**SAVEETHA SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES CHENNAI-602105**

**STUDENT MANAGEMENT SYSTEM**

**A CAPSTONE PROJECT REPORT**

*Submitted in the partial fulfillment for the completion of the course*

**CSA4317 INTERNET PROGRAMMING WITH MOBILE APP INTEGRATION**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**Submitted by**

**K Sreedhar (192210669)**

**R Anjan Kumar Reddy (192210162)**

**Under the Supervision of**

**MS. L. Reetha**

**NOV 2024**

## DECLARATION

We, **K Sreedhar, R Anjan Kumar Reddy** students of **Bachelor of engineering** of Computer Science and Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha School of Engineering, Chennai, hereby declare that the work presented in this Capstone Project Work entitled **STUDENT MANAGEMENT SYSTEM** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics.

Supervisor,

K SREEDHAR (192210669)

R ANJAN KUMAR REEDY (192210162)

Date: 23-11-2024

Place: CHENNAI

## CERTIFICATE

This is to certify that the project entitled **“Student management system”** submitted by K Sreedhar, R Anjan Kumar Reddyhas been carried out under my supervision. The project has been submitted as per the requirements in the current semester of **B.E Computer science engineering**

Supervisor,

**Ms. L Reetha**

# Table of Contents

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** |  | **TOPICS** | **PAGE NO.** |
| 1 | **Abstract** |  | **05** |
| 2 | **Introduction** |  | **06** |
| 3 | **Project Description** |  | **07** |
| 4 | **Problem Description** |  | **08** |
| 5 | **Tool Description** |  | **09** |
| 6 | **Operations** |  | **12** |
| 7 | **Module Description** |  | **14** |
| 8 | **Implementation** |  | **16** |
| 9 | **Result** |  | **18** |
| 10 | **Conclusion & References** |  | **19** |

## ABSTRACT

A **Student Management System (SMS)** is a robust and efficient software application designed to streamline the management of student data, academic records, and administrative tasks in educational institutions. This system automates and centralizes essential functions such as student enrolment, attendance tracking, grade management, course scheduling, and communication between stakeholders, including students, teachers, and administrators.

The SMS is built with a user-friendly interface and leverages modern technologies to ensure secure data storage, real-time updates, and scalability. It reduces manual workload, minimizes errors, and enhances decision-making through detailed reporting and analytics. By integrating with existing institutional frameworks, the system fosters transparency and improves overall operational efficiency.

This project emphasizes the importance of digital transformation in education, aiming to create a seamless experience for all users while supporting the institution's academic and administrative goals. It supports role-based access control, ensuring data privacy and integrity.

Features such as real-time updates, report generation, and easy data accessibility contribute to informed decision-making and improved educational outcomes. The implementation of this system reduces manual workload, minimises errors, and facilitates seamless information flow within the institution, making it an indispensable tool for modern educational management.

The project leverages object-oriented programming principles and database integration to provide a robust, user-friendly interface for administrators, faculty, and students. Key functionalities include student registration, course enrollment, grade management, and report generation. Advanced features such as role-based access control, search functionality, and notification systems ensure a secure and seamless user experience.

## INTRODUCTION

A Student Management System (SMS) is a comprehensive platform designed to facilitate the efficient administration of educational institutions by automating various academic and administrative tasks. Utilising PHP and MySQL, a web-based SMS leverages the robust capabilities of these technologies to provide a dynamic, scalable, and user-friendly interface for managing student information, streamlining administrative processes, and enhancing communication between students, teachers, and administrators.

PHP, a popular server-side scripting language, offers the versatility and functionality required to develop interactive and dynamic web applications. Its ease of integration with various databases, including MySQL, makes it an ideal choice for developing a robust student management system. MySQL, an open-source relational database management system, provides a reliable and efficient way to store, retrieve, and manage vast amounts of data, ensuring data integrity and performance even with high volumes of transactions.

The core features of an SMS typically include student enrollment and registration, attendance tracking, grade management, timetable scheduling, and communication modules. These functionalities aim to minimise manual paperwork, reduce redundancy, and provide real time access to essential data. Additionally, an SMS can offer role-based access, ensuring that students, teachers, and administrators have appropriate permissions tailored to their specific needs and responsibilities.

Implementing a student management system using PHP and MySQL not only enhances the operational efficiency of educational institutions but also improves the overall learning experience for students. By centralising and automating critical processes, it allows educators to focus more on teaching and less on administrative burdens, ultimately fostering a more organised and effective educational environment.

This project focuses on developing a reliable and scalable Student Management System tailored to meet the evolving demands of educational institutions. By integrating cutting-edge technologies, the system aims to simplify administrative workflows, ensure data accuracy, and provide a seamless experience for all stakeholders involved in the educational ecosystem.

## PROJECT DESCRIPTION

The **Student Management System (SMS)** is a software application developed to simplify and automate the management of student-related operations within educational institutions. This system is designed to provide an all-in-one platform for handling tasks such as student registration, course allocation, attendance tracking, examination management, grading, and communication between stakeholders.

The SMS aims to replace traditional manual processes, which are often time-consuming and error-prone, with an efficient and reliable digital solution. It features a user-friendly interface that ensures easy navigation and accessibility for administrators, teachers, students, and parents. The system also incorporates a secure database to store and manage large volumes of student records, ensuring data privacy and integrity.

Key features of the system include:

* **Student Registration and Profile Management**: Allows efficient recording and updating of student details.
* **Attendance Management**: Tracks daily attendance with reporting tools for absentee analysis.
* **Academic Records and Grading**: Enables teachers to input grades and generate report cards.
* **Communication Tools**: Facilitates interaction between students, parents, and staff through notifications or announcements.
* **Analytics and Reports**: Provides insights through data-driven reports to support decision-making.

The system is developed using modern technologies to ensure scalability, security, and reliability, making it adaptable to institutions of various sizes. By implementing this system, institutions can improve operational efficiency, enhance transparency, and focus on their primary objective—providing quality education.

## PROBLEM DESCRIPTION

* **Problem Description**
* Managing student-related information in educational institutions has traditionally been a manual and time-intensive process. This method poses significant challenges, including inefficiencies, errors, and a lack of accessibility to critical data. As institutions grow, the complexity of handling large volumes of information—such as student enrolment, attendance records, academic performance, and communication—becomes increasingly difficult.
* **Key problems include:**
* **Manual Processes and Errors**: Traditional methods rely heavily on paper-based or spreadsheet systems, which are prone to errors, duplication, and data loss.
* **Time-Consuming Administration**: Administrative staff spend excessive time on repetitive tasks, such as maintaining records, calculating grades, and generating reports.
* **Inefficient Communication**: Lack of a centralized platform results in poor communication between students, parents, teachers, and administrators.
* **Data Inaccessibility**: Limited access to up-to-date student data hampers decision-making and delays responses to student or institutional needs.
* **Lack of Scalability**: Existing systems often fail to accommodate the needs of growing institutions, leading to inefficiency in managing expanding operations.
* These challenges not only hinder the smooth functioning of institutions but also divert resources from their core objective—providing quality education.
* The **Student Management System (SMS)** addresses these issues by offering a centralized, automated, and user-friendly platform that ensures accurate data handling, real-time access, and seamless communication. This system streamlines administrative tasks, enhances transparency, and supports the institution's growth while reducing the workload on staff and educators.

**Tools:**

Tools Used in Developing a Student Management SysteThe development of a Student Management System (SMS) requires a combination of software tools, frameworks, and technologies to ensure efficiency, scalability, and user-friendliness. Below are the key tools typically used:

## 1. Programming Languages

## Frontend Development:

## HTML, CSS, JavaScript: For designing user interfaces and ensuring responsiveness.

## Frameworks: React.js, Angular, or Vue.js for building interactive and dynamic frontends.

## Backend Development:

## Python: Using frameworks like Django or Flask for handling server-side logic.

## PHP: With frameworks like Laravel for robust backend development.

## Java: Using frameworks like Spring Boot for enterprise-grade applications.

## Node.js: For building scalable, event-driven applications.

## 2. Database Management

## Relational Databases:

## MySQL, PostgreSQL, or Microsoft SQL Server for structured data storage and management.

## NoSQL Databases:

## MongoDB or Firebase for flexible, unstructured data storage.

## 3. Development Environment and IDEs

## Visual Studio Code: Lightweight and extensible IDE for coding.

## PyCharm: For Python-based backend development.

## Eclipse or IntelliJ IDEA: For Java development.

## 4. Server and Hosting Tools

## Apache or NGINX: For handling web server requirements.

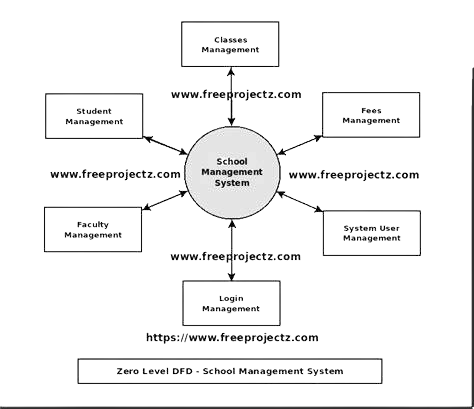
## Cloud Hosting Platforms: AWS, Google Cloud, or Microsoft Azure for scalability and reliability.

## 5. Version Control and Collaboration

## Git: For version control.

## GitHub, GitLab, or Bitbucket: For collaboration and repository management.

## BLOCK DIAGRAM



**Operations:** Operations of a Student Management System

The Student Management System (SMS) is designed to manage various administrative, academic, and communication-related tasks efficiently. Below are the key operations it facilitates:

1. Student Registration and Enrollment

* Add and maintain student profiles, including personal details, contact information, and academic history.
* Process student admissions with automated form submissions and verification.
* Assign courses and sections based on curriculum or elective choices.

2. Attendance Management

* Record daily or session-wise student attendance.
* Track absenteeism and generate attendance reports.
* Send alerts to parents/guardians for irregular attendance.

3. Academic Management

* Manage course schedules and timetables.
* Track and record student grades and academic performance.
* Automate the generation of report cards and transcripts.

4. Communication Management

* Facilitate messaging between students, teachers, and parents.
* Send announcements, reminders, or notifications via SMS, email, or in-app alerts.
* Provide access to school news, events, or circulars.

5. Fee and Payment Management

* Record and track student fee payments.
* Generate invoices and receipts automatically.
* Provide alerts for pending or overdue payments.

6. Examination Management

* Schedule and manage exams, including seating arrangements and invigilation assignments.
* Record and publish examination results.
* Analyze performance through graphical reports and insights.

7. Library Management

* Track book issuance, returns, and availability.
* Maintain a record of late returns and fines.
* Provide a search function for available resources.

8. Report Generation

* Generate detailed reports on student performance, attendance, and fee status.
* Create summaries for administrative decision-making.
* Provide data analytics to identify trends and areas for improvement.

9. Security and Access Control

* Provide role-based access (e.g., administrators, teachers, students, parents).
* Secure sensitive data with encryption and authentication protocols.

10. Customization and Scalability

* Allow institutions to customize workflows and features as per their needs.
* Support scalability to accommodate growing student populations or additional campuses.

These operations ensure that the SMS is a holistic tool for enhancing efficiency, reducing administrative workload, and improving the overall educational experience for all stakeholders.

\

**MODULE DISCRIBTION**:

**Module Description for Student Management System**

The Student Management System is divided into several modules, each focusing on a specific aspect of student and administrative management. Below is a description of the key modules:

**1. Authentication and Authorization Module**

* **Description:** Handles user login, registration, and role-based access control.
* **Features:**
  + Secure login for students, faculty, and administrators.
  + Role-based access (e.g., admin, teacher, student).
  + Password recovery and management.

**2. Student Information Management Module**

* **Description:** Maintains detailed student records and allows administrators to add, update, and delete student information.
* **Features:**
  + Student profile creation (name, contact, address, etc.).
  + Academic records (grades, courses, attendance).
  + Search and filter functionality.

**3. Course Management Module**

* **Description:** Manages course creation, allocation, and enrollment processes.
* **Features:**
  + Add or update course details (course name, code, credits).
  + Assign courses to instructors and students.
  + View and manage course schedules.

**4. Attendance Management Module**

* **Description:** Tracks and reports student attendance.
* **Features:**
  + Mark attendance for classes.

## IMPLEMENTATION

CODE ON SQL:

CREATE DATABASE student management;

USE student management;

CREATE TABLE students (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR (100) NOT NULL,

email VARCHAR (100) NOT NULL,

phone VARCHAR (15) NOT NULL,

course VARCHAR (50) NOT NULL,

created at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

HTML AND PHP CODE:

<!DOCTYPE html>

<html>

<head>

<title>Add Student</title>

</head>

<body>

<h2>Add Student</h2>

<form method="post" action="">

Name: <input type="text" name="name" required><br> Email: <input type="email" name="email

" required><br> Phone: <input type="text" name="phone" required><br>

Course: <input type="text" name="course" required><br>

<input type="submit" value="Add Student">

</form>

</body>

</html>

<?php

include 'db.php:

$sql = "SELECT FROM students";

Sresult $conn->query($sql);

<!DOCTYPE html>

<html>

<head>

<title>Student Management System</title>

</head>

<body>

<h2>Students List</h2>

<table border="1">

<tr>

<th>ID</th>

<th>Name</th>

<th>Email</th>

<th>Phone</th>

<th>Course</th> <th>Actions</th>

</tr>

<?p

if ($result->num\_rows > 0) {

while($row = $result->fetch\_assoc()) {

echo "<tr>

<td>".$row["id"]."</td>

<td>".$row["name"]."</td>

<td>".$row["email"]."</td>

<td>".$row["phone"]."</td>

<td>".$row["course"]."</td>

<td>

<a href='update.php?id=".$row["id"]."'>Edit</a>

<a href='delete.php?id=".$row["id"]."'>Delete</a>

</td>

</tr>";

}

} else {

echo "<tr><td colspan='6'>No students found</td></tr>";

}

$conn->close();

?>

</table>

<a href="create.php">Add New Student</a>

</body>

</html>

<!DOCTYPE html>

<html>

<head>

<title>Update Student</title>

</head>

<body>

<h2>Update Student</h2>

<form method="post" action="">

<input type="hidden" name="id" value="<?php echo $row['id']; ?>">

Name: <input type="text" name="name" value="<?php echo $row['name']; ?>" required><br>

Email: <input type="email" name="email" value="<?php echo $row['email']; ?>" required><br>

Phone: <input type="text" name="phone" value="<?php echo $row['phone']; ?>" required><br>

Course: <input type="text" name="course" value="<?php echo $row['course']; ?>" required><br>

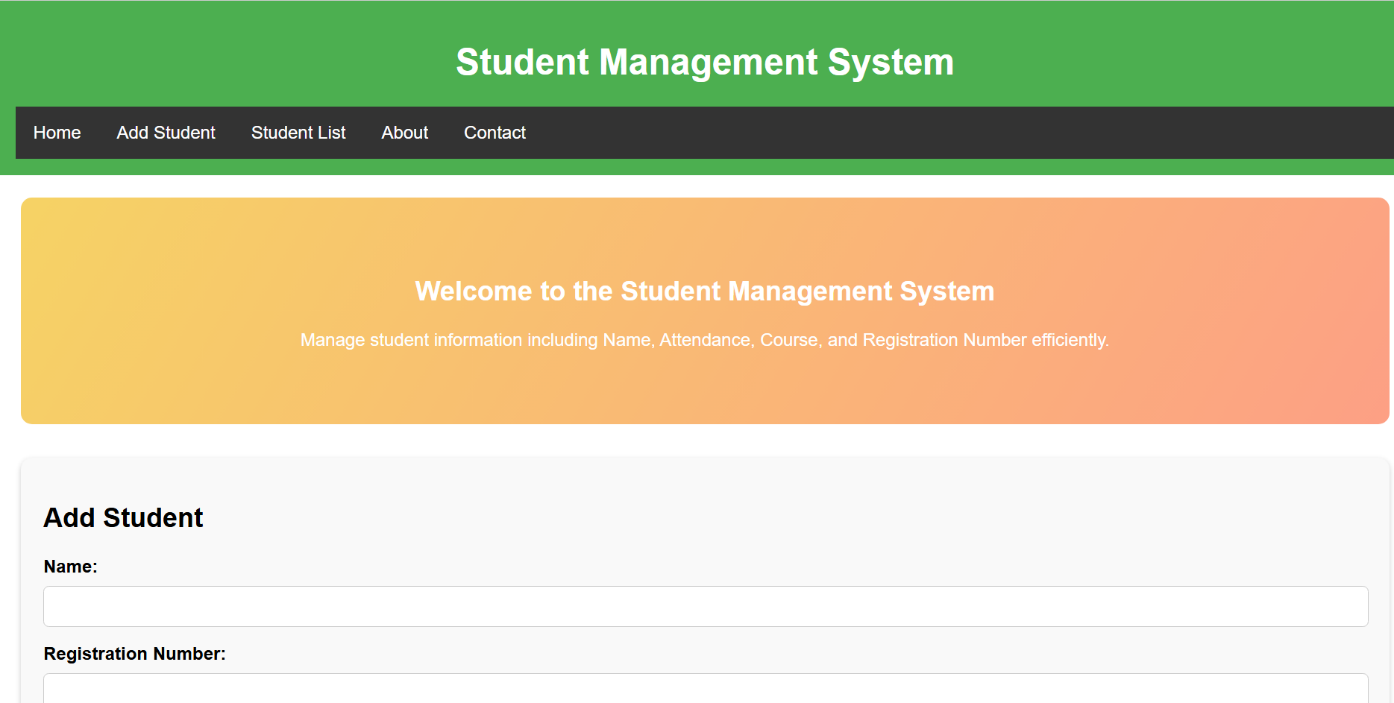
<input type="submit" value="Update Student">

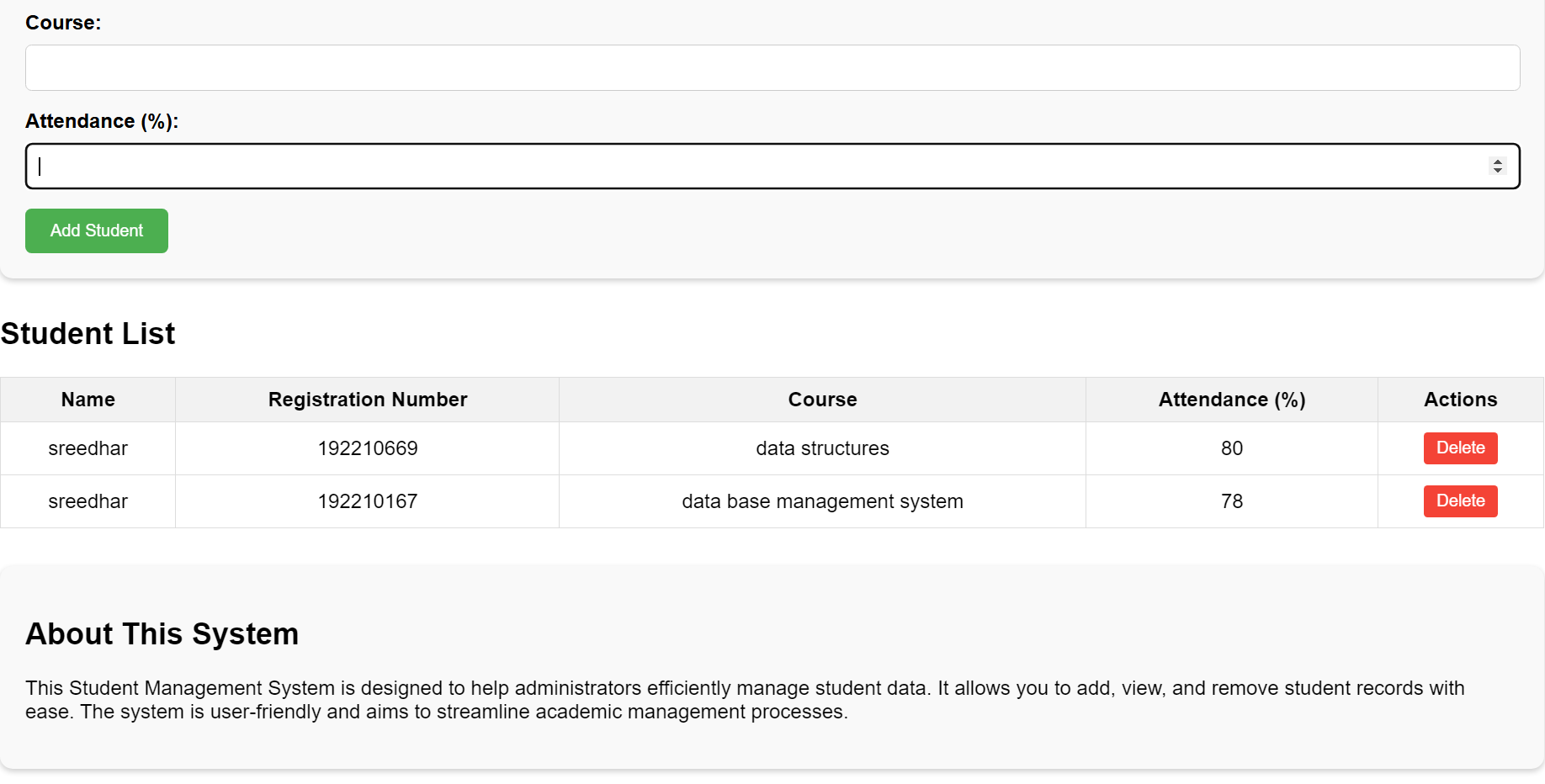
</form>

</body>

</html>

## RESULT





CONCLUSION

The Student Management System (SMS) implemented using PHP and MySQL offers a robust and efficient solution for managing educational institutions' data. This system streamlines various administrative and academic processes, including student enrolment, attendance tracking, grade management, and communication. By leveraging PHP's server-side scripting capabilities and MySQL's powerful database management, the SMS ensures secure, scalable, and dynamic handling of student information.

The integration of these technologies facilitates real-time data access, enhances user experience, and reduces manual effort, leading to increased productivity. Moreover, the system's user-friendly interface and customizable features cater to the unique needs of different institutions.

In essence, the PHP and MySQL-based Student Management System is a valuable tool for modern educational environments, promoting better organization, improved data accuracy, and more effective management of student-related activities.

In conclusion, the implementation of a Student Management System marks a critical step toward digital transformation in education, ensuring that institutions are equipped to handle the demands of a dynamic and fast-paced learning environment. With its scalability and adaptability, the SMS serves as a reliable foundation for fostering academic and administrative excellence.

**Reference:**

o Yue, Zhi-gang, and You-wei Jin. "The development and design of the student management system based on the network environment." 2010 International conference on multimedia communications. IEEE, 2010.

o Yue, Z.G. and Jin, Y.W., 2010, August. The development and design of the student management system based on the network environment. In 2010 International conference on multimedia communications (pp. 5-8). IEEE.

o Yue, Zhi-gang, and You-wei Jin. "The development and design of the student management system based on the network environment." In 2010 International conference on multimedia communications, pp. 5-8. IEEE, 2010.

o Yue, Z. G., & Jin, Y. W. (2010, August). The development and design of the student management system based on the network environment. In 2010 International conference on multimedia communications (pp. 5-8). IEEE.

o Yue, Zhi-gang, and You-wei Jin. "The development and design of the student management system based on the network environment." 2010 International conference on multimedia communications. IEEE, 2010.