintain records of the students and also patient information record.

2.2 Advantages

- This system decreases the chance of error.
- This system requires less time for completion of any work.
- It is used to maintain the information such as student's details, marks and attendance.
- Work load and man power is very fast.

2.3 Requirement Analysis

SOFTWARE

Windows10 is a personal computer operating system developed and released by Microsoft as part of the windows NT family of operating systems.it was released on July 29, 2015. It is the first version of windows that receives ongoing feature updates. Devices in enterprise environments can receive these updates at a slower pace, or use long-term support milestones that only receive critical updates, such as security patches etc.

BACK END

Interactive enterprise manager screens display details about a SQL statement. This includes the SQL text, Top activity by various dimensions, CPU and wait activity over time, key SQL statistics, and execution plans, SQL profiles and SQL

plan baselines will be displayed if they exist, and a monitored execution is displayed, if available.

FRONT END

Java database connectivity (JDBC) is an application programming interface (API) for the programming language java, which defines how a client may access a database. It is java-based data access technology and user for java database connectivity. It is Part of the java standard edition platform, form oracle corporation.it provides methods to query and update data in a database, and is oriented towards relational databases. A JDBC-to-MySQL bridge enables connections to any MySQL accessible data source in the java virtual machine (JVM) host environment.

Software requirements

- Operating system - Windows 10
- Backend - MySQL
- Front end - Java swings (NetBeans IDE 14)
- Platform - JDBC jar

Hardware components

- Processor - Intel core i5
- Processor speed - 2.1 GHz
- Ram - 8 GB
- SSD - 512 SSD

Chapter 3

DESIGN AND IMPLEMENTATION

This chapter describes entities, the attributes and from that how the design has been achieved to provide the ER diagram. It also covers how the schema diagram is evolved.

SCHEMA DESCRIPTION:

TEACHER_PROFILE: It contains Emp_ID, Name, DOB, DOJ, Phone_no, Qualification, Address, Salary

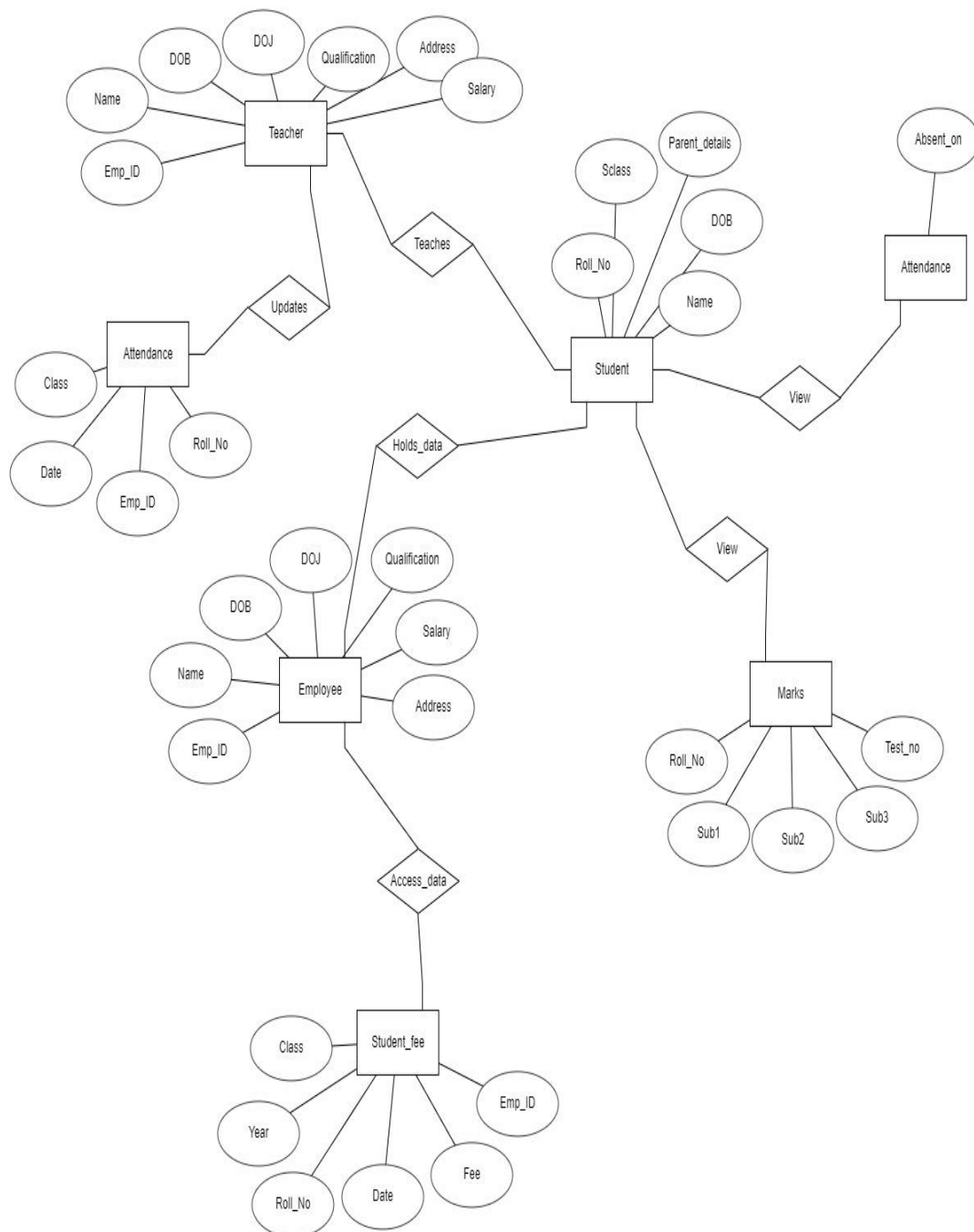
ATTENDANCE: It contains Emp_ID, Class, Date, Roll_no, M_ID

STUDENT_PROFILE: It contains Roll_no, Name, DOB, Address, Parent_details, S_Class

STUDENT_MARKS: It contains Roll_no, Sub1, Sub2, Sub3, Sub4, Sub5, Test_no, M_no

EMPLOYEE_PROFILE: It contains Emp_id, Name, DOB, DOJ, Phone_no, Qualification, Address, Salary

STUDENT_FEE: It contains Class, Year, Roll_no, Fee, Date, Emp_id, F_ID



3.2 SCHEMA DIAGRAM

Figure 3.2 gives schema diagram. It provides the linking of primary key of the different tables and each entity has its own attributes.

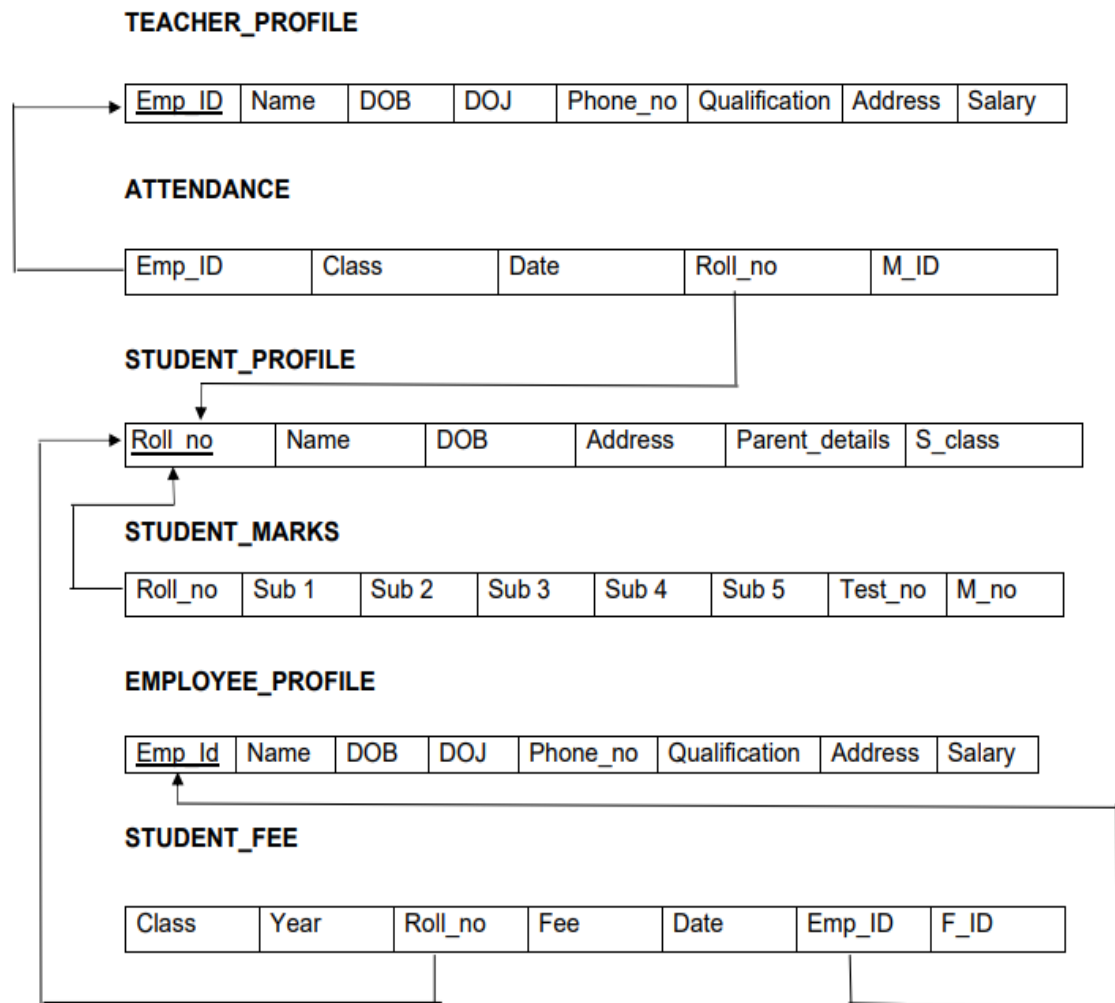


Figure 3.2 SCHEMA DIAGRAM

3.3 PSUDOCODE

INERT:

To insert data into the table, following query is used.

This query is used in student fees.

```
try{
    Class.forName("com.mysql.jdbc.Driver");
    Connection Con =
DriverManager.getConnection("jdbc:mysql://localhost/school", "root", "");
    java.sql.PreparedStatement add;
    add = Con.prepareStatement("insert into
student_fee (Class,Year,Roll_No, Fee, Date, Emp_ID)
values (?, ?, ?, ?, ?, ?)");
    add.setInt(1,
Integer.valueOf(jTextField5.getText()));
    add.setString(2, jTextField6.getText());
    add.setString(3, jTextField8.getText());
    add.setString(4, jTextField10.getText());
    add.setString(5, jTextField7.getText());
    add.setString(6, jTextField9.getText());
    int row =add.executeUpdate();
    JOptionPane.showMessageDialog(this, "Fee
details Saved");
    Displayfee();

} catch (Exception e) {
    e.printStackTrace();
}
```

DELETE:

To delete data from the table, following query is used.

This query is used in student fees.

```
try{
    DefaultTableModel model =
(DefaultTableModel)Displayfee.getModel();
    // get the selected row index
    int selectedRowIndex = Displayfee.getSelectedRow();
    int Class1=Integer.valueOf(model.getValueAt(selectedRowIndex,
0).toString());
    Class.forName("com.mysql.jdbc.Driver");
    Connection Con =
DriverManager.getConnection("jdbc:mysql://localhost/school","root","");
    PreparedStatement add=Con.prepareStatement("delete from
student_fee where Class=?");
    System.out.println("delete from student_fee where
Class1="+Class1);
    add.setInt(1,Class1);

    int row =add.executeUpdate();
    JOptionPane.showMessageDialog(this,"Fee_details Deleted");
    Displayfee();

}
catch(Exception e){
    e.printStackTrace();
}
```

UPDATE:

To update data onto the table, following query is used.

This query is used in student fees updation.

```
try{
    int i = Displayfee.getSelectedRow();
    DefaultTableModel model =
(DefaultTableModel)Displayfee.getModel();
    int F_id=Integer.valueOf(model.getValueAt(i, 6).toString());

    Class.forName("com.mysql.jdbc.Driver");
    Connection Con =
DriverManager.getConnection("jdbc:mysql://localhost/school","root","");
    PreparedStatement add=Con.prepareStatement("UPDATE
`student_fee` SET Class=?,Year=?,Fee=?,Date=?,Emp_ID=? WHERE
Roll_no=?");
    if(i >=0)
    {
        model.setValueAt(jTextField5.getText(), i, 0);
        model.setValueAt(jTextField6.getText(), i, 1);
        model.setValueAt(jTextField8.getText(), i, 2);
        model.setValueAt(jTextField10.getText(), i, 3);
        model.setValueAt(jTextField7.getText(), i, 4);
        model.setValueAt(jTextField11.getText(), i, 5);
        JOptionPane.showMessageDialog(this,"Fee_details Updated");
    }
    else
    {
        JOptionPane.showMessageDialog(null,"Error");
    }
}
```

```
}  
catch(Exception e){  
    e.printStackTrace();  
}
```

CHOOSE THE FILE:

To choose the file for uploading, the following query is used.

The query is used in uploading the student's marks.

```
// TODO add your handling code here:  
JFileChooser fileChooser = new JFileChooser();  
fileChooser.setCurrentDirectory(new  
File(System.getProperty("user.home")));  
int result = fileChooser.showOpenDialog(this);  
if (result == JFileChooser.APPROVE_OPTION) {  
    File selectedFile =  
fileChooser.getSelectedFile();  
    path=selectedFile.getAbsolutePath();
```

LOAD FROM EXCEL FILE:

To browse the excel sheet from desktop, the following query is used.

The query is used in uploading the student's marks.

```
try{  
    FileInputStream fis=new FileInputStream(new File(path));  
    Connection Con = null;  
Statement St = null;  
ResultSet RS = null;  
Class.forName("com.mysql.jdbc.Driver");  
    Con=DriverManager.getConnection("jdbc:mysql://localhost/school",  
"root", "");  
//creating workbook instance that refers to .xls file
```

```
HSSFWorkbook wb=new HSSFWorkbook(fis);
//creating a Sheet object to retrieve the object
HSSFSheet sheet=wb.getSheetAt(0);
//evaluating cell type
FormulaEvaluator
formulaEvaluator=wb.getCreationHelper().createFormulaEvaluator();
int cc=0;
for(Row row: sheet)    //iteration over row using for each loop
{
    if(cc==0)
    {
        cc++;
        continue;
    }
    String rollno="";
    int smarks[]=new int[5];
    int cnt=0;
    for(Cell cell: row)    //iteration over cell using for each loop
    {
        switch(formulaEvaluator.evaluateInCell(cell).getCellType())
        {
            case Cell.CELL_TYPE_NUMERIC:    //field that represents numeric cell
            type
            //getting the value of the cell as a number
            //System.out.print(cell.getNumericCellValue()+ "ahan p\t\t");
            smarks[cnt++]=(int)cell.getNumericCellValue();
            break;
            case Cell.CELL_TYPE_STRING:    //field that represents string cell type
```

```
//getting the value of the cell as a string
//System.out.print(cell.getStringCellValue()+ "\t\t");
    rollno=cell.getStringCellValue();
break;
}
}
cnt=0;
System.out.println(rollno);
int testno=Integer.parseInt(jTextField2.getText().toString());
String q="delete from student_marks where Roll_No='"+rollno+"' and
Test_no="+testno;
Statement dins=Con.createStatement();
    dins.executeUpdate(q);
String Query= "Insert into student_marks(Roll_No,"
    + "Sub1,Sub2,Sub3,Sub4,Sub5,Test_no) values ('"+rollno+"',"
+smarks[0]+",""+smarks[1]+",""+smarks[2]+",""+smarks[3]+",""+smarks[4]
+",""+testno+" ")";
    System.out.println (Query);
    Statement ins=Con.createStatement();
    ins.executeUpdate(Query);
    St=Con.createStatement();
    RS=St.executeQuery("select * from student_marks");
    jTable1.setModel(DbUtils.resultSetToTableModel(RS));
    jTable1.setEnabled(false);
//System.out.println();
}
}
catch(Exception e){ }
```

ACCESS THE DATA IN RANGE FORMAT USING BETWEEN:

To access the date and marks on which day he was absent as well as viewing his marks scored on each test, the following query is used.

This is used in students viewing of marks and attendance.

```
String s1=new String();
    s1=jTextField1.getText();
    System.out.println(s1);
    String s2=new String();
    s2=jTextField2.getText();
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection Con=DriverManager.getConnection("jdbc:mysql://localhost/school",
"root", "");

        Statement St=Con.createStatement();
        ResultSet RS=St.executeQuery("select * from attendance where
Roll_no='"+login.tid+"' and Date BETWEEN '"+s1+"' and '"+s2+"'");
        Object[][] data=new Object[10][1];
        Object[] columnNames={"Absent_On"};
        int cnt=0;
        while(RS.next())
        {
            String date=RS.getString("Date");
            data[cnt][0]=date;
            cnt++;
        }
        DefaultTableModel model=new DefaultTableModel(data,columnNames);
        jTable1.setModel(model);
    }
    catch(Exception e){ }
```

Chapter 4

RESULTS AND SNAPSHOTS

This chapter includes result and snapshots of the implementation.

Figure 4.1 describes the Dashboard which contains teachers, student and employee.



Fig: 4.1 DASHBOARD

Figure 4.2 describes the login page



Fig 4.2: LOGIN PAGE

Figure 4.3 describes Teacher Dashboard which contains profile, attendance, marks.

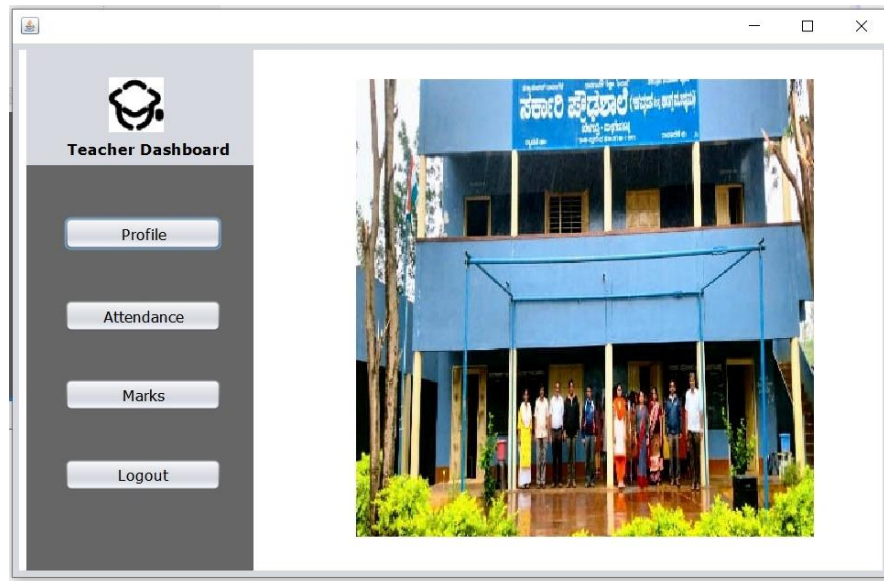


Fig: 4.3 TEACHER DASHBOARD

Figure 4.4 describes teacher profile which has Name, address and phone no.

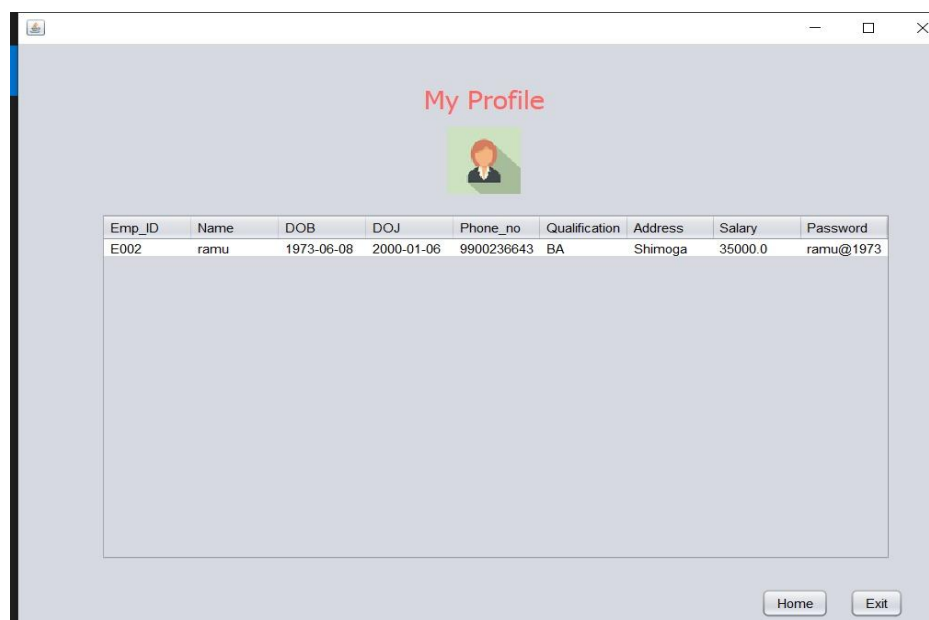


Fig: 4.4 TEACHER PROFILE

Figure 4.5 describes marking of attendance by teacher for students.

Class: 7
Date: 2023-01-19

Buttons: Load, Mark Absent

RollNo	Name	Status
R102	Soma	<input type="checkbox"/>
R103	Bhama	<input checked="" type="checkbox"/>

Fig: 4.5 ATTENDANCE MARKING

Figure 4.6 describes updating the marks of the students by browsing from excel sheet.

Test No: 1

Buttons: Select File, Upload

Roll_no	Sub1	Sub2	Sub3	Sub4	Sub5	Test_no	M_no
R102	20	21	22	21	23	1	32
R103	20	22	22	21	18	1	33
R102	20	21	22	21	23	2	37
R103	20	22	22	21	18	2	38

Home

Fig: 4.6 MARKS ENTRY

Figure 4.7 describes student dashboard.

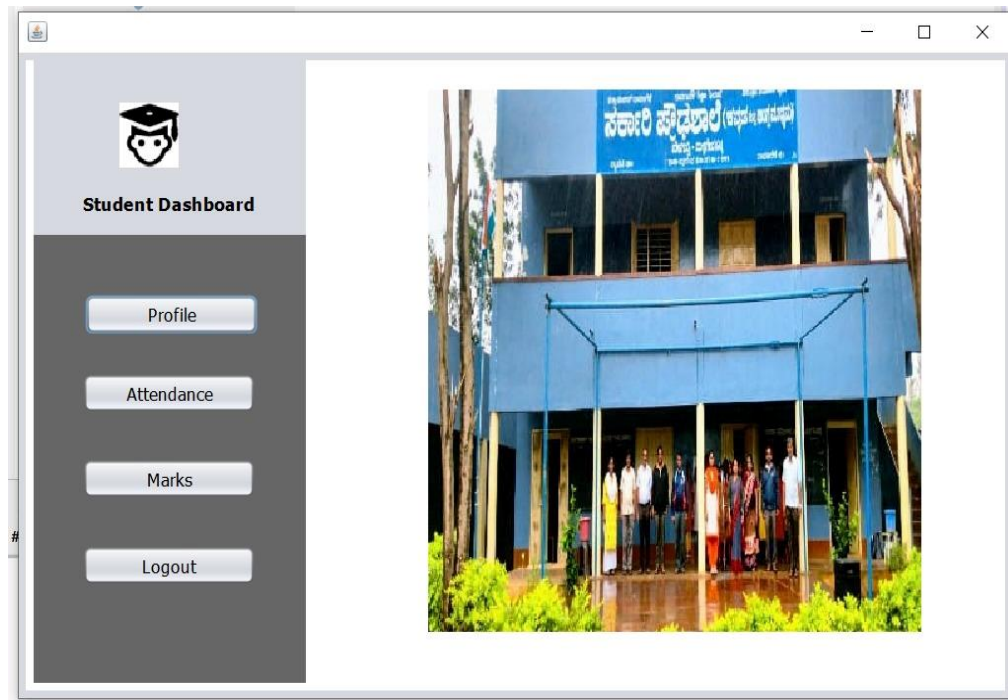


Fig: 4.7 STUDENT DASHBOARD

Figure 4.8 describes student profile which includes Name, Roll no, Parent details.

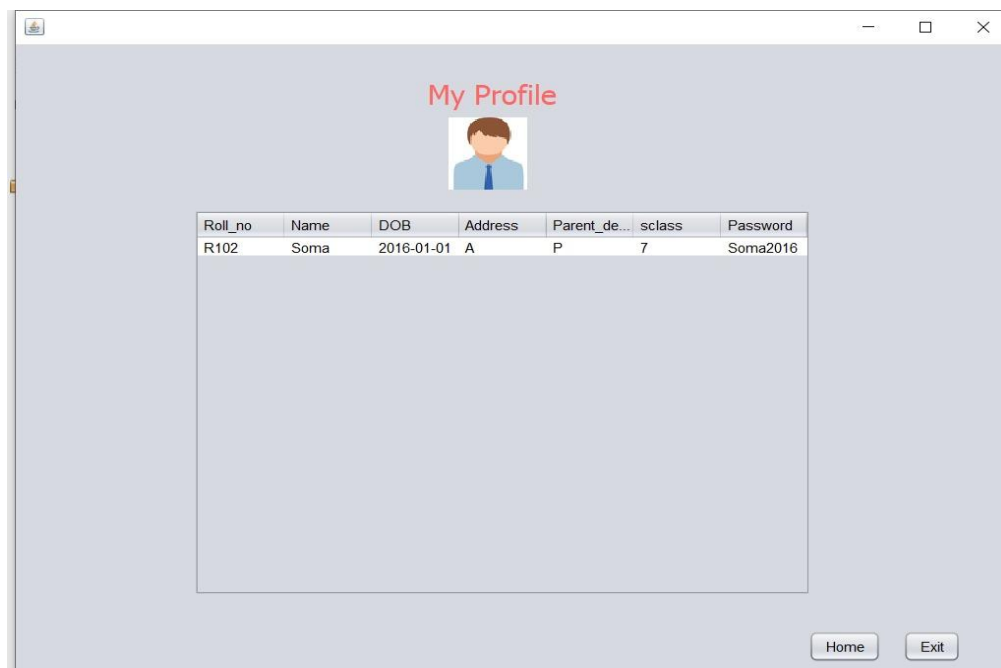


Fig: 4.8 STUDENT PROFILE

Figure 4.9 describes student's attendance which provides the date he was absent on.

Attendance

Date_from: 2023-01-10 Date_to: 2023-01-13

Search

Absent_On
2023-01-10
2023-01-13

Back

Exit

Fig: 4.9 STUDENT ATTENDANCE

Figure 4.10 describes the student's marks which provides the marks scored by the student on each subject for the test no he attended.

Marks

Test_no: 1 Go

Sub1	Sub2	Sub3	Sub4	Sub5
20	21	22	21	23

Back

Exit

Fig: 4.10 STUDENT MARKS

Figure 4.11 describes employee dashboard which has his profile and fee.

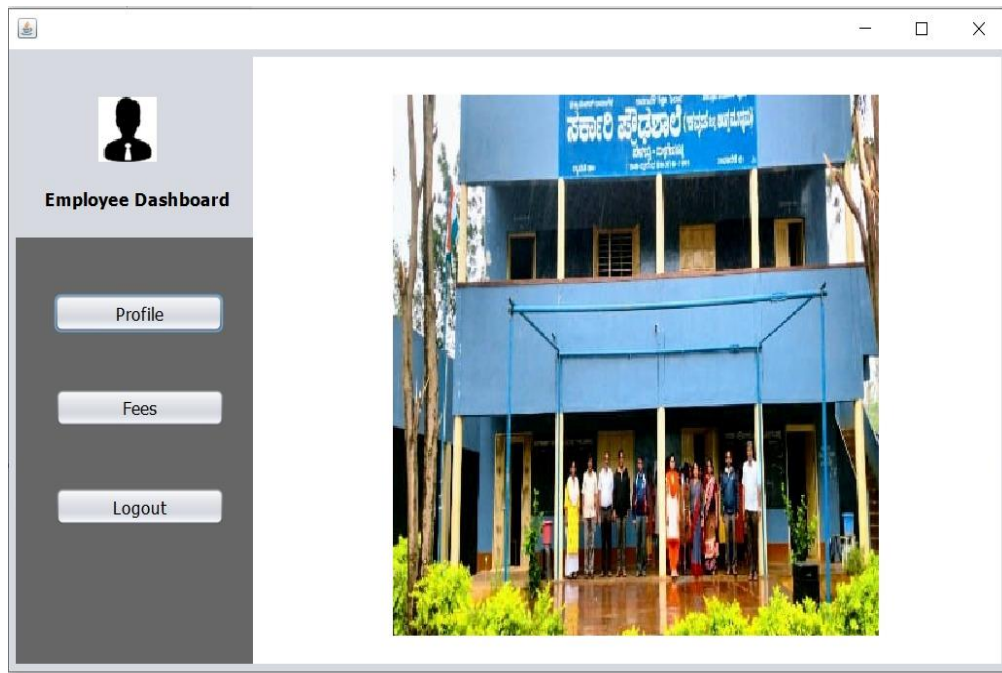


Fig: 4.11 EMPLOYEE DASHBOARD

Figure 4.12 describes employee profile which includes his ID, Name, Address, Phone no.

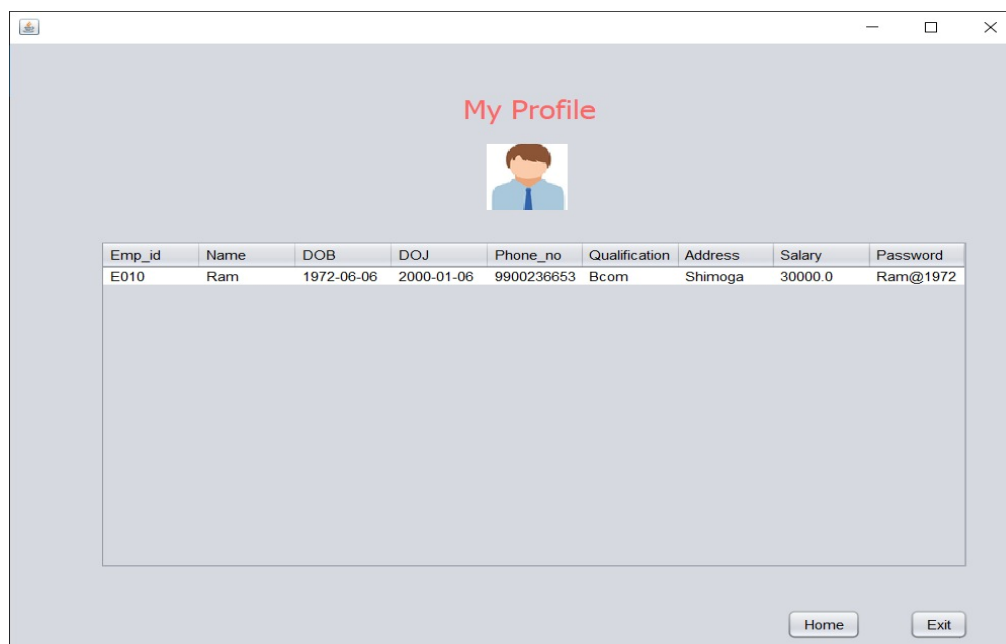


Fig: 4.12 EMPLOYEE PROFILE

Figure 4.13 describes fee details of student paid to school during admission.

Fees

Class:

Year:

Roll_No:

Fee:

Date:

Emp_ID:

Class	Year	Roll_No	Fee	Date	Emp_ID	F_id
7	2023	R103	5000.0	2023-01-02	E011	58
7	2023	R102	5000.0	2023-01-01	E011	59

Fig: 4.13 FEES

Backend Database:

Figure 4.14 describes the tables created in the backend.

Filters

Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> attendance		4	InnoDB	utf8mb4_general_ci	48.0 K B	-
<input type="checkbox"/> employee_profile		2	InnoDB	utf8mb4_general_ci	16.0 K B	-
<input type="checkbox"/> student_fee		1	InnoDB	utf8mb4_general_ci	48.0 K B	-
<input type="checkbox"/> student_marks		4	InnoDB	utf8mb4_general_ci	32.0 K B	-
<input type="checkbox"/> student_profile		2	InnoDB	utf8mb4_general_ci	16.0 K B	-
<input type="checkbox"/> teacher_profile		2	InnoDB	utf8mb4_general_ci	16.0 K B	-
6 tables	Sum	15	InnoDB	utf8mb4_general_ci	176.0 K B	0 B

☐ Check all With selected

Table name: Number of columns:

Fig: 4.14 BACKEND TABLES

Figure 4.15 describes the student's data for attendance.

phpMyAdmin

Recent

Favorites

New

company

information_schema

mysql

order

performance_schema

phpmyadmin

school

school

New

attendance

employee_profile

student_fee

student_marks

student_profile

teacher_profile

school_database

student_database

test

Server: 127.0.0.1

Database: school

Table: attendance

Browse

Structure

SQL

Search

Insert

Export

Import

Privileges

Operations

Triggers

Showing rows 0 - 3 (4 total, Query took 0.0005 seconds)

SELECT * FROM 'attendance'

Profiling

Edit inline

Edit

Explain SQL

Create PHP code

Refresh

Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

Extra options

+

-

T

→

▼

Emp_ID

Class

Date

Roll_no

M_id

☐

Edit

Copy

Delete

E001

7 2023-01-17

R102

6

☐

Edit

Copy

Delete

E001

7 2023-01-10

R102

7

☐

Edit

Copy

Delete

E001

7 2023-01-13

R102

8

☐

Edit

Copy

Delete

E001

7 2023-01-19

R103

10

↑

Check all

With selected:

Edit

Copy

Delete

Export

Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

Query results operations

Print

Copy to clipboard

Export

Display chart

Create view

Fig: 4.15 ATTENDANCE

Figure 4.16 describes the marks data at the backend.

phpMyAdmin

Server: 127.0.0.1 > Database: school > Table: student_marks

Recent

Favorites

New

company

information_schema

mysql

order

performance_schema

phpmyadmin

school

school

New

attendance

employee_profile

student_fee

student_marks

student_profile

teacher_profile

school_database

student_database

test

Browse

Structure

SQL

Search

Insert

Export

Import

Privileges

Operations

Triggers

Showing rows 0 - 3 (4 total, Query took 0.0003 seconds.)

SELECT * FROM 'student_marks'

Profiling

Edit inline

Edit

Explain SQL

Create PHP code

Refresh

Show all

Number of rows: 25

Filter rows:

Search this table

Sort by key: None

Extra options

Roll_no

Sub1

Sub2

Sub3

Sub4

Sub5

Test_no

M_no

Edit

Copy

Delete

R102

20

21

22

21

23

1

32

Edit

Copy

Delete

R103

20

22

22

21

18

1

33

Edit

Copy

Delete

R102

20

21

22

21

23

2

37

Edit

Copy

Delete

R103

20

22

22

21

18

2

38

Check all

With selected:

Edit

Copy

Delete

Export

Show all

Number of rows: 25

Filter rows:

Search this table

Sort by key: None

Query results operations

Print

Copy to clipboard

Export

Display chart

Create view

Fig: 4.16 MARKS

FRONTEND DESIGN:

Figure 4.17 describes login successful message if the user login with the correct user_ID and password.



Fig: 4.17 LOGIN SUCCESSFUL

Figure 4.18 describes the fee details saved, which is the insertion operation done when we add a new data to that field.

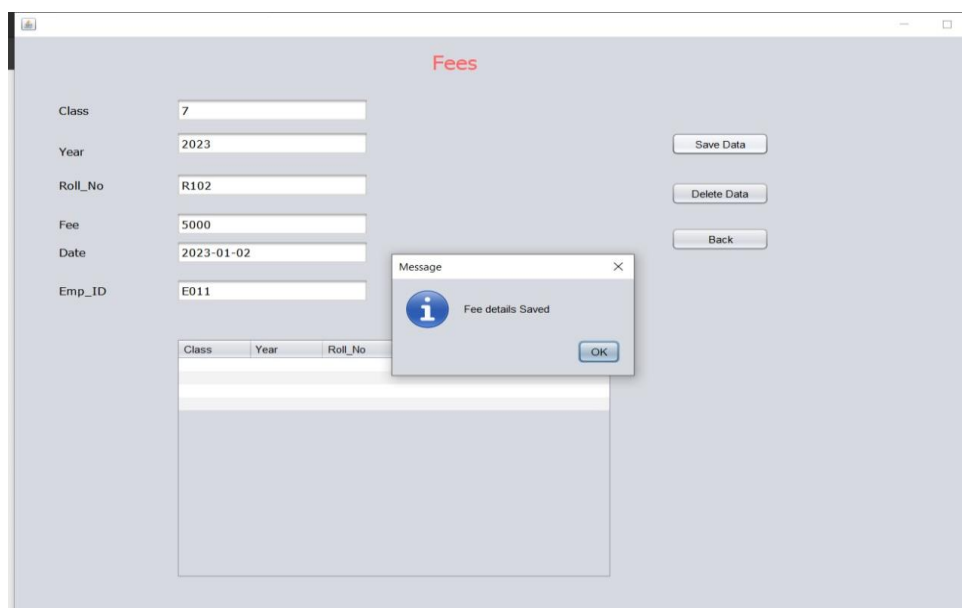


Fig: 4.18 INSERT

Figure 4.19 describes the delete message when the data inserted is wrong and that need to be discarded.

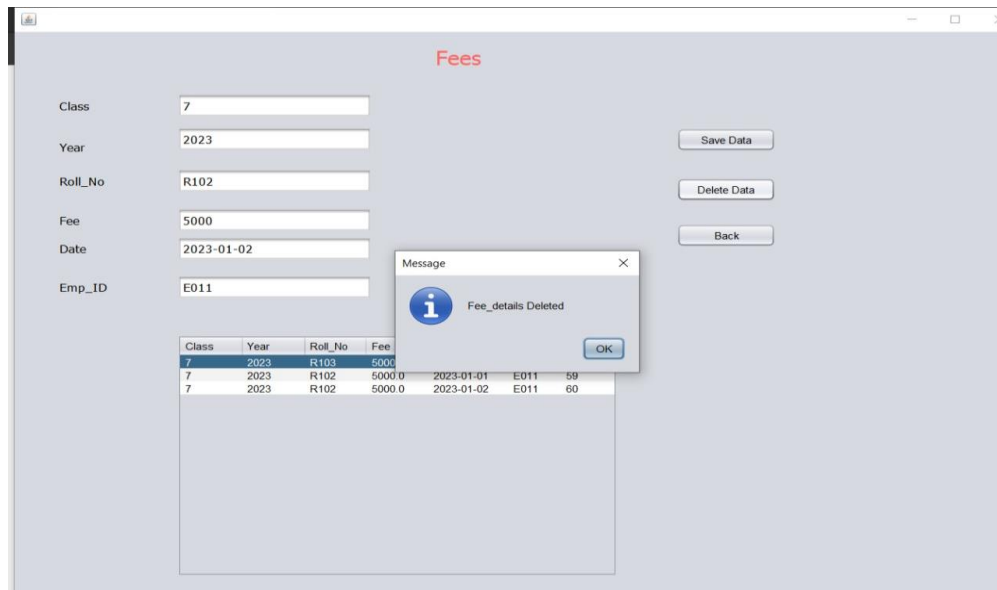


Fig: 4.19 DELETE

Figure 4.20 describes the fee details updated message; it is used when the student's information needs to be modified.

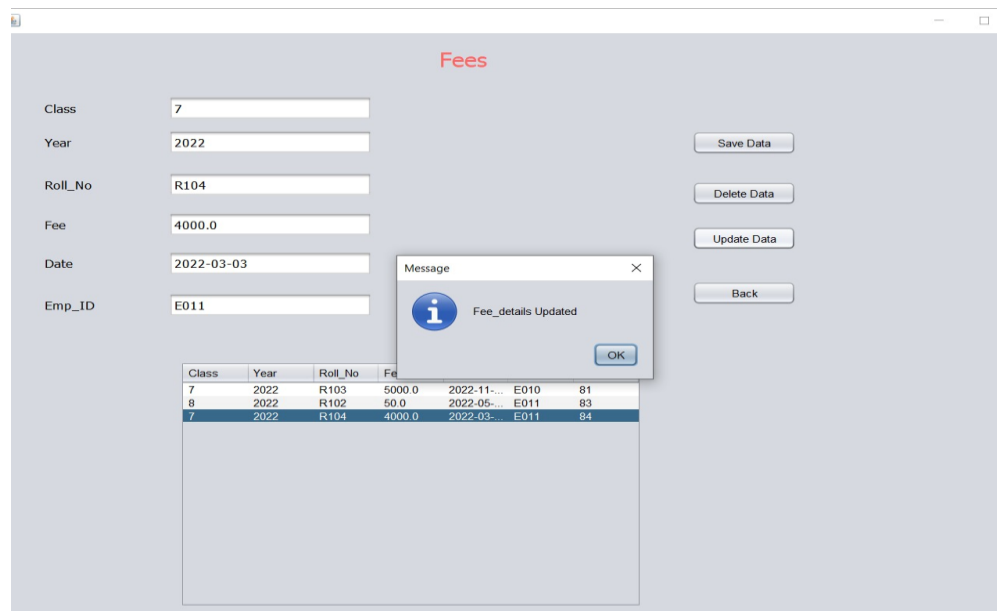


Fig: 4.20 UPDATE

CONCLUSION

Today management is one of the most essential features of all form. Management provides sophistication to perform any kind of task in a particular form. This is school database management system; it is used to manage most school related activities. The primary aim of is to provide flexible use of school record.

We may also conclude that by using school software, processing the new records and appending the data can be done quickly and simply with just a few keystrokes or mouse clicks with new, easy to learn and use Graphical User Interface (GUI) school management solution. This website provides a computerized version of school management system which will benefit the students as well as the staff of the school, makes entire process online where teachers can search access student record and update the student academic details. This project is to design an efficient maintain database system and it will make easy to access the records based on requirement.

Future Scope

The future scope of this project **SCHOOL DATABASE MANAGEMENT SYSTEM** is very spacious. There are many additional features, which are planned to be incorporated during the future enhancements of this project maybe provide facility that many more features such as online lectures video tutorials can be added by teachers as well as online assignments submission facility. Project also fulfills each user need in the best way possible. The Future version of System there is some point that we may implement on them developing an application for android devices that works on the same database using MySQL.

REFERENCES

- [1]. Ramez Elmasri and Shamkant B. Navathe, “Database systems Models, Languages, Design and Application Programming” , Pearson, 7th Edition, 2017.
- [2]. Raghu Ramakrishnan, and Gehrke, “Database management systems” , McGraw Hill, 3rd Edition, 2014.
- [3]. Herbert Schildt: JAVA the Complete Reference, 7th/9th Edition, Tata McGraw Hill, 2007.
- [4]. Jim Keogh: J2EE-TheCompleteReference, McGraw Hill, 2007.