Priyanka School Management System

A Project Report Submitted

in Partial Fulfillment of the Requirements

for the Degree of

Bachelor of Computer Application

Submitted By

Priyanka Kumari

20BCA012

Faculty of Computing and Information Technology

Usha Martin University, Ranchi

June, 2023

BONAFIDE CERTIFCATE

Certified that this project report titled “Priyanka School Management System” is the bonafide work of Ms. Priyanka Kumari who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

(Signature of the concerned Supervisor of the Organization with Organization Seal)

(Certificate to be countersigned by the HOD.)

DECLARATION

I do hereby declare that this report entitled “Priyanka School Management System”, submitted by Priyanka Kumari, bearing Roll No:-20BCA012 in the fulfillment of the requirement for the degree of Bachelor of Computer Application to Usha Martin University, Ranchi, is my own and it is not submitted to any other institute.

Priyanka Kumari

20BCA012

CERTIFICATE

This is to certify that entitled “Priyanka School Management System” being submitted by Priyanka Kumari, bearing Roll No- 20BCA012, in the fulfillment of the requirement for the degree of Bachelor of Computer Application to Usha Martin University, Ranchi, is a bonafide work carried out under my/our supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

Dr. Md. Amir Khusru Akhtar

Guide

Dr. Sharmistha Roy

HOD

External Examiner

ACKNOWLEDGEMENTS

I express my deepest sense of gratitude to my guide Dr. Amir Khusru Akhtar and co-guide Ranjan Ganguli, Faculty of Computing and Information Technology, Usha Martin University, Ranchi, for suggesting the subject of work and constant supervision throughout this work. His co-operation and timely suggestions have been unparalleled stimuli for me to travel eventually towards the completion of this project report. Indeed his continuous involvement has helped me in bringing of this project work which otherwise would have remained a distant dream.

I am indeed thankful to Dr. Sharmistha Roy, HOD, Faculty of Computing and Information Technology, Usha Martin University, Ranchi, for giving me permission to carry out my project work. I would like to express my gratitude to all teaching and non teaching-staff members of Faculty of Computing and Information Technology, Usha Martin University, Ranchi, for their co-operation in my work.

Priyanka Kumari

20BCA012

Abstract

Priyanka School Management System helps headmasters to get the most accurate information to make more effective decisions. Teachers and headmasters gain time saving administrative tools, parents gain immediate access to their children’s grades and students can track their own progress. Priyanka School Management System[1] equipped features makes it possible to generate schedules and reports in minutes and to retrieve attendance records, grade checks, report cards, transcripts, and form letters in just a few clicks. Priyanka School Management Systems helps Teachers to complete grade book, track students attendance, input class notes, create lesson plans and detailed reports, and communicate with other staff members, students, and parents all via e-mail. It also helps Students to access assignments and tests, and view attendance records, grades, report cards, and progress reports all online. They also can communicate through mail and forums with teachers and other students online. My School (PSMS) is a web enabled application developed in Java and powerful MYSQL database back end. To implement My School application[2], schools do not need expensive hardware and software, they just need an internet connection and desktops. Our system works as a centralized database and application that schools can easily access the system from anywhere based on the login credentials. My School is a platform independent system that virtually any user can access from anywhere through a standard internet accessible system.

Contents

Chapters

List of Tables

List of Figures

List of Abbreviations

1.	PSMS	Priyanka School Management System
2.	SRS	Software Requirements Specification
3.	DDS	Design Documentation Specification
4.	PHP	Hypertext Preprocessor
5.	UAT	User Acceptance Testing
6.	ER	Entity Relationship
7.	DFD	Data Flow Diagram
8.	ERP	Enterprise Resource Planning

Chapter 1

1. Introduction

The title of the project is “Priyanka School Management System[3]”. This project will handle all the activities that a modern school requires to computerize its day-to-day jobs. It provides facilities to keep records of student, fees, teaching and non-teaching staff with all required details. It has facilities to generate various types of reports, which are required by the management during normal business operations to operate the business effectively. Education system forms the backbone of every nation. And hence it is important to provide a strong educational foundation to the young generation to ensure the development of open-minded global citizens securing the future for everyone. Advanced technology available today can play acrucial role in streamlining education-related processes to promote solidarity among students, teachers, parents and the school staff. Education is central to development. With this aim currently our government has given special emphasis to the educational sector and school improvement activities such as continuous professional development for teachers, training and upgrading teachers and capacitating schools with manpower and materials are among the major actions which have been taken in both primary and secondary schools. In order to facilitate and simplify these actions one of the major tool is to have automated school management system. Priyanka School Management System(PSMS)[4] consists of tasks such as registering students, attendance record keeping to control absentees, producing report cards, producing official transcript, preparing timetable and producing different reports for teachers, parents, officials from education bureaus and other stakeholders. Automation is the utilization of technology to replace human with a machine that can perform more quickly and more continuously. By automating PSMS[5] documents that took up many large storage rooms can be stored on few disks. Transcript images can be annotated. It reduces the time to retrieve old transcripts from hours to seconds. However, the school system in the government schools of Addis Ababa is not automated and the record officers generate transcripts and reports manually and the school administrators use their experienced knowledge of miss and hit approaches to prepare timetables.

1.1 Objective

Priyanka School Management system make faculty jobs more accessible by giving them an easy place to find and sort information. This system allows teachers and student managers to follow with their student engagement. The idea is to create a scenario that makes the lives of administration and teachers easier. The project takes most of the activities such as online admission, online hiring of teachers, fee-details, class-details, student-details, teachers-details, etc.

The word Management simply means getting things done by human and other resources of the organization. It involves planning, organizing, actuating, and controlling performance to determine and accomplish set objectives by the use of human beings and other resources. Thus, if we conclude the meaning of Priyanka School Management System, then we can say that it is the procedure of getting school activities done with the help of staff and other resources. The Priyanka School Management Software makes all these tasks online and automatic without having the need to depend on another party. The PSMS is important because it simplifies the school administration and other activities with its features. You can record the attendance, performance, fee payment, etc of students with this software. This provides the users with a single database to store every data in it. Your data is never lost and never used by any other unauthorized party.

Priyanka School Management Systems plays an important role in the current instructional system. School authorities nowadays are using this amazing Priyanka School Management System so as to manage and supply a better educational software system to students effectively[6]. However, maintaining and keeping track of faculty body activities isn't a straightforward method within the aggressive world. It needs toil and infrequently it's long. To better perform the college body activities of instructional institutes and to assure parents the time period progress and security of their kids, instructional institutes utilize Priyanka School Management software systems today. Such applications usually provide several options that facilitate to boost the performance of colleges with minimum effort. The School Management System will do so by avoiding the manual paper works and automation of the many educational and body activities. It also allows us to take a glance at why institutes got to implement it.

Aspects of Priyanka School Management

The school management ERP is a comprehensive management software system that ensures the smooth functioning of daily operations at a reasonable worth. Here are some points that will help you know the aspects of school management:

1. Student information:

It stores the entire data of each and every single student in a single database for the benefit of corroboratory, modifying, or change. Admissions, academics, attendance, examination.

2. Attendance Management

The school group action management module keeps track of the attendance of students. The biometric helps in prompt time and tracking of students that reduces group action calculation errors.

3. Fee Management

This easy school Management package, fee management now's created hassle-free and easy. Parents can pay from anywhere and at any time using online mode of payment.

4. Timetable Management

It also maintains and updates the timetable for students and lecturers. It also sets up daily schoolroom schedules, future events, and vacation notifications.

5. Reports Management

Reports and grades module creates a test analysis report of students that works between students, teachers, and parents. The grading system also helps the parents and as well as teachers to judge students' progress then proceed with necessary mentoring and guidance.

Material management workflows, or flow charts, offer an in-depth study and knowledge of it. The material management method commonly includes the receiving and examination of raw materials also because of the storage of these raw materials before they're going through producing runs to assemble into a finished product. Analyzing material management operations from a close and activity-level perspective not only advantages the material management operation but it also may help with simpler collaboration with the acquisition and producing functions that work closely with material management.

Priyanka School Management System automates the method for administration from registration to admission. It also has reduced the loss of knowledge and extra delays or efforts in looking for the data. This software simply operates through the internet and mobile and on the other hand, it’s the efficient and best software for each school to avail. The principles and ideas of school management are:

- The policies of the school are developed especially for instructional principles and government policies. This also involves the goal of grade school education.
- Strengthen body organization, maintain job responsibilities, and boost work efficiency.
- Follow a democratic approach, create personnel, funds, opinions, rewards. This also creates a penalty public in a trial to guide the school toward a united and harmonious direction.
- Strengthen personnel management, increase employees’ motivation. Look, listen, and understand additional.
- Promote attention to mobile management. Emphasize delicate field designing and also improve the operating environment, and encourage employees’ skilled spirit.

Chapter 2

Hardware and Software Requirement

2.1 Hardware Requirements

System-LENOVO E3340

Device Name-DESKTOP-DRl90M7

RAM-4GB

System Type-64-bit operating system,x64-based processor

2.1 Software Requirements

Operating System-Windows

Front End- Html,CSS,Bootstrap

Backend- JSP

Coding Languages- JAVA

Editor- Eclipse

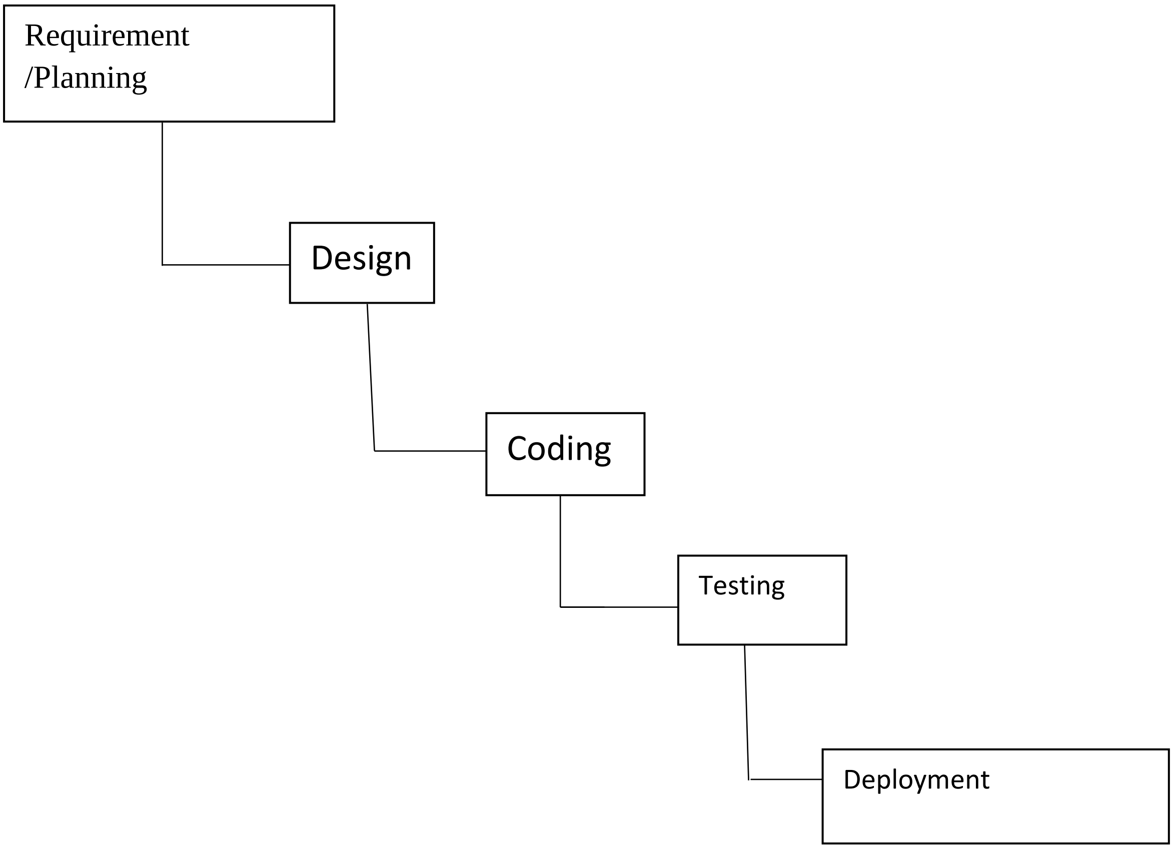
Database- Mysql

Database Server- Xampp Server

Chapter 3

SDLC (Software Development Life Cycle)

Software Development life cycle is a process used by the software industry to design, develop and test high quality software. The SDLC aims to produce a high- quality software that meets or exceeds customer expectations, reaches completion with times and cost estimates.



Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas.

Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

Designing the Product Architecture

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification.

This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product.

A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

Coding or Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle.

Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

Deployment in the Software and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing).

Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

Chapter 4

Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

Types of Software Testing[7]

1. Unit Testing
2. Integration Testing
3. System Testing
4. Regression Testing

Testing is the process of executing a program to find errors. To make our software perform well it should be error-free. If testing is done successfully it will remove all the errors from the software.

Types of Testing:-

1. Unit Testing[8]

Unit testing is a type of software testing that focuses on individual units or components of a software system. The purpose of unit testing is to validate that each unit of the software works as intended and meets the requirements. Unit testing is typically performed by developers, and it is performed early in the development process before the code is integrated and tested as a whole system.

2. Integration Testing

Integration testing is a method of testing how different units or components of a software application interact with each other. It is used to identify and resolve any issues that may arise when different units of the software are combined. Integration testing is typically done after unit testing and before functional testing, and is used to verify that the different units of the software work together as intended.

3. Regression Testing

Regression testing is a method of testing that is used to ensure that changes made to the software do not introduce new bugs or cause existing functionality to break. It is typically done after changes have been made to the code, such as bug fixes or new features, and is used to verify that the software still works as intended.

4. System Testing

System Testing is carried out on the whole system in the context of either system requirement specifications or functional requirement specifications or in the context of both. The software is tested such that it works fine for the different operating systems. It is covered under the black box testing technique.

Chapter 5

System Design

5.1 Entity Relationship Diagram

ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system. It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.

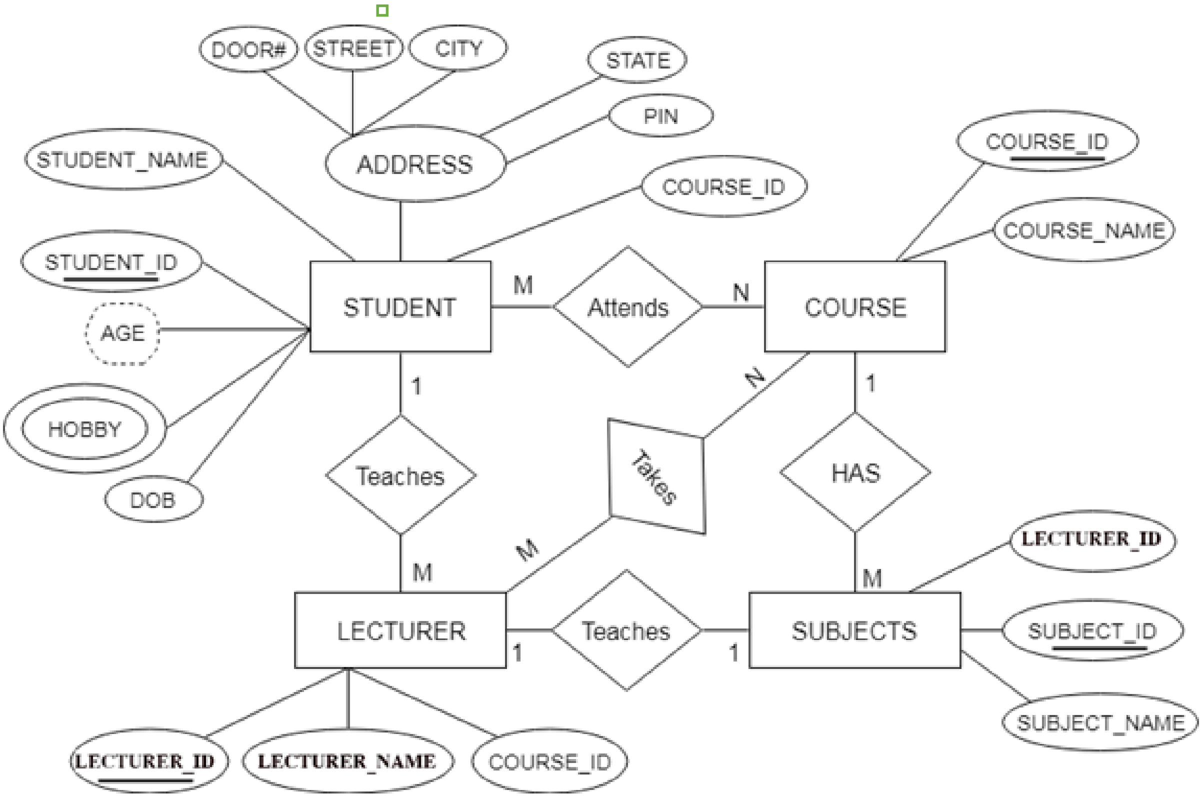


Fig:-ER Model of Priyanka School Management System

5.2 Use case diagram

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

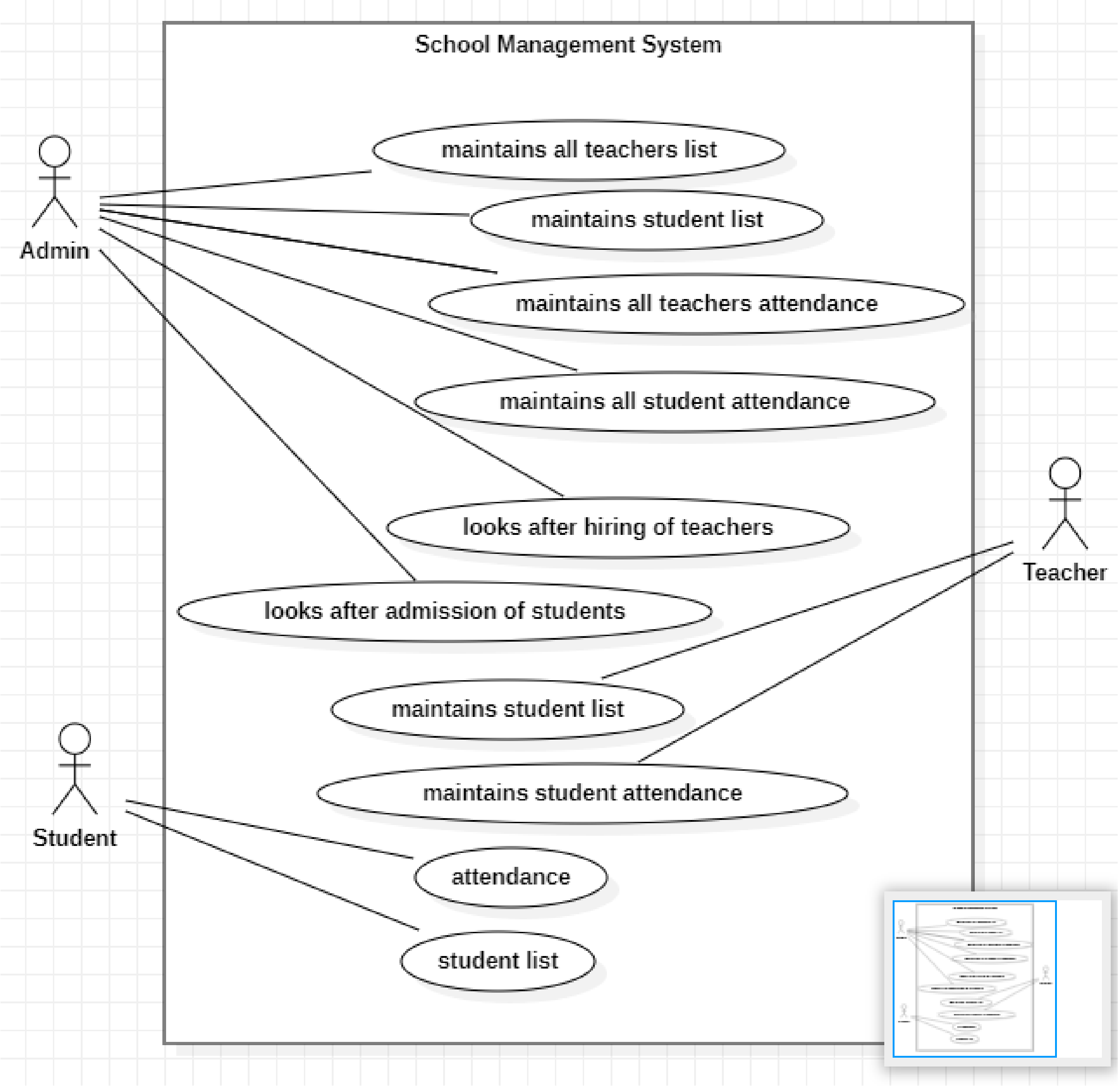

School Management System
Admin
maintains all teachers list
maintains student list
maintains all teachers attendance
maintains all student attendance
looks after hiring of teachers
looks after admission of students
Teacher
maintains student list
maintains student attendance
Student
attendance
student list

Fig 1.2:-Use case diagram of Priyanka School Management System

5.3 Data Flow Diagram

A data flow diagram (DFD)[9] is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement [6]. They are often elements of a formal methodology such as Structured Systems Analysis and Design Method

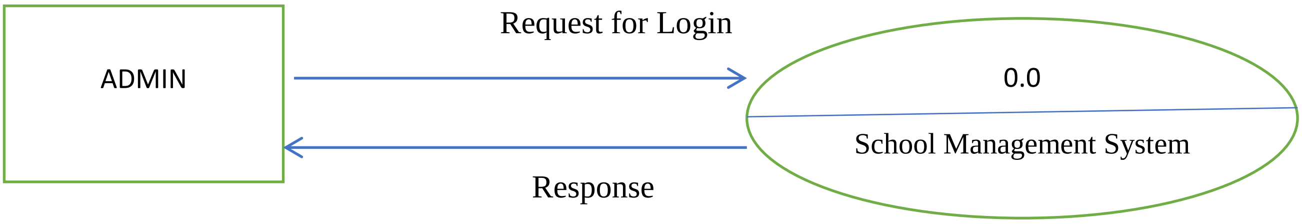


Fig:-0-Level DFD Diagram Of Priyanka School Management System

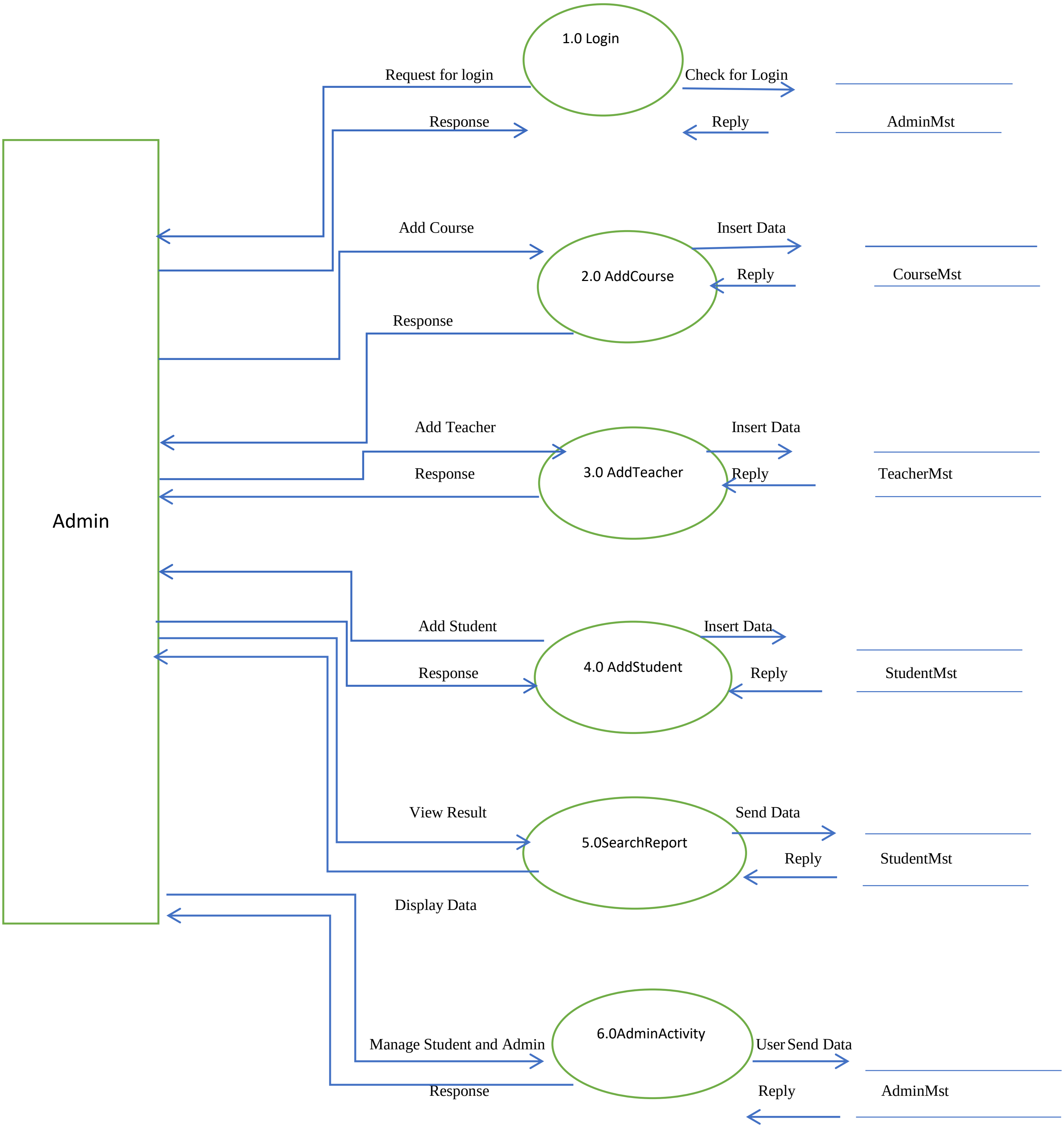


Fig:-1-Level Admin Side DFD Of Priyanka School Management System

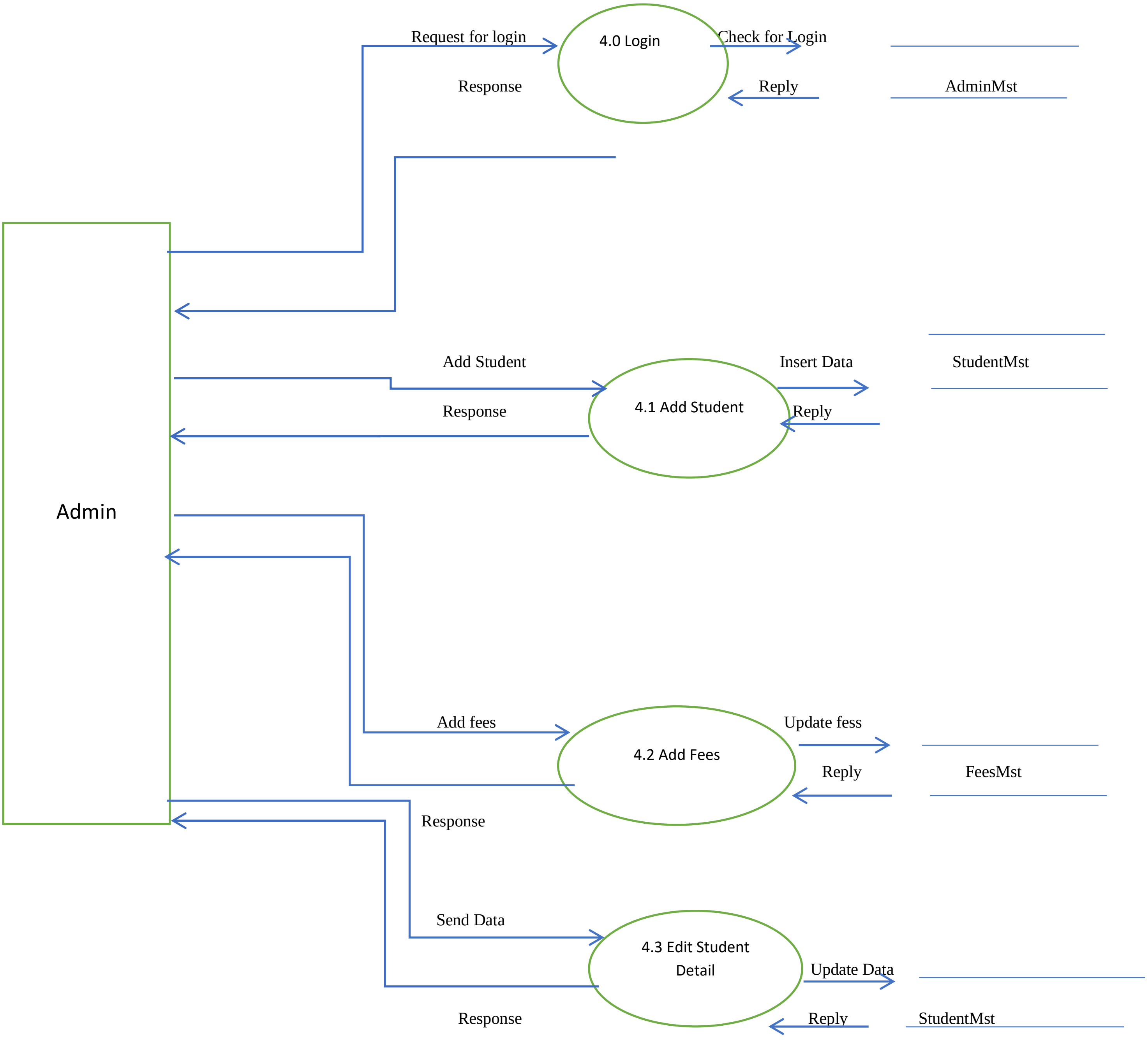


Fig:-2-Level Admin Side DFD(4.0)

Chapter 6

Methodology

Prototyping Model has been used to develop this application. The Prototyping model is a technique for quickly building a function but incomplete model of the information system. There are several kinds of prototypes but they all intend to reduce risk by building a quick and dirty replica or mockup of the intended system. It can be used to demonstrate technical feasibility when the technical risk is high. It can also be used to better understand and elicit user requirements. In either case, the goal is to reduce risk and limit costs by increasing understanding of proposed solutions before committing more resources, as shown in figure 1.

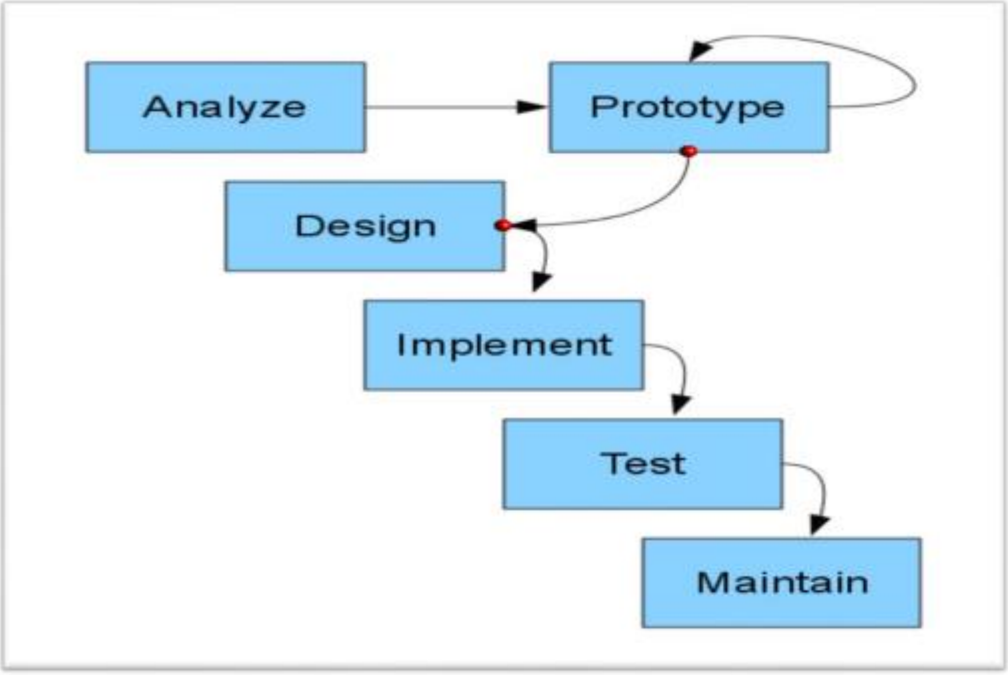
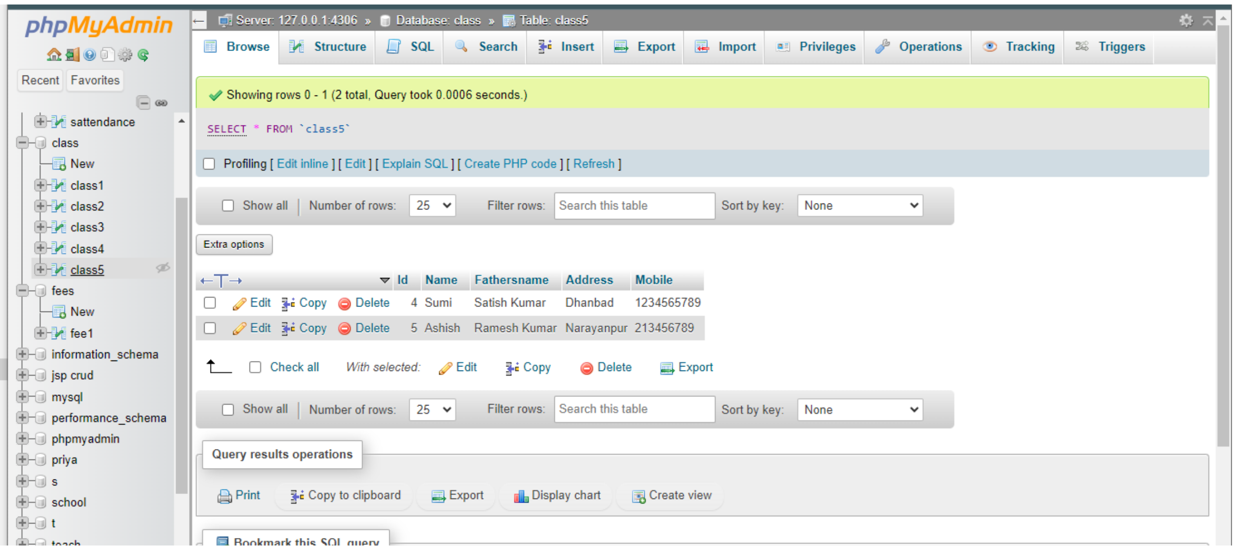


Fig 1.1:- Prototyping Model Process

Chapter 7

Snapshot of the Running Model



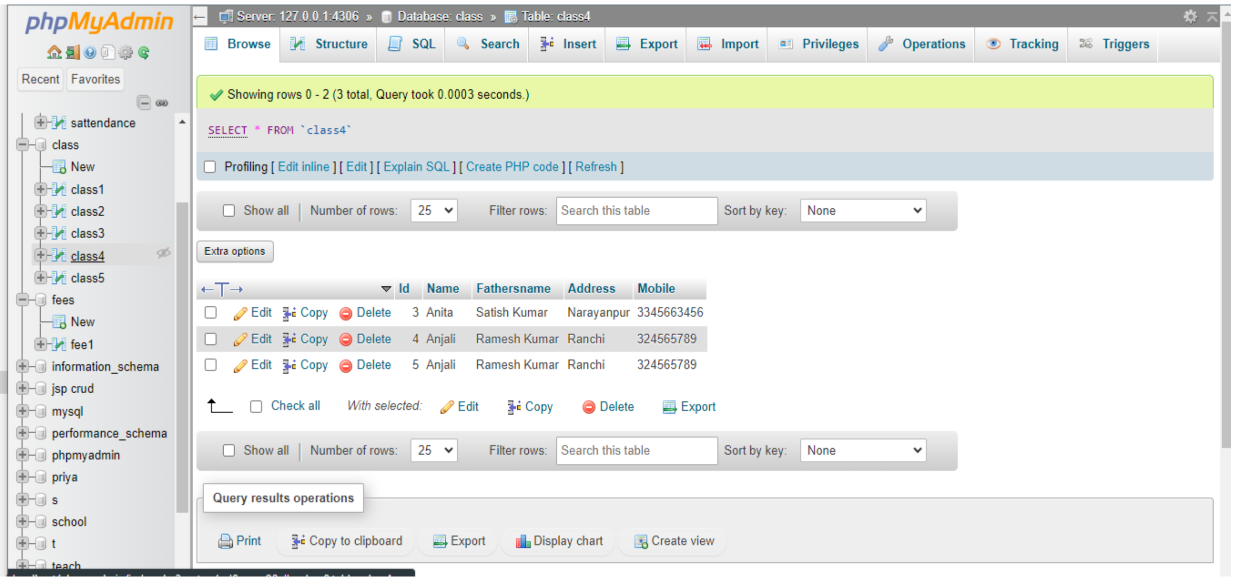
Server: 127.0.0.1:4306 » Database: class » Table: class5

✓ Showing rows 0 - 1 (2 total, Query took 0.0006 seconds.)

```
SELECT * FROM `class5`
```

Id	Name	Fathersname	Address	Mobile
4	Sumi	Satish Kumar	Dhanbad	1234565789
5	Ashish	Ramesh Kumar	Narayanpur	213456789

Table 1:-Students List Of Class 5



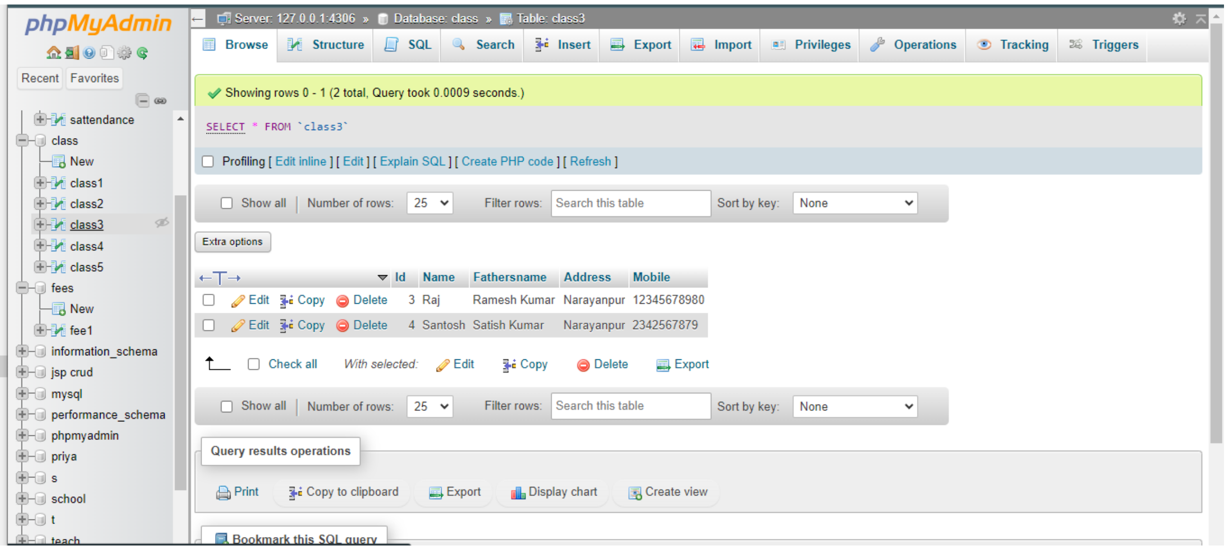
Server: 127.0.0.1:4306 » Database: class » Table: class4

✓ Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)

```
SELECT * FROM `class4`
```

Id	Name	Fathersname	Address	Mobile
3	Anita	Satish Kumar	Narayanpur	3345663456
4	Anjali	Ramesh Kumar	Ranchi	324565789
5	Anjali	Ramesh Kumar	Ranchi	324565789

Table 2:-Students List Of Class 4



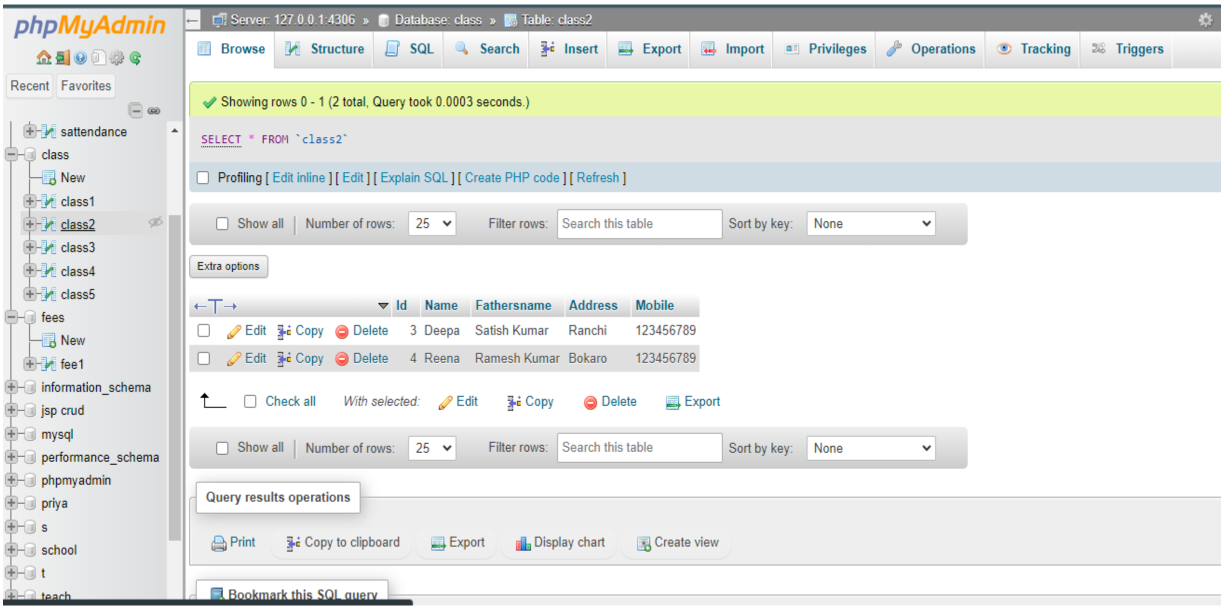
✓ Showing rows 0 - 1 (2 total, Query took 0.0009 seconds.)

```
SELECT * FROM `class3`
```

Number of rows: 25 | Sort by key: None

Id	Name	Fathersname	Address	Mobile
3	Raj	Ramesh Kumar	Narayanpur	12345678980
4	Santosh	Satish Kumar	Narayanpur	2342567879

Table 3:-Students List Of Class 3



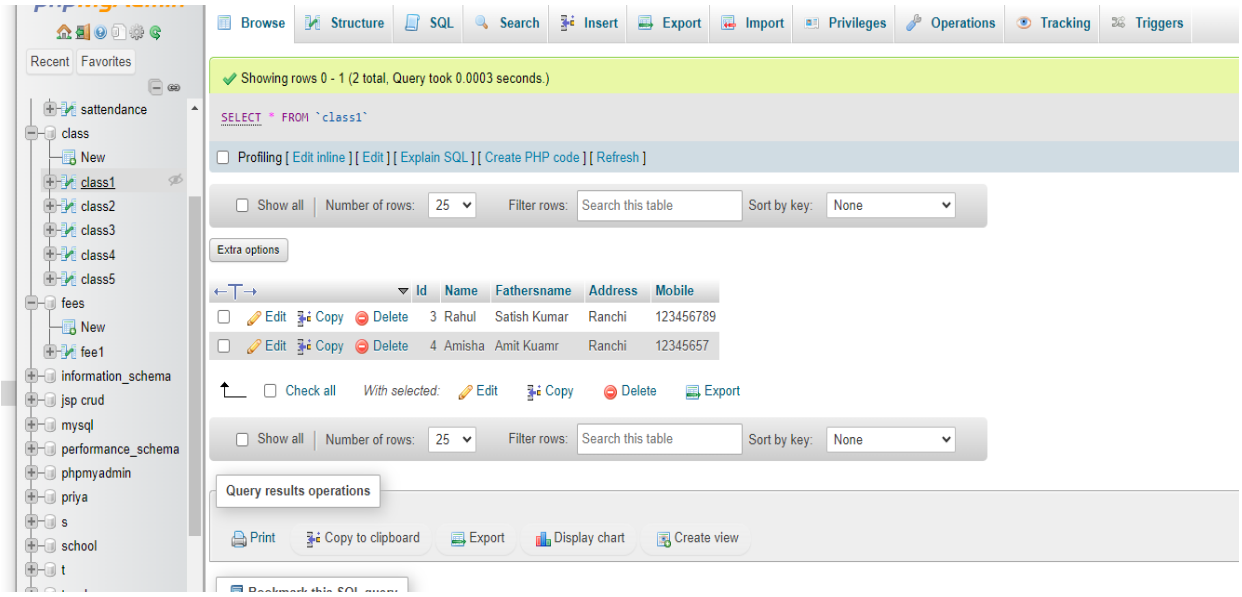
✓ Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)

```
SELECT * FROM `class2`
```

Number of rows: 25 | Sort by key: None

Id	Name	Fathersname	Address	Mobile
3	Deepa	Satish Kumar	Ranchi	123456789
4	Reena	Ramesh Kumar	Bokaro	123456789

Table 4:-Students List Of Class 2



✓ Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)

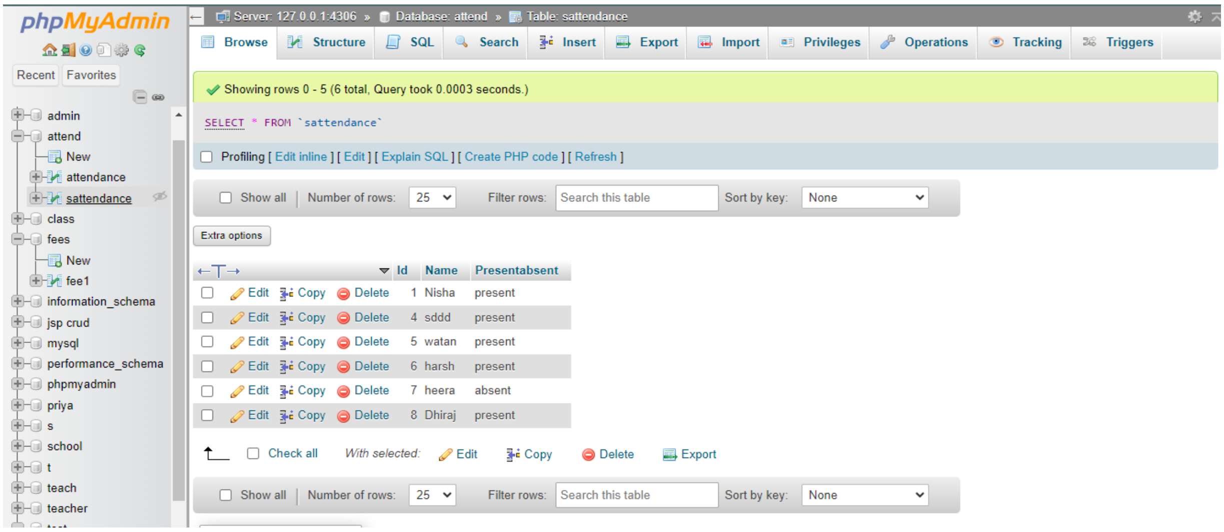
```
SELECT * FROM `class1`
```

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

	Id	Name	Fathersname	Address	Mobile
□ Edit Copy Delete	3	Rahul	Satish Kumar	Ranchi	123456789
□ Edit Copy Delete	4	Amisha	Amit Kuamr	Ranchi	12345657

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Table 5:-Students List Of Class 1



Server: 127.0.0.1:4306 » Database: attend » Table: sattendance

✓ Showing rows 0 - 5 (6 total, Query took 0.0003 seconds.)

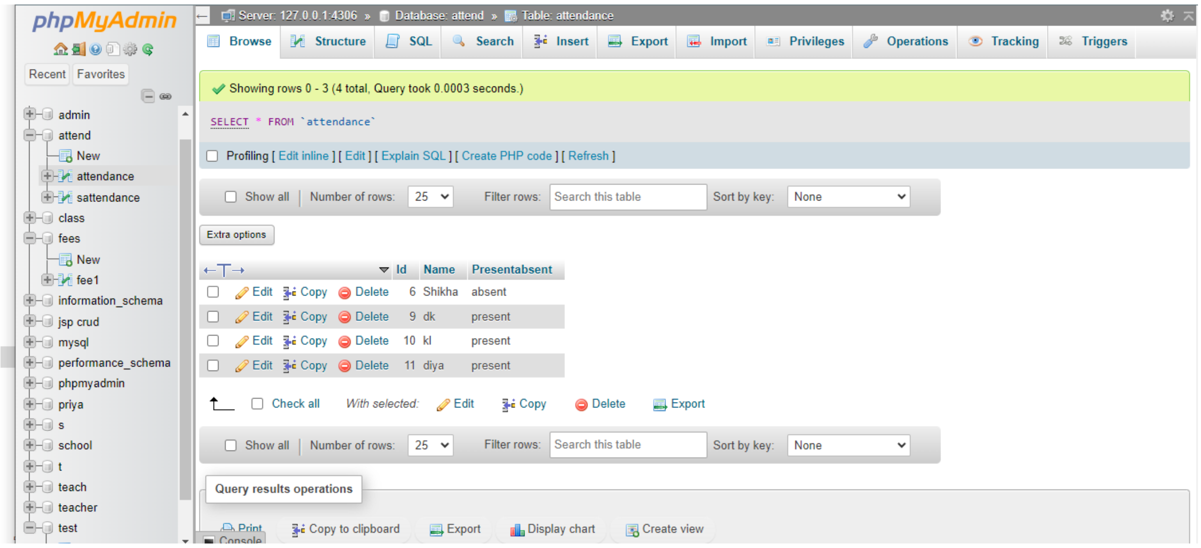
```
SELECT * FROM `sattendance`
```

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

	Id	Name	Presentabsent
□ Edit Copy Delete	1	Nisha	present
□ Edit Copy Delete	4	sddd	present
□ Edit Copy Delete	5	watan	present
□ Edit Copy Delete	6	harsh	present
□ Edit Copy Delete	7	heera	absent
□ Edit Copy Delete	8	Dhiraj	present

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Table 6 :-Student Attendance List


phpMyAdmin
Recent Favorites
admin
attend
New
attendance
sattendance
class
fees
New
fee1
information_schema
jsp crud
mysql
performance_schema
phpmyadmin
priya
s
school
t
teach
teacher
test
Server: 127.0.0.1:4306 » Database: attend » Table: attendance
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Extra options
Check all With selected: Edit Copy Delete Export
Query results operations
Print Copy to clipboard Export Display chart Create view
Console

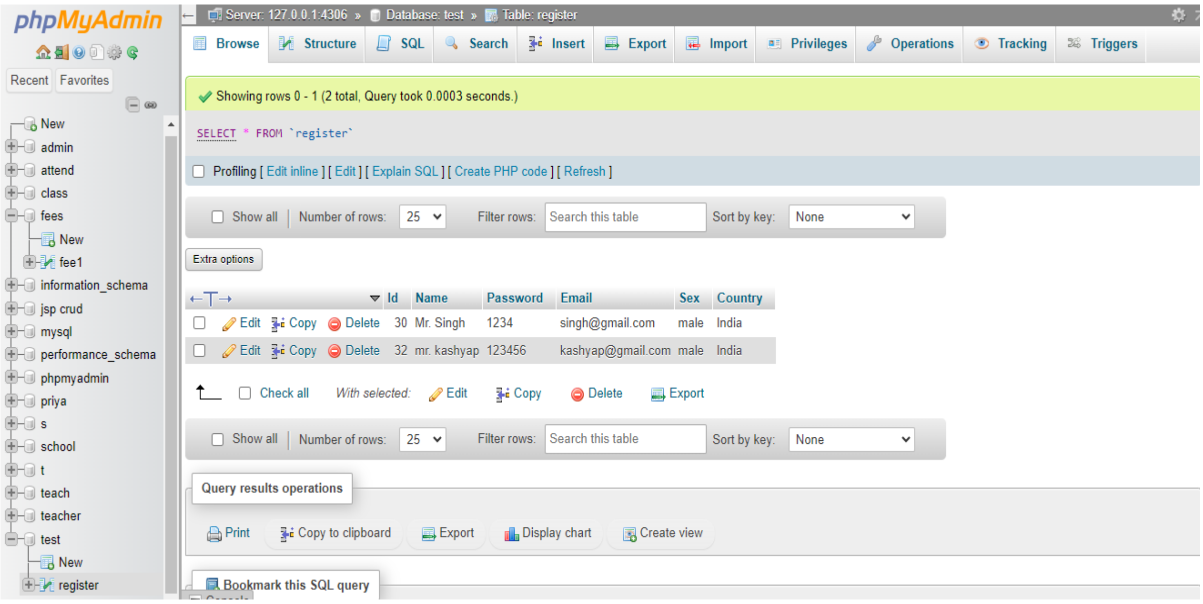
✓ Showing rows 0 - 3 (4 total, Query took 0.0003 seconds.)

```
SELECT * FROM `attendance`
```

□ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

				Id	Name	Presentabsent
□	Edit	Copy	Delete	6	Shikha	absent
□	Edit	Copy	Delete	9	dk	present
□	Edit	Copy	Delete	10	kl	present
□	Edit	Copy	Delete	11	diya	present

Table 7 :-Teacher Attendance List


phpMyAdmin
Recent Favorites
New
admin
attend
class
fees
New
fee1
information_schema
jsp crud
mysql
performance_schema
phpmyadmin
priya
s
school
t
teach
teacher
test
New
register
Server: 127.0.0.1:4306 » Database: test » Table: register
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Extra options
Check all With selected: Edit Copy Delete Export
Query results operations
Print Copy to clipboard Export Display chart Create view
Bookmark this SQL query
Console

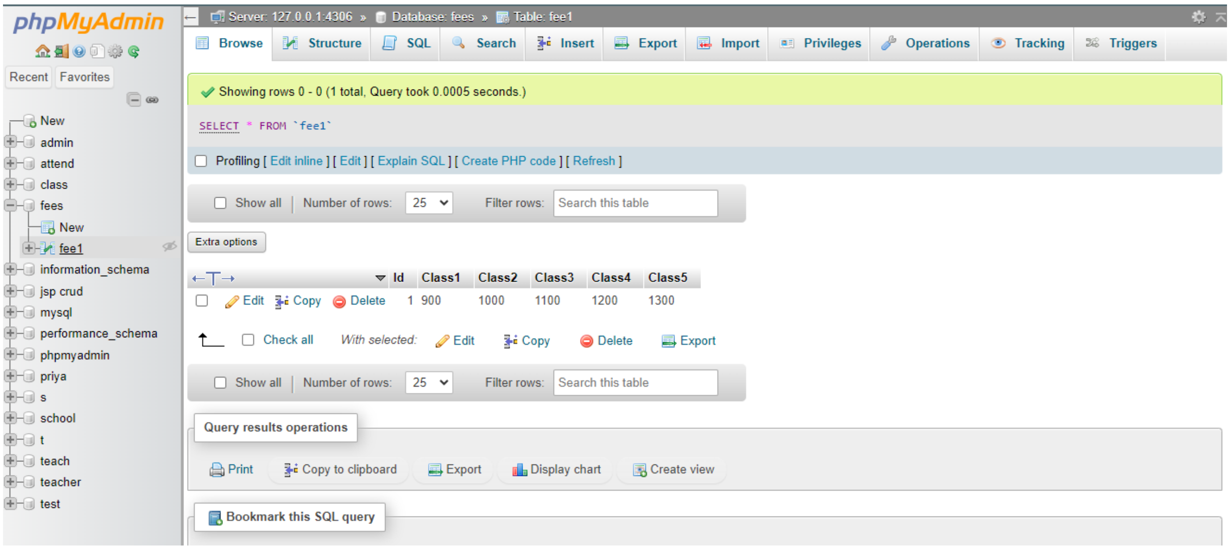
✓ Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)

```
SELECT * FROM `register`
```

□ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

				Id	Name	Password	Email	Sex	Country
□	Edit	Copy	Delete	30	Mr. Singh	1234	singh@gmail.com	male	India
□	Edit	Copy	Delete	32	mr. kashyap	123456	kashyap@gmail.com	male	India

Table 8 :-All Teachers Name List

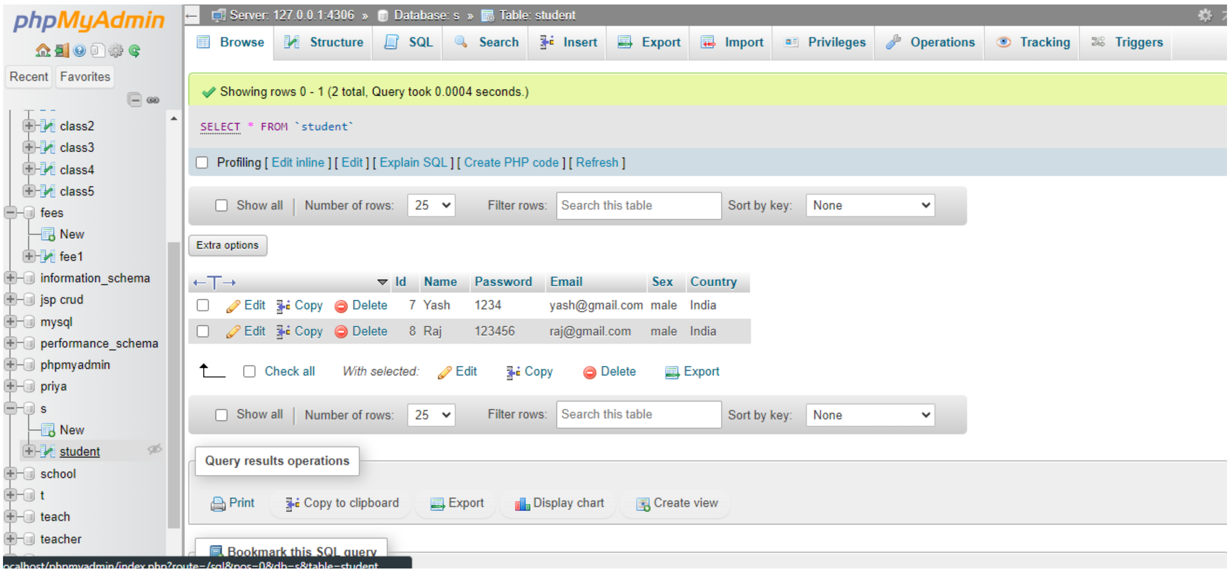


✓ Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)

```
SELECT * FROM `fee1`
```

	Id	Class1	Class2	Class3	Class4	Class5
□ Edit Copy Delete	1	900	1000	1100	1200	1300

Table 9 :-Student Fees List

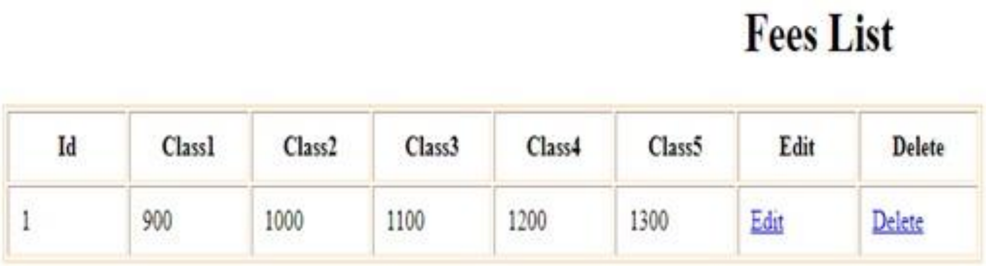


✓ Showing rows 0 - 1 (2 total, Query took 0.0004 seconds.)

```
SELECT * FROM `student`
```

	Id	Name	Password	Email	Sex	Country
□ Edit Copy Delete	7	Yash	1234	yash@gmail.com	male	India
□ Edit Copy Delete	8	Raj	123456	raj@gmail.com	male	India

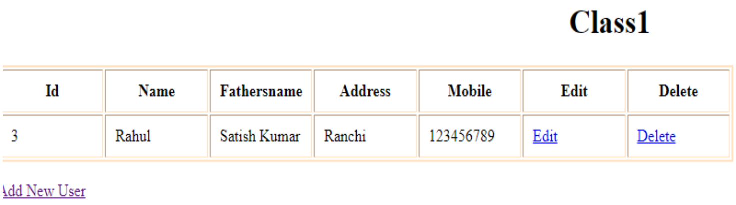
Table 10 :- All Students Name List



Fees List

Id	Class1	Class2	Class3	Class4	Class5	Edit	Delete
1	900	1000	1100	1200	1300	Edit	Delete

Figure 3.1:- Student Fees List

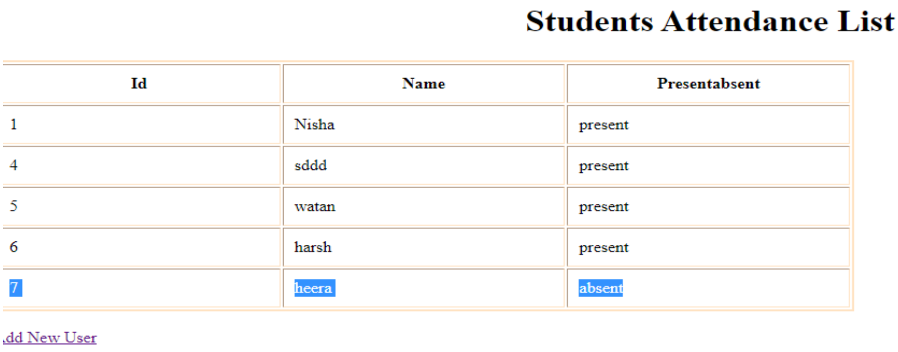


Class1

Id	Name	Fathersname	Address	Mobile	Edit	Delete
3	Rahul	Satish Kumar	Ranchi	123456789	Edit	Delete

Add New User

Figure 3.2:- Class 1 Students List

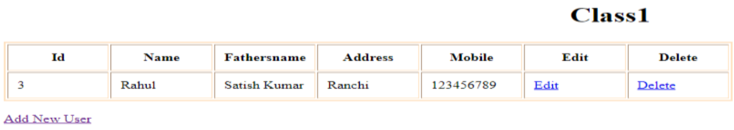


Students Attendance List

Id	Name	Presentabsent
1	Nisha	present
4	sddd	present
5	watan	present
6	harsh	present
7	heera	absent

.dd New User

Figure 3.3:- Student Attendance List

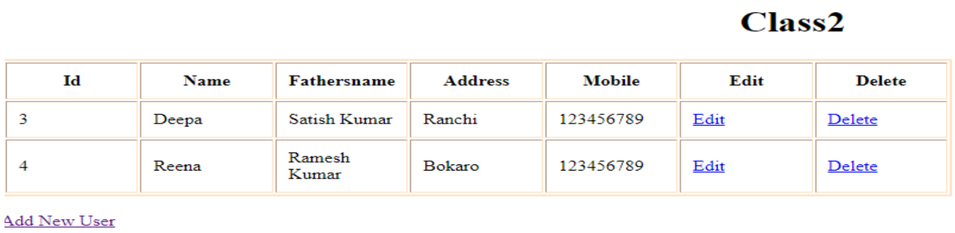


Class1

Id	Name	Fathersname	Address	Mobile	Edit	Delete
3	Rahul	Satish Kumar	Ranchi	123456789	Edit	Delete

Add New User

Figure 3.4:- Class 1 Students List

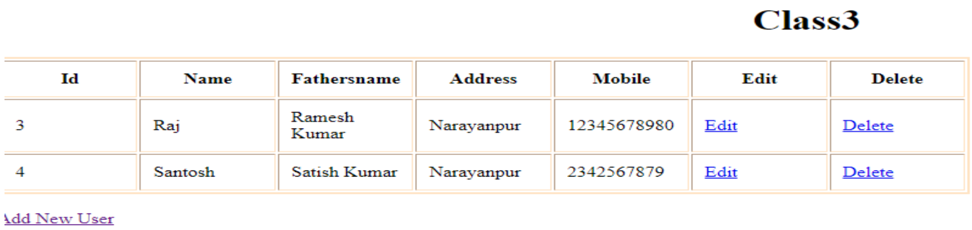


Class2

Id	Name	Fathersname	Address	Mobile	Edit	Delete
3	Deepa	Satish Kumar	Ranchi	123456789	Edit	Delete
4	Reena	Ramesh Kumar	Bokaro	123456789	Edit	Delete

Add New User

Figure 3.5:- Class 2 Students List

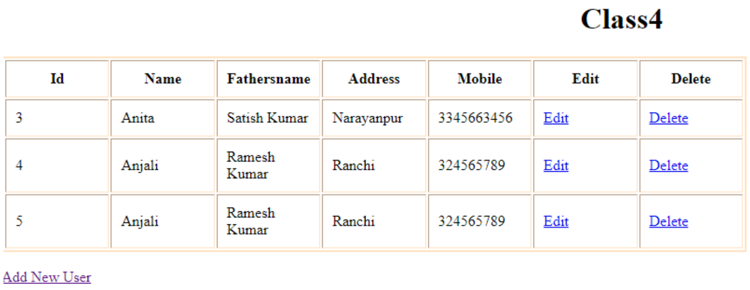


Class3

Id	Name	Fathersname	Address	Mobile	Edit	Delete
3	Raj	Ramesh Kumar	Narayanpur	12345678980	Edit	Delete
4	Santosh	Satish Kumar	Narayanpur	2342567879	Edit	Delete

Add New User

Figure 3.6:- Class 3 Student List

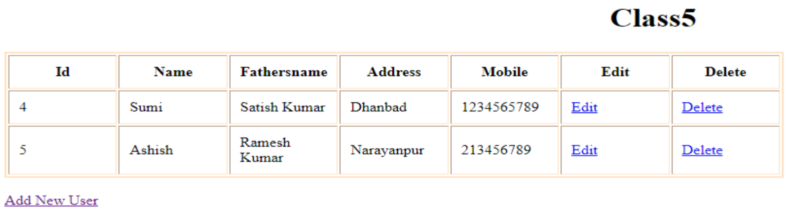


Class4

Id	Name	Fathersname	Address	Mobile	Edit	Delete
3	Anita	Satish Kumar	Narayanpur	3345663456	Edit	Delete
4	Anjali	Ramesh Kumar	Ranchi	324565789	Edit	Delete
5	Anjali	Ramesh Kumar	Ranchi	324565789	Edit	Delete

Add New User

Figure 3.7:- Class 4 Students List

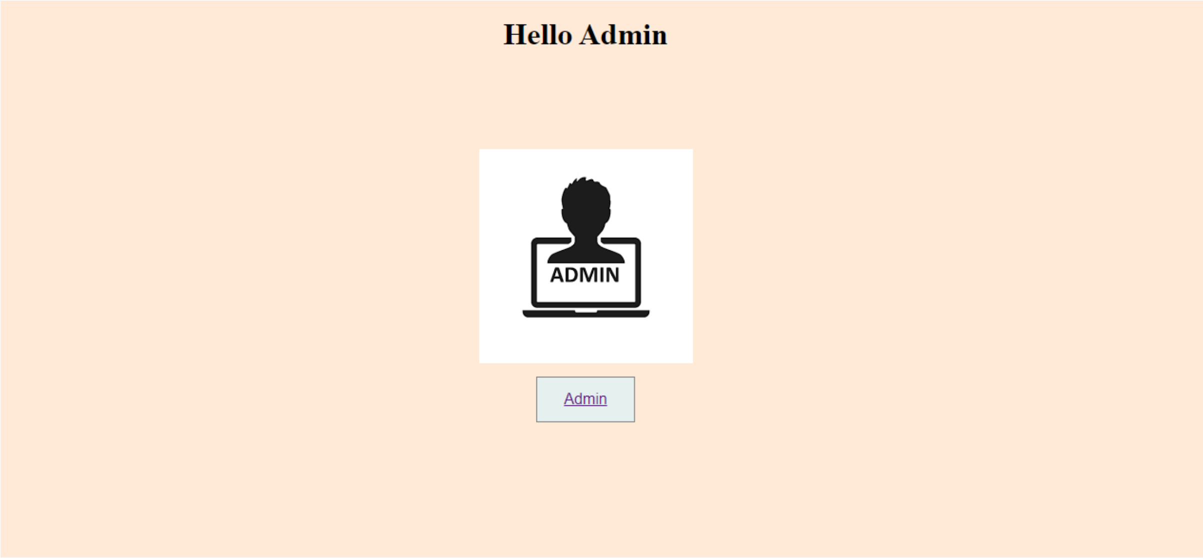


Class5

Id	Name	Fathersname	Address	Mobile	Edit	Delete
4	Sumi	Satish Kumar	Dhanbad	1234565789	Edit	Delete
5	Ashish	Ramesh Kumar	Narayanpur	213456789	Edit	Delete

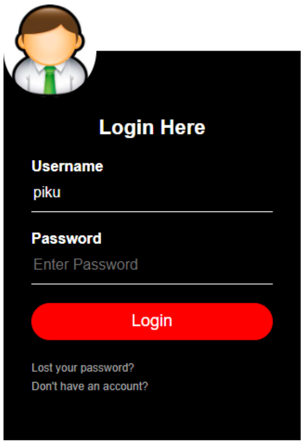
Add New User

Figure 3.8:- Class 5 Students List



Hello Admin

Figure 3.9:- Output Screen 1



Login Here

Username
piku

Password
Enter Password

Figure 3.10:- Output Screen 2

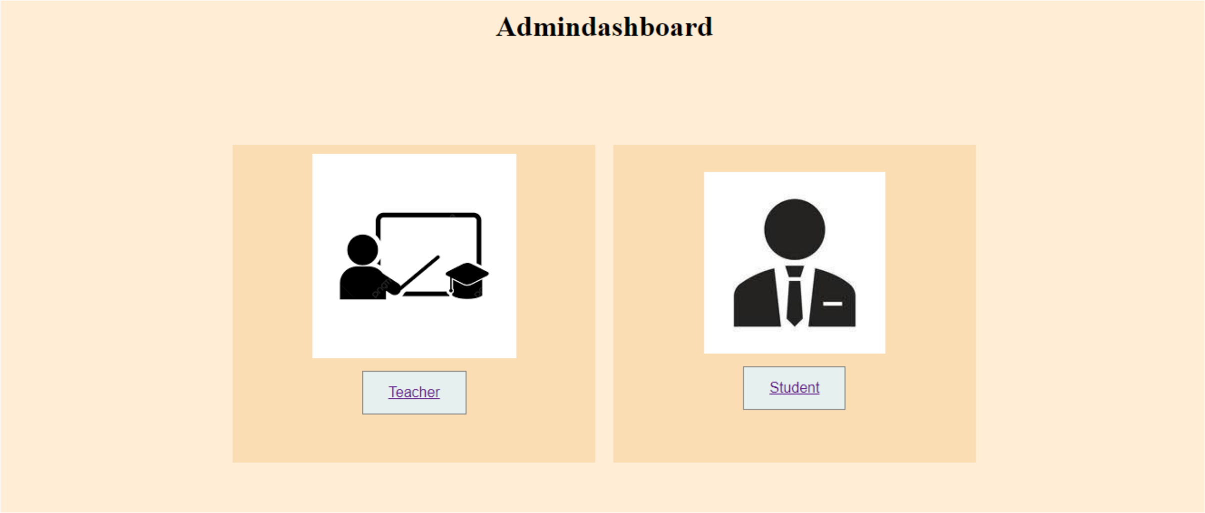


Figure 3.11:-Admin Dashboard

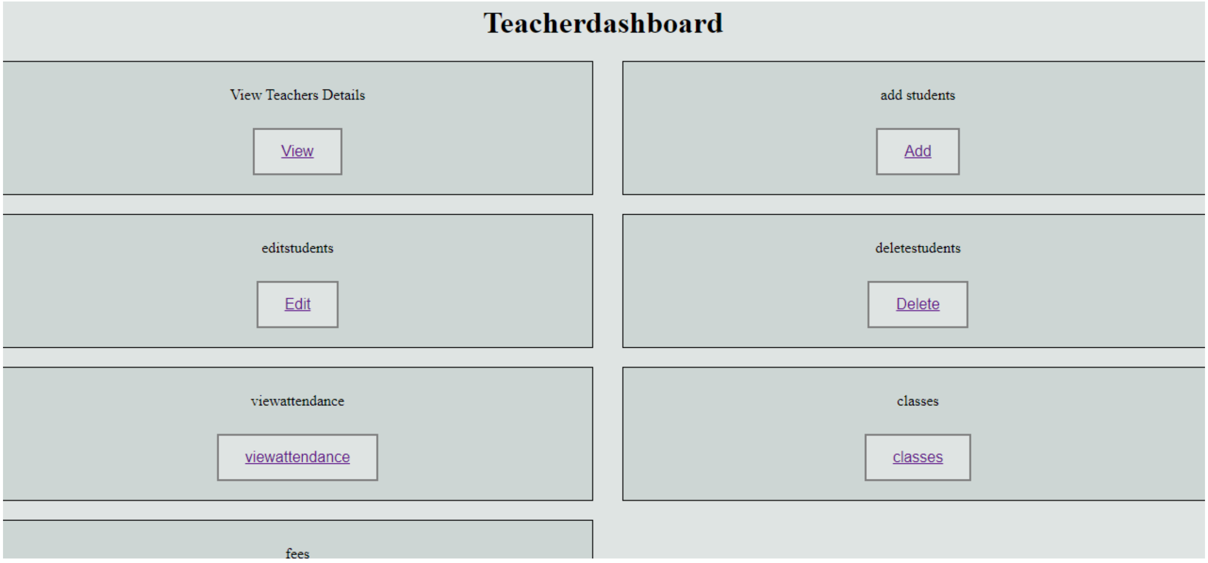
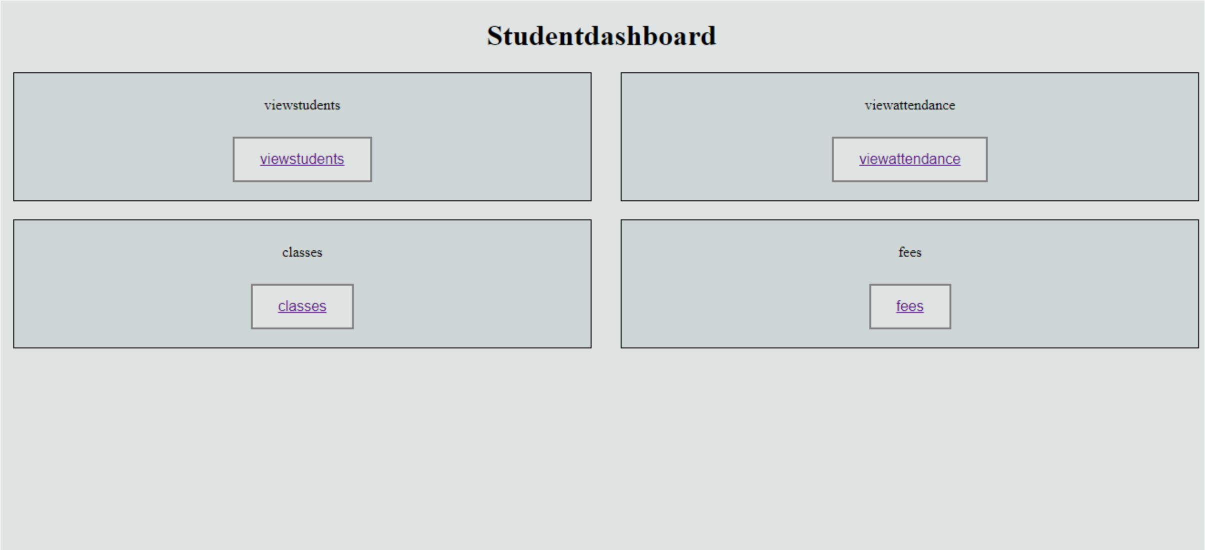
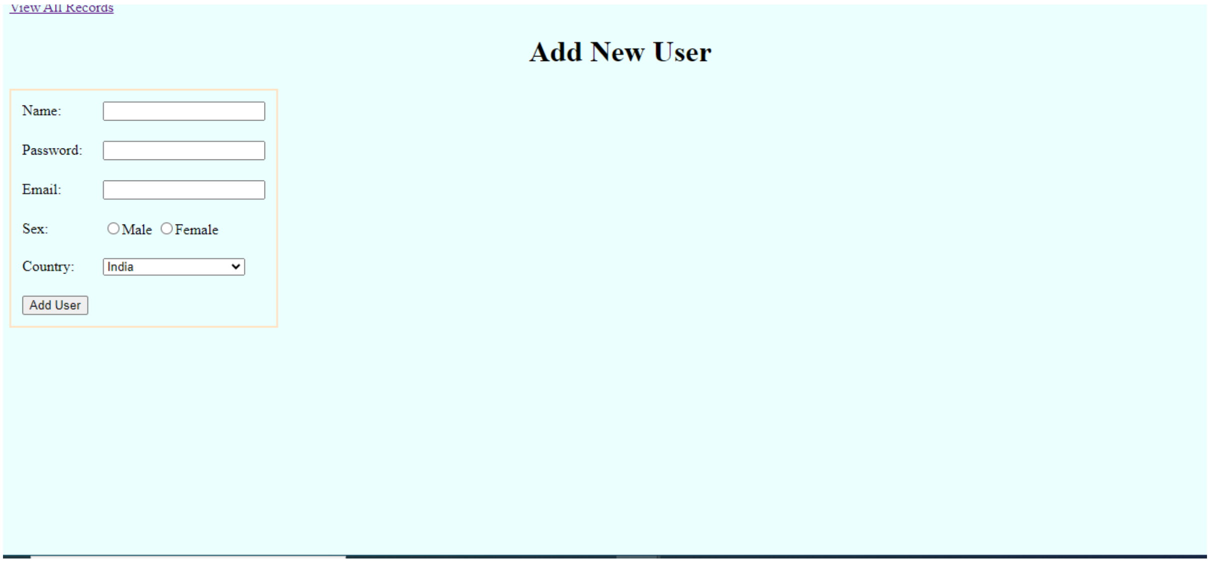


Figure 3.12:-Teachers board



Studentdashboard

Figure 3.13:-Student dash board



Add New User

Name:

Password:

Email:

Sex: ○Male ○Female

Country: India

Figure 3.14:-Add new user

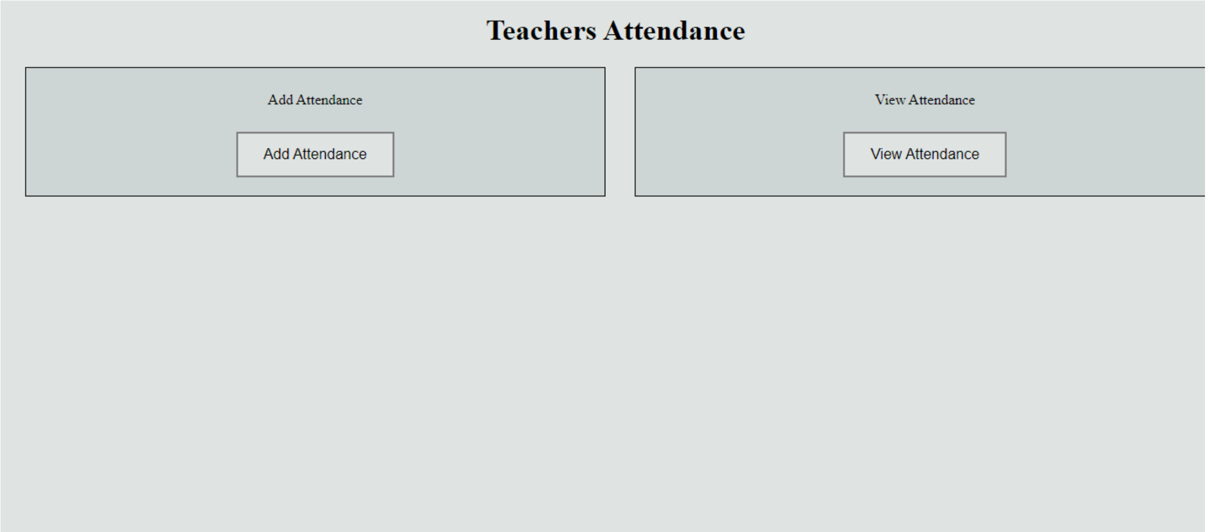


Figure 3.15:-Teacher attendance

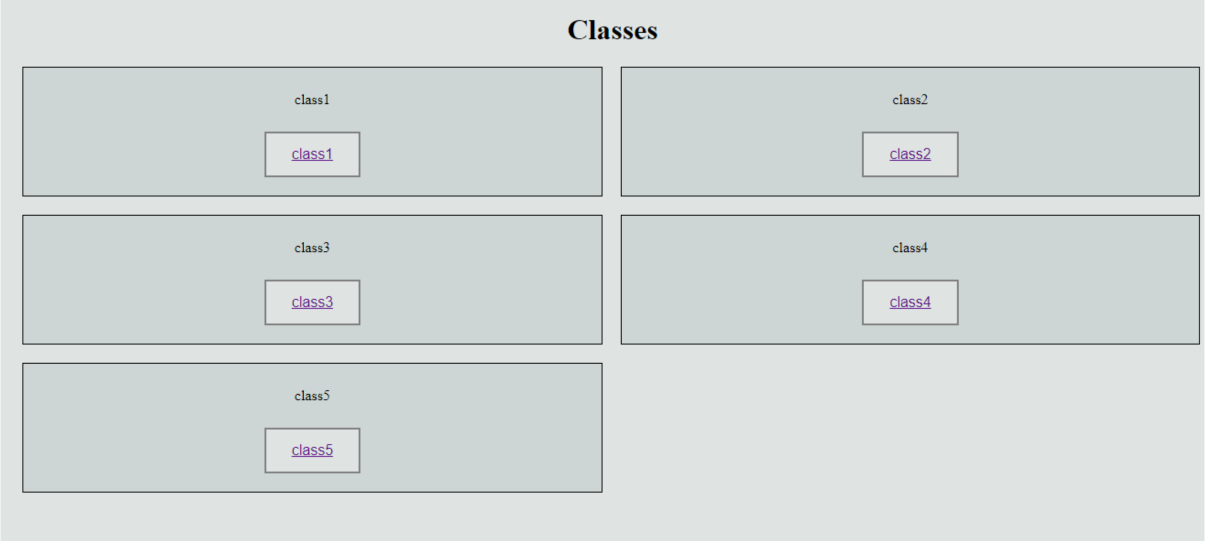


Figure 3.16:-Classes

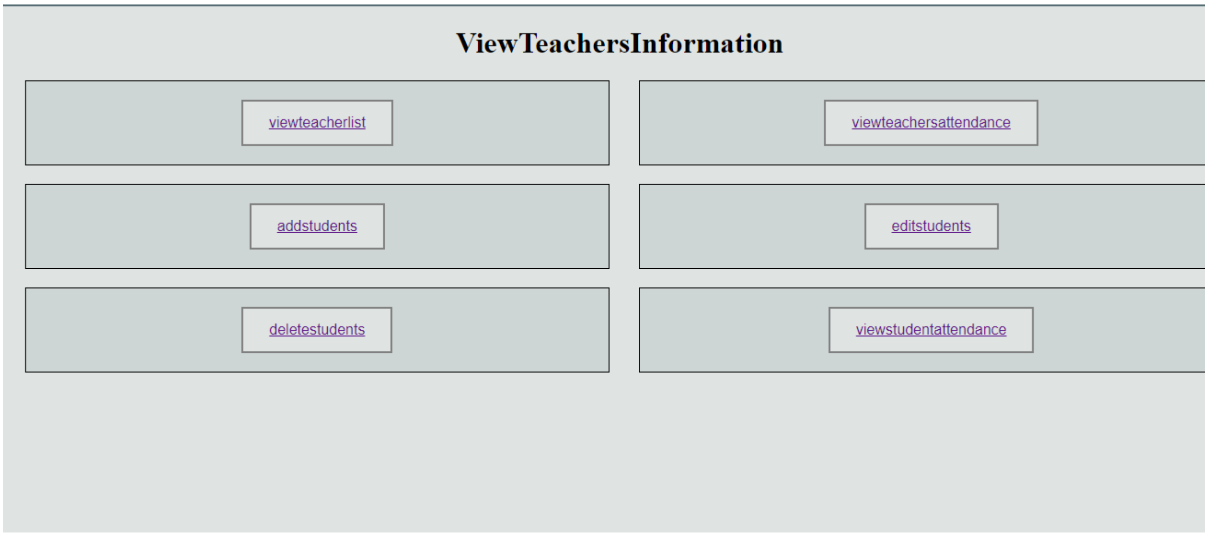


Figure 17:-Teacher information

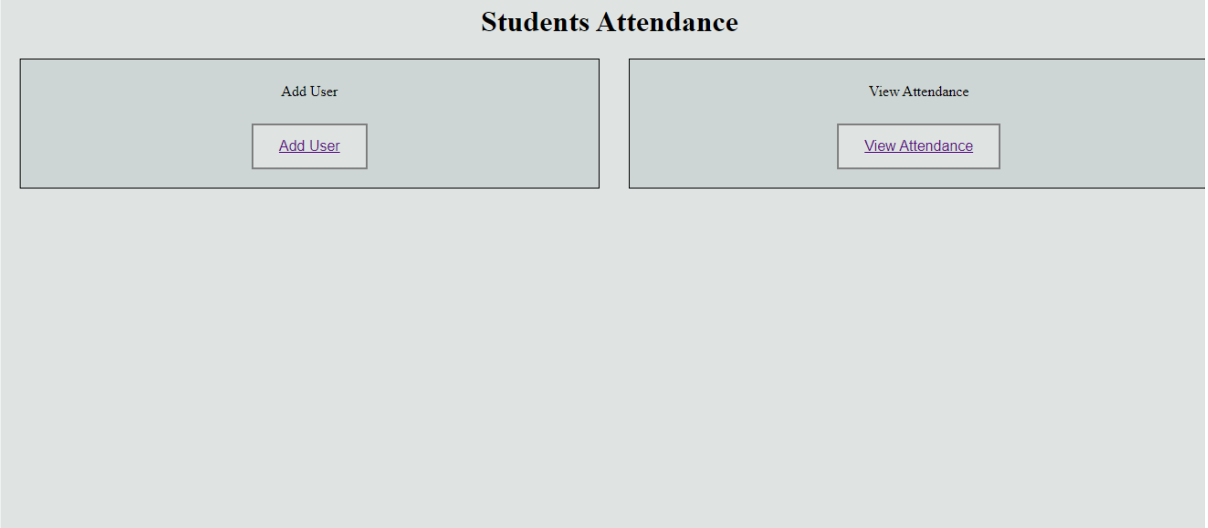
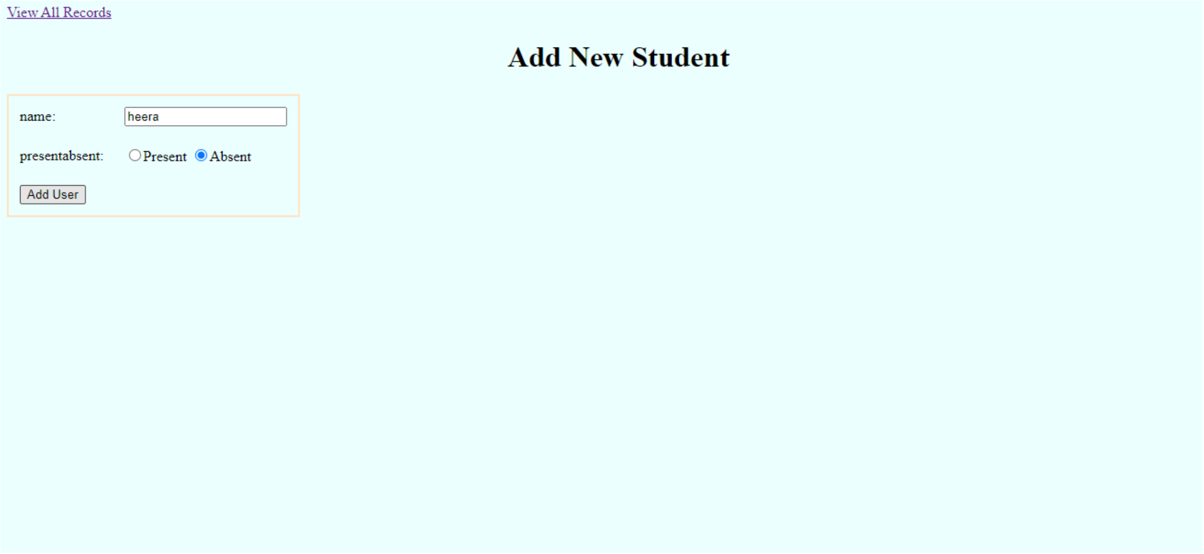


Figure 18:-Students Attendance



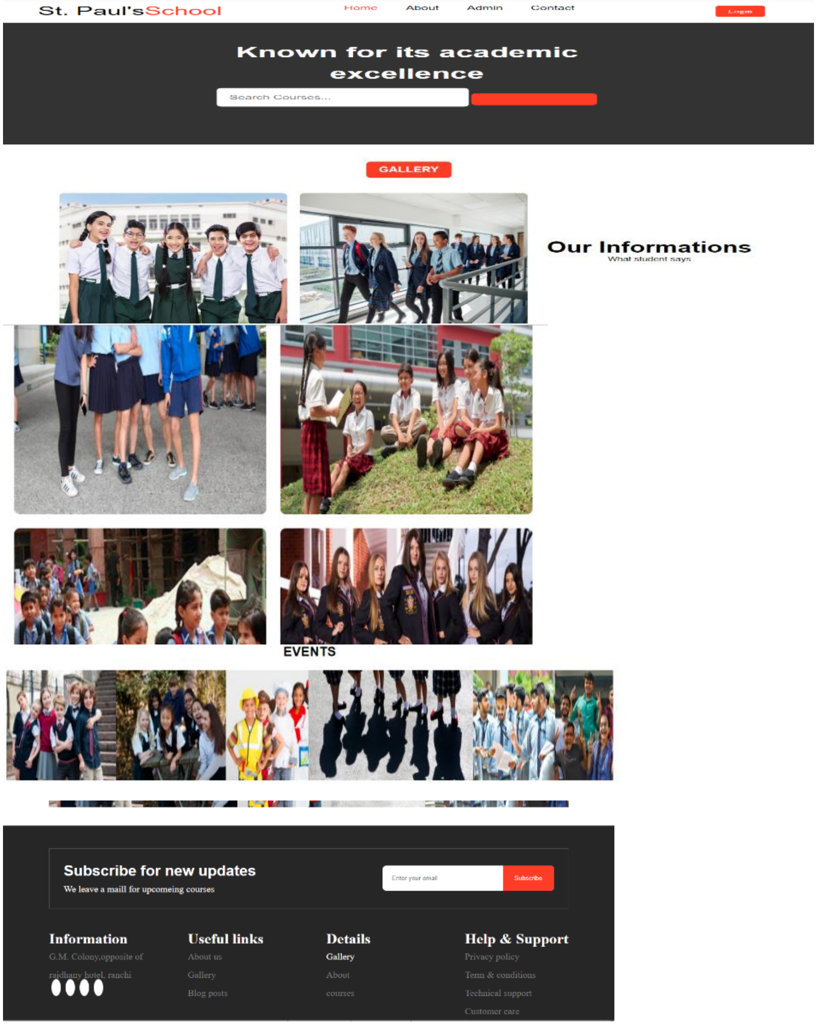
View All Records

Add New Student

name: heera

presentabsent: ○ Present ● Absent

Figure 19:-Add new Student



Known for its academic excellence

Our Informations

What student says

Subscribe for new updates

We leave a maill for upcomeing courses

Information

G.M. Colony,opposite of rajdhany hotel, ranchi

Useful links

Details

Help & Support

Figure 20:- Home page

Chapter 8

Source Code

```
//adduser

<%@page import="com.javatpoint.dao.UserDao"%>

<jsp:useBean id="u" class="com.javatpoint.bean.User"></jsp:useBean>

<jsp:setProperty property="*" name="u"/>



<%

int i=UserDao.save(u);

if(i>0){

response.sendRedirect("adduser-success.jsp");

}else{

response.sendRedirect("adduser-error.jsp");

}

%>

//addusererror

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Add User Error</title>
```

```
</head>

<body>



<p>Sorry, an error occurred!</p>

<jsp:include page="userform.html"></jsp:include>



</body>

</html>

//adduserform

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Add User Error</title>

</head>

<body>



<p>Sorry, an error occurred!</p>

<jsp:include page="userform9.html"></jsp:include>



</body>

</html>

//adduser-success
```

```
<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Add User Success</title>

</head>

<body>



<p>Record successfully saved!</p>

<jsp:include page="userform.html"></jsp:include>



</body>

</html>

//admin

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Insert title here</title>

</head>

<style>

    body{

        background-color: antiquewhite;
    }

    h1{

        font-family: 'Times New Roman';

        text-align: center;

    }

    .container {

        display: flex;

         justify-content: center;

         align-items: center;

         padding-top: 80px;


        }

       .middle {

        text-align: center;

        }

        button{

        border: 1px solid gray;

        background-color: rgb(232, 240, 239);

        padding: 14px 28px;

         font-size: 16px;

         cursor: pointer;

         margin: 10px;

    }
```

```
</style>

<body>

<h1>Hello Admin</h1>

<div class="container">

<div class="middle">

<a  href="loginadmin.html">

<imgsrc=".\1.png"><br></a>

<button class="button button"><a
href="loginadmin.html"><span>Admin</span></a></button>

</div>

</div>

</body>

</html>

//admin dashboard

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Insert title here</title>

</head>

<style>

   body{

      background-color: papayawhip;
```

```
    }
    h1{
        font-family: 'Times New Roman';
        text-align: center;
    }
    .container {
        display: flex;
         justify-content: center;
         align-items: center;
         padding-top: 80px;

        }
        .leftpart{
    width: 30%;
   height: available;
   text-align: center;
   height: 350px;
   background-color: wheat;
   border-collapse: collapse;
margin: 10px;
}
.left {
    padding: 10px;
    text-align: center;
}
        .rightpart{
    width: 30%;
   height: available;
   text-align: center;
   height: 350px;
   background-color: wheat;
   border-collapse: collapse;
   margin: 10px;
}
   .right {
    padding: 30px;
    text-align: center;
}
button{
        border: 1px solid gray;
        background-color: rgb(232, 240, 239);
        padding: 14px 28px;
         font-size: 16px;
         cursor: pointer;
         margin: 10px;
    }
```

```
</style>

<body>

<h1>Admindashboard</h1>

<div class="container">

<div class="leftpart">

<div class="left">

<a  href="Teacherdashboard.html">

<imgsrc=".\2.png"><br></a>

<button                                class="button                              button"><a
href="Teacherdashboard.html"><span>Teacher</span></a></button>

</div>

</div>

<div class="rightpart">

<div class="right">

<a href="Studentdashboard.html">

<imgsrc=".\2.jpg"><br></a>

<button class="button button"><a href="Studentdashboard.html">Student</a></button>

</div>

</div>




</div>
```

```
</body>
</html>
//edit user
<%@page import="com.javatpoint.dao.UserDao"%>
<jsp:useBean id="u" class="com.javatpoint.bean.User"></jsp:useBean>
<jsp:setProperty property="*" name="u"/>
<%
int i=UserDao.update(u);
response.sendRedirect("viewusers.jsp");
%>
//edit form 3
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Edit Form</title>
</head>
<body>
<%@page import="com.javatpoint3.dao.UserDao3,com.javatpoint3.bean.User3"%>

<%
String id=request.getParameter("id");
User3 u=UserDao3.getRecordById(Integer.parseInt(id));
```

```
%>


<h1>Edit Form</h1>

<form action="edituser3.jsp" method="post">

<input type="hidden" name="id" value="<%=u.getId() %>"/>

<table>

<tr><td>Name:</td><td>

<input type="text" name="name" value="<%= u.getName()%>"/></td></tr>

<tr><td>Presentabsent:</td><td>


<input type="radio" name="presentabsent" value="present"/>Present

<input type="radio" name="presentabsent" value="absent"/>Absent </td></tr>




<tr><td colspan="2"><input type="submit" value="Edit User"/></td></tr>

</table>

</form>


</body>

</html>

//fee.dao

package com.fees.dao;
```

```
import java.sql.*;

import java.util.ArrayList;

import java.util.List;

import com.fees.bean.User;

public class UserDao {


public static Connection getConnection(){

   Connection con=null;

   try{

Class.forName("com.mysql.jdbc.Driver");

      con=DriverManager.getConnection("jdbc:mysql://localhost:4306/fees","root","");

   }catch(Exception e){System.out.println(e);}

   return con;

}

public static int save(User u){

   int status=0;

   try{

      Connection con=getConnection();

PreparedStatementps=con.prepareStatement(

"insert into fee1(class1,class2,class3,class4,class5) values(?,?,?,?,?)");

ps.setInt(1,u.getClass1());

ps.setInt(2,u.getClass2());

ps.setInt(3,u.getClass3());
ps.setInt(4,u.getClass4());
ps.setInt(5,u.getClass5());
        status=ps.executeUpdate();
    }catch(Exception e){System.out.println(e);}
    return status;
}
public static int update(User u){
    int status=0;
    try{
        Connection con=getConnection();
PreparedStatementps=con.prepareStatement(
"update fee1 set class1=?,class2=?,class3=?,class4=?,class5=? where id=?");
ps.setInt(1,u.getClass1());
ps.setInt(2,u.getClass2());
ps.setInt(3,u.getClass3());
ps.setInt(4,u.getClass4());
ps.setInt(5,u.getClass5());
ps.setInt(6,u.getId());
        status=ps.executeUpdate();
    }catch(Exception e){System.out.println(e);}
    return status;
}
public static int delete(User u){
    int status=0;

    try{

        Connection con=getConnection();

PreparedStatementps=con.prepareStatement("delete from fee1 where id=?");

ps.setInt(1,u.getId());

        status=ps.executeUpdate();

    }catch(Exception e){System.out.println(e);}



    return status;

}

public static List<User>getAllRecords(){

    List<User> list=new ArrayList<User>();



    try{

        Connection con=getConnection();

PreparedStatementps=con.prepareStatement("select * from fee1");

ResultSetrs=ps.executeQuery();

        while(rs.next()){

            User u=new User();

u.setId(rs.getInt("id"));

            u.setClass1(rs.getInt("class1"));

            u.setClass2(rs.getInt("class2"));

            u.setClass3(rs.getInt("class3"));
            u.setClass4(rs.getInt("class4"));
            u.setClass5(rs.getInt("class5"));
list.add(u);
        }
    }catch(Exception e){System.out.println(e);}
    return list;
}
public static User getRecordById(int id){
    User u=null;
    try{
        Connection con=getConnection();
PreparedStatementps=con.prepareStatement("select * from fee1 where id=?");
ps.setInt(1,id);
ResultSetrs=ps.executeQuery();
        while(rs.next()){
            u=new User();
u.setId(rs.getInt("id"));
            u.setClass1(rs.getInt("class1"));
            u.setClass2(rs.getInt("class2"));
            u.setClass3(rs.getInt("class3"));
            u.setClass4(rs.getInt("class4"));
            u.setClass5(rs.getInt("class5"));
        }
```

```
   }catch(Exception e){System.out.println(e);}

   return u;

}

}

//front page

<!DOCTYPE html>

<html>

<head>

	<title>School Management System</title>

	<link rel="stylesheet" type="text/css" href="Front_Page.css">

	<link rel="stylesheet" type="text/css" href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css">

<link href="https://unpkg.com/aos@2.3.1/dist/aos.css" rel="stylesheet">

</head>

<body>

<header>

	<nav>

		<div class="logo">

			St. Paul's<span>School</span>

		</div>

		<div class="menu">

			<a href="Home.html">Home</a>

			<a href="Admin.html">Admin</a>
```

```
            <a href="Teacher.html">Teacher</a>
            <a href="Student.html">Student</a>
        </div>
        <div class="icon">
            <i class="fa fa-search"></i>
            <a href="login.html">Login</a>

        </div>
    </nav>


    <section class="h-text">
        <h1 data-aos="zoom-in-down" data-aos-delay="100">Known for its academic excellence</h1>
        <input type="text" name="" placeholder="Search Courses...">
        <button><i class="fa fa-search"></i></button>


    </section>
</header>

<section class="gallery">
    <div class="menu">
```

```
            <ul>
                <li class="active" data-filter="*"> GALLERY </li>
        </div>
        <div class="menu-item">
            <div class="item web"  data-aos="flip-left" data-aos-offset="100" data-aos-delay="100">
                <imgsrc=".\image1.png" width="100" height="100">
            </div>

            <div class="item seo"  data-aos="flip-left" data-aos-offset="100" data-aos-delay="200">
                <imgsrc=".\image2.png" width="100" height="100">
            </div>
            <div class="item graphics"  data-aos="flip-left" data-aos-offset="100" data-aos-delay="300">
                <imgsrc=".\image3.png" width="100" height="100">
            </div>
            <div class="item web" data-aos="zoom-in" data-aos-offset="100" data-aos-delay="100">
                <imgsrc=".\imagee4.png" width="100" height="100">
            </div>

            <div class="item seo" data-aos="zoom-in" data-aos-offset="100" data-aos-delay="200">
```

```
                    <imgsrc=".\image5.png" width="100" height="100">

          </div>

          <div class="item seo" data-aos="zoom-in" data-aos-offset="100" data-aos-delay="300">

                    <imgsrc=".\image6.png" width="100" height="100">

          </div>


     </div>


</section>




<!--- filter gallery end---->


<!-- slider start--->


<section class="heading">

     <div class="heading-info" data-aos="zoom-in-up" data-aos-offset="100" data-aos-delay="100">

          <h1>Our <span>Informations</span></h1>

   <p>What student says</p>

</div>
```

```
<div   class="member-image"     data-aos="zoom-in-down"   data-aos-offset="100"   data-aos-delay="300">



        <imgsrc=".\image13.jpeg"  width="100%" height="100%">

        <p>

        </p>

        <h2>EVENTS</h2>


        </div>

<div class="m-images">

        <imgsrc=".\image7.png">

        <imgsrc=".\image8.png">

        <imgsrc=".\image9.png">

        <imgsrc=".\image10.png">

        <imgsrc=".\image11.png">

        <imgsrc=".\images12.jpeg">



</div>

</section>
```

```
<footer>

        <div class="newsletter">

                <div data-aos="fade-up" data-aos-offset="200">

                        <h2>Subscribe for new updates</h2>

                        <p>We leave a maill for upcomeing courses</p>

                </div>

                <div class="n-text" data-aos="fade-up-right" data-aos-offset="200">

                        <label>

                                <input type="text" name="" placeholder="Enter your email">

                                <button>Subscribe</button>



                        </label>

                </div>

        </div>




        <div class="f-contact" data-aos="zoom-in-up" data-aos-offset="200">

                <div>

                        <h1>Information</h1>
                        <p>G.M. Colony,opposite of rajdhany hotel, ranchi</p>


                        <i class="fa fa-whatsapp"></i>

                        <i class="fa fa-instagram"></i>

                        <i class="fa fa-telegram"></i>

                        <i class="fa fa-twitter"></i>

                </div>




                <div>

                        <h1>Useful links</h1>

                        <p>About us</p>

                        <p>Gallery</p>

                        <p>Blog posts</p>




                </div>


                <div>

                        <h1>Details</h1>


                        <p>Gallery</p>

                        <p>About</p>
```

```
                <p>courses</p>


            </div>


            <div>
                <h1>Help & Support</h1>
                <p>Privacy policy</p>
                <p>Term & conditions</p>
                <p>Technical support</p>
                <p>Customer care</p>


            </div>
        </div>
</footer>




<script    src="https://code.jquery.com/jquery-3.6.0.min.js"    integrity="sha256-
/xUj+3OJU5yExlq6GSYGSHk7tPXikynS7ogEvDej/m4="
crossorigin="anonymous"></script>


<script src="https://unpkg.com/isotope-layout@3/dist/isotope.pkgd.min.js"></script>
<script>
        $('.portfolio-item').isotope({
```

```
  // options
itemSelector: '.item',
layoutMode: 'fitRows'
});
                    $('.portfolio-menu ul li').click(function(){
                    $('.portfolio-menu ul li').removeClass('active');
                    $(this).addClass('active');

                              var selector = $(this).attr('data-filter');
                    $('.portfolio-item').isotope({
                              filter:selector
                    });
                    return false;
          });
</script>

<script src="https://unpkg.com/aos@2.3.1/dist/aos.js"></script>
<script>
AOS.init();

</script>

</body>
```

```
</html>

//home page

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Insert title here</title>

</head>


<style>

    body{

        background-color: antiquewhite;

    }

    h1{

        font-family: 'Times New Roman';

        text-align: center;

    }

    .container {

        width: available;

    height: available;

    display: flex;

         justify-content: center;

         align-items: center;
         padding-top: 80px;


        }


.leftpart{
    width: 30%;
   height: available;
   text-align: center;
   height: 350px;
   background-color: wheat;
   border-collapse: collapse;
   margin: 5px;
}
.left {
    padding: 10px;
    text-align: center;
}
.middlepart{
  width: 30%;
   height: available;
   text-align: center;
   margin: 5px;
   height: 350px;
   background-color: lavender;

   border-collapse: collapse;

}

.middle {

    padding: 8px;

    text-align: center;

}


.rightpart{

    width: 30%;

   height: available;

   text-align: center;

   height: 350px;

   background-color: wheat;

   border-collapse: collapse;

   margin: 5px;

}

   .right {

    padding: 30px;

    text-align: center;

}

    button{

        border: 2px solid gray;
        background-color: rgb(209, 241, 236);

        padding: 14px 28px;

         font-size: 16px;

         cursor: pointer;

         margin: 10px;

    }


</style>

<body>

<h1>Homepage</h1>

<div class="container">

<div class="leftpart">

<div class="left">

<a  href="Admin.html">

<imgsrc=".\1.png"><br></a>

<button class="button button"><a href="Admin.html"><span>Admin</span></a></button>

</div>

</div>

<div class="middlepart">

<div class="middle">

<a href="Teacher.html">

<imgsrc=".\2.png"><br></a>

<button class="button button"><a href="Teacher.html">Teacher</a></button>
```

```
</div>

</div>

<div class="rightpart">

<div class="right">

<a href="Student.html">

<imgsrc=".\2.jpg"><br></a>

<button class="button button"><a href="Student.html">Student</a></button>

</div>

</div>

</div>

</body>

</html>

//index

<a href="viewusers.jsp">View All Records</a><br/>


<h1>Add New User</h1>

<form action="adduser.jsp" method="post">

<table>

<tr><td>Name:</td><td><input type="text" name="name"/></td></tr>

<tr><td>Password:</td><td>

<input type="password" name="password"/></td></tr>

<tr><td>Email:</td><td><input type="email" name="email"/></td></tr>

<tr><td>Sex:</td><td>
```

```
<input type="radio" name="sex" value="male"/>Male

<input type="radio" name="sex" value="female"/>Female </td></tr>

<tr><td>Country:</td><td>

<select name="country" style="width:155px">

<option>India</option>

<option>Pakistan</option>

<option>Afghanistan</option>

<option>Berma</option>

<option>Other</option>

</select>

</td></tr>

<tr><td colspan="2"><input type="submit" value="Add User"/></td></tr>

</table>

</form>

//java point3

package com.javatpoint3.bean;


public class User3 {

        private int id;

        private String name,presentabsent;

        public static void main(String[] args) {
	}
	public int getId() {
		return id;
	}
	public void setId(int id) {
		this.id = id;
	}
	public String getName() {
		return name;
	}
	public void setName(String name) {
		this.name = name;
	}
	public String getPresentabsent() {
		return presentabsent;
	}
	public void setPresentabsent(String presentabsent) {
		this.presentabsent = presentabsent;
	}

}
//java point 7
package com.javatpoint7.dao;
```

```
import java.sql.*;

import java.util.ArrayList;

import java.util.List;

import com.javatpoint7.bean.User7;

public class UserDao7 {


public static Connection getConnection(){

   Connection con=null;

   try{

Class.forName("com.mysql.jdbc.Driver");

      con=DriverManager.getConnection("jdbc:mysql://localhost:4306/class","root","");

   }catch(Exception e){System.out.println(e);}

   return con;

}

public static int save(User7 u){

   int status=0;

   try{

      Connection con=getConnection();

PreparedStatementps=con.prepareStatement(

"insert into class2(name,fathersname,address,mobile) values(?,?,?,?)");

ps.setString(1,u.getName());

ps.setString(2,u.getFathersname());

ps.setString(3,u.getAddress());
ps.setString(4,u.getMobile());


        status=ps.executeUpdate();
    }catch(Exception e){System.out.println(e);}
    return status;
}
public static int update(User7 u){
    int status=0;
    try{
        Connection con=getConnection();
PreparedStatementps=con.prepareStatement(
"update class2 set name=?,fathersname=?,address=?,mobile=? where id=?");
ps.setString(1,u.getName());
ps.setString(2,u.getFathersname());
ps.setString(3,u.getAddress());
ps.setString(4,u.getMobile());


ps.setInt(5,u.getId());
        status=ps.executeUpdate();
    }catch(Exception e){System.out.println(e);}
    return status;
}
public static int delete(User7 u){
    int status=0;
    try{
        Connection con=getConnection();
PreparedStatementps=con.prepareStatement("delete from class2 where id=?");
ps.setInt(1,u.getId());
        status=ps.executeUpdate();
    }catch(Exception e){System.out.println(e);}

    return status;
}
public static List<User7>getAllRecords(){
    List<User7> list=new ArrayList<User7>();

    try{
        Connection con=getConnection();
PreparedStatementps=con.prepareStatement("select * from class2");
ResultSetrs=ps.executeQuery();
        while(rs.next()){
            User7 u=new User7();
u.setId(rs.getInt("id"));
u.setName(rs.getString("name"));
u.setFathersname(rs.getString("fathersname"));
u.setAddress(rs.getString("address"));
u.setMobile(rs.getString("mobile"));


list.add(u);
        }
    }catch(Exception e){System.out.println(e);}
    return list;
}
public static User7 getRecordById(int id){
    User7 u=null;
    try{
        Connection con=getConnection();
PreparedStatementps=con.prepareStatement("select * from class2 where id=?");
ps.setInt(1,id);
ResultSetrs=ps.executeQuery();
        while(rs.next()){
            u=new User7();
u.setId(rs.getInt("id"));
u.setName(rs.getString("name"));
u.setFathersname(rs.getString("fathersname"));
u.setAddress(rs.getString("address"));
u.setMobile(rs.getString("mobile"));


        }
```

```
   }catch(Exception e){System.out.println(e);}
   return u;
}
}
//login form design
body{
        margin: 0;
        padding: 0;
        background: url(School.jpg);
        background-size: cover;
        background-position: centre;
        font-family: sans-serif;
}
.loginbox{
        width: 320px;
        height: 420px;
        background: #000;
        color: #fff;
        top: 50%;
        left: 50%;
        position: absolute;
        transform: translate(-50%,-50%);
        box-sizing: border-box;
```

```
            padding: 70px 30px;
}
.avatar{
    width: 100px;
    height: 100px;
    border-radius: 50%;
    position: absolute;
    top: -50px;
    left: calc(50% -  50%);
}
h1{
            margin: 0;
            padding: 0 0 20px;
            text-align: center;
            font-size: 22px;
}
.loginbox p{
            margin: 0;
            padding: 0;
            font-weight: bold;
}
.loginbox input{
            width: 100%;
```

```
	margin-bottom: 20px;

}

.loginbox input[type="text"], input[type="password"]

{

	border: none;

	border-bottom: 1px solid #fff;

	background: transparent;

	outline: none;

	height: 40px;

	color: #fff;

	font-size: 16px;

}

.loginbox input[type="submit"]

{

	border: none;

	outline: none;

	height: 40px;

	background: #fb2525;

	color: #fff;

	font-size: 18px;

	border-radius: 20px;

}

.loginbox input[type="submit"]:hover
```

```
{
        cursor: pointer;
        background: #ffc107;
        color: #000;
}
.loginbox a{
        text-decoration: none;
        font-size: 12px;
        line-height: 20px;
        color: darkgrey;
}
.loginbox a:hover
{
 color: #ffc107;
}
//login admin
<html>
<head>
<title>login</title>
<body>
```

```
<link rel="stylesheet" type="text/css" href="Login Form Design.css">


<div class="loginbox">

        <imgsrc=".\3.jfif" class="avatar">

        <h1>Login Here</h1>

        <form action="Admindashboard.html" method="post">

                <p>Username</p>

                <input type="text" name=" " placeholder="Enter Username">

                <p>Password</p>

                <input type="Password" name=" " placeholder="Enter Password">

                <input type="submit" name=" " value="Login">

                <a href="#">Lost your password?</a><br>

                <a href="#">Don't have an account?</a>


        </form>

</div>




</body>

</head>

</html>
```

```
//login teacher

<html>

<head>

<title>login</title>

<body>

<h1>login</h1>









<link rel="stylesheet" type="text/css" href="Login Form Design.css">


<div class="loginbox">

	<imgsrc=".\3.jfif" class="avatar">

	<h1>Login Here</h1>

	<form action="viewteacherinformation.html" method="post">

		<p>Username</p>

		<input type="text" name=" " placeholder="Enter Username">

		<p>Password</p>

		<input type="Password" name=" " placeholder="Enter Password">

		<input type="submit" name=" " value="Login">

		<a href="#">Lost your password?</a><br>

		<a href="#">Don't have an account?</a>
```

```
            </form>

</div>




</body>

</head>

</html>

//user form

<style>

    body{

        background-color: azure;

    }

    .button {

  background-color: rgb(223, 235, 233);

  padding: 20ps;

  border: .5px solid #25d0ee;

  float: right;

  margin-top: 10px;

}

.button:hover{

    background-color: #e64124

}
```

```
   h1{
       font-family: 'Times New Roman';
       text-align: center;
   }
   .container {
        display: flex;
        justify-content: center;
        align-items: center;
        padding-top: 80px;
}
   table{
       table-layout: fixed;
       border: 2px solid;
       max-width: 800px;
       border-color: bisque;

         }

th, td {
  padding: 10px;
}
</style>
<a href="viewusers.jsp">View All Records</a><br/>
```

```
<h1>Add New User</h1>

<form action="adduser.jsp" method="post">

<table>

<tr><td>Name:</td><td><input type="text" name="name"/></td></tr>

<tr><td>Password:</td><td>

<input type="password" name="password"/></td></tr>

<tr><td>Email:</td><td><input type="email" name="email"/></td></tr>

<tr><td>Sex:</td><td>

<input type="radio" name="sex" value="male"/>Male

<input type="radio" name="sex" value="female"/>Female </td></tr>

<tr><td>Country:</td><td>

<select name="country" style="width:155px">

<option>India</option>

<option>Pakistan</option>

<option>Afghanistan</option>

<option>Berma</option>

<option>Other</option>

</select>

</td></tr>

<tr><td colspan="2"><input type="submit" value="Add User"/></td></tr>

</table>

</form>
```

```
//view teacher information

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Insert title here</title>

</head>

<style>

     body{

        background-color: rgb(224, 228, 227);

    }

    h1{

        font-family: 'Times New Roman';

        text-align: center;

    }


       .Apply{



        border: 2px solid gray;

        background-color: rgb(224, 228, 227);

        padding: 14px 28px;

         font-size: 16px;

         cursor: pointer;
        margin: 10px;

    }




        .P {

  font-size: 20px;

  margin-right: 5px;

}

.column {

    border: 1px solid;

    background-color: rgb(207, 214, 212);

  float: left;

  width: 46%;

  padding: 10px;

  margin-left: 20px;

  text-align: center;

  margin-bottom: 20px;

}

.column1 {

    border: 1px solid;

    background-color: rgb(207, 214, 212);

  float: right;

  width: 46%;
  padding: 10px;

  margin-right: 10px;

  text-align: center;

  margin-bottom: 20px;

}

</style>

<h1>ViewTeachersInformation</h1>

<body>

<div class="column">


<button class="Apply"><a href="viewteacherlist.html">viewteacherlist</a></button>


</div>


<div class="column1">


<button class="Apply"><a href="viewusers4.jsp">viewteachersattendance</a></button>




</div>


<div class="column">
```

```
<button class="Apply"><a href="adduserform2.jsp">addstudents</a></button>

</div>

<div class="column1">

<button class="Apply"><a href="viewusers2.jsp">editstudents</a></button>

</div>

<div class="column">

<button class="Apply"><a href="viewusers2.jsp">deletestudents</a></button>

</div>

<div class="column1">

<button class="Apply"><a href="index3.jsp">viewstudentattendance</a></button>
```

```
</div>



</body>
</html>
//view teacher list
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="viewusers4.jsp">viewteachers</a>


</body>
</html>
//view user
<!DOCTYPE html>
```

```
<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>View Users</title>

</head>

<body>


<%@page import="com.javatpoint.dao.UserDao,com.javatpoint.bean.*,java.util.*"%>

<%@ tagliburi="http://java.sun.com/jsp/jstl/core" prefix="c"%>


<h1>Users List</h1>

<style>

    body{

        background-color: Fire Brick;

    }

    .button {

  background-color: rgb(223, 235, 233);

  padding: 20ps;

  border: .5px solid #25d0ee;

  float: right;

  margin-top: 10px;

}
```

```
.button:hover{

    background-color: #e64124

}

    h1{

        font-family: 'Times New Roman';

        text-align: center;

    }

    .container {

         display: flex;

         justify-content: center;

         align-items: center;

         padding-top: 80px;

}

    table{

        table-layout: fixed;

        border: 2px solid;

        max-width: 800px;

        border-color: bisque;



           }



th, td {

  padding: 10px;
}
</style>

<%
List<User> list=UserDao.getAllRecords();
request.setAttribute("list",list);
%>

<table border="1" width="90%">
<tr><th>Id</th><th>Name</th><th>Password</th><th>Email</th>
<th>Sex</th><th>Country</th><th>Edit</th><th>Delete</th></tr>
<c:forEach items="${list}" var="u">
<tr><td>${u.getId()}</td><td>${u.getName()}</td><td>${u.getPassword()}</td>
<td>${u.getEmail()}</td><td>${u.getSex()}</td><td>${u.getCountry()}</td>
<td><a href="editform.jsp?id=${u.getId()}">Edit</a></td>
<td><a href="deleteuser.jsp?id=${u.getId()}">Delete</a></td></tr>
</c:forEach>
</table>
<br/><a href="adduserform.jsp">Add New User</a>
</html>
```

Chapter 9
Conclusion

Even if schools in India were hesitant to use educational technology, they are slowly and steadily accepting school management systems for all the benefits that it offers them. In the future, as the need to adopt technological solutions for education increases, more schools will integrate a school management system for completing their daily tasks at the school. It will not only increase their productivity and make work faster and more efficient but also decrease manpower necessary for completing the smaller clerical jobs. In all aspects, the school management system will become a smarter option for schools to run more effectively.

Future Scope

School ERP is software that assists teachers, admins, parents, and students with its various features for decreasing their workload. Soon more schools will adopt the school management system to advance their school technologically. In this project, we developed an automated school management system that facilitates the various activities taking place at schools. The system developed in the project consists of windows and web applications. These are two different applications on the same database. The windows application takes most of the activities such as offline student registering, transcript and report card generation and producing the timetable. The web application facilitates attendance recording by the homeroom teachers and to view reports, to view status of students by students, teachers and parents.

References

[1] “CRUD Operations in Student Management System in Java,” *GeeksforGeeks*, May 06, 2022. https://www.geeksforgeeks.org/crud-operations-in-student-management-system-in-java/ (accessed Jul. 05, 2023).

[2] “SQL Server CRUD Operations,” *GeeksforGeeks*, Jun. 13, 2022. https://www.geeksforgeeks.org/sql-server-crud-operations/ (accessed Jul. 05, 2023).

[3] R. Banu, “SCHOOL MANAGEMENT SYSTEM”, Accessed: Jul. 05, 2023. [Online]. Available: https://www.academia.edu/4699500/SCHOOL_MANAGEMENT_SYSTEM

[4] “School Management System | School MIS | Academia ERP.” https://www.academiaerp.com/school-management-system/ (accessed Jul. 05, 2023).

[5] “Student Information Management System,” *GeeksforGeeks*, May 16, 2020. https://www.geeksforgeeks.org/student-information-management-system/ (accessed Jul. 05, 2023).

[6] “School Management Software, School Management System Application.” https://www.geniusedusoft.com/school-management-system/what-is-school-management-system.html (accessed Jul. 05, 2023).

[7] “Types of Software Testing,” *GeeksforGeeks*, Aug. 01, 2017. https://www.geeksforgeeks.org/types-software-testing/ (accessed Jul. 05, 2023).

[8] “Unit Testing | Software Testing,” *GeeksforGeeks*, Apr. 18, 2019. https://www.geeksforgeeks.org/unit-testing-software-testing/ (accessed Jul. 05, 2023).

[9] “What is DFD(Data Flow Diagram)?,” *GeeksforGeeks*, May 01, 2020. https://www.geeksforgeeks.org/what-is-dfddata-flow-diagram/ (accessed Jul. 05, 2023).