



SAVEETHA SCHOOL OF ENGINEERING
SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES
CHENNAI-602105



STUDENT RESULT 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
N. Ravi Teja Reddy(192210667)

Under the Supervision of
Ms.L. Reetha

NOV 2024

DECLARATION

I am N.Ravi Teja Reddy student of Bachelor of Engineering in 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 Result 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.

N.Ravi Teja Reddy (192210667)

Date:

Place:

CERTIFICATE

This is to certify that the project entitled “**Student Result Management System**” submitted by N. Ravi Teja Reddy (192210667) has been carried out under my supervision. The project has been submitted as per the requirements in the current semester of B.E. Computer Science and Engineering

Supervisor
Ms. L.Reetha

Table of Contents

S.NO	TOPICS	PAGE NO.
1	Abstract	5
2	Introduction	6
3	Project Description	7
4	Problem Description	8
5	Tool Description	9
6	Operations	11
7	Module Description	12
8	Implementation	14
9	Result	17
10	Conclusion & References	19

ABSTRACT

The Student Result Management System (SRMS) is a web-based application designed to streamline the management, storage, and retrieval of student academic results. The system addresses the inefficiencies of traditional result handling methods, which are often time-consuming, error-prone, and difficult to scale.

The SRMS enables educational institutions to digitize the entire result management process, from data entry to report generation. The system provides distinct user roles, including administrators, teachers, and students, ensuring secure and tailored access to functionalities. Administrators can manage student and course records, while teachers can input and update marks for their respective subjects. Students can easily view their grades and academic progress in real-time.

Built using modern web technologies, the SRMS ensures high performance, data security, and user-friendly navigation. It incorporates robust authentication mechanisms and efficient database management for accurate record-keeping. The system enhances transparency and fosters seamless communication between stakeholders, contributing to a more organized academic environment.

This project demonstrates the integration of technology in education, offering a scalable and reliable solution for managing student results effectively. By leveraging this system, institutions can focus more on academic excellence while reducing administrative burdens, paving the way for an innovative and efficient academic ecosystem. Ultimately, the SRMS fosters a smoother academic experience for students, teachers, and administrators, building a foundation for a more technology-driven future in education.

INTRODUCTION

In today's digital world, educational institutions face the challenge of efficiently managing student results. Traditional methods, which often involve manual record-keeping and calculations, are time-consuming, prone to errors, and difficult to scale. The Student Result Management System (SRMS) is designed to address these issues by automating the result management process, offering a more reliable and efficient solution for storing, managing, and retrieving academic data in a secure manner. This system streamlines the entire process, enabling easy access for administrators, teachers, and students.

The primary objective of the SRMS is to replace manual result management with an automated system that ensures accurate data entry, grade calculation, and real-time result access. The system is built to handle different user roles—administrators, teachers, and students—providing them with the necessary tools and access based on their needs. Administrators can oversee the entire system, teachers can input and update grades, and students can view their results instantly. This approach improves the efficiency of result management and ensures that data is always up-to-date and accurate.

The SRMS includes key features such as automated result entry, grade calculation, and the generation of academic reports. It provides secure, role-based access for different users, ensuring that each stakeholder can only access the information relevant to them. The system also offers real-time access to results, promoting transparency and communication. With a focus on data security, the SRMS uses encryption and secure login protocols to protect sensitive information. The responsive design ensures that the system is accessible across various devices, providing convenience and flexibility to all users.

PROJECT DESCRIPTION

This project involves the design and development of a full-stack web application for a Student Result Management System (SRMS) using the MERN stack—MongoDB, Express.js, React, and Node.js. The SRMS offers an efficient platform for educational institutions to manage, store, and retrieve student academic results. It provides a seamless, secure, and automated solution for administrators, teachers, and students, ensuring accurate record-keeping and real-time access to academic performance data.

Key Features and Functionalities:

1. User Authentication and Role-Based Access:

Administrators can manage system settings, teachers can input and update grades, and students can access their academic results. This ensures data protection and privacy through secure authentication and role-based access control.

2. Result Entry and Report Generation:

Teachers can enter student grades for various assessments, and the system automatically calculates final results based on predefined criteria. The platform also generates detailed academic reports and transcripts.

3. Real-Time Result Access:

Once results are entered, students can view their academic performance instantly. This real-time access enhances transparency and allows students to track their progress throughout the academic year.

4. Admin Dashboard:

The system includes an intuitive admin dashboard for managing users, courses, and academic results. Administrators have full control over the system, enabling them to oversee student data, manage teacher accounts, and monitor the performance and also ensures smooth operations across the platform.

PROBLEM DESCRIPTION

The primary goal of this project is to build a fully functional Student Result Management System (SRMS), aimed at streamlining the process of managing and accessing academic results for educational institutions. The software addresses common challenges faced in traditional result management systems, such as inefficient record-keeping, manual grade calculations, and delayed access to results.

As educational institutions increasingly move toward digital solutions, there is a growing demand for efficient and scalable systems to manage student results. However, many institutions still rely on outdated manual processes that can be error-prone and time-consuming. This project aims to bridge that gap by developing a web-based result management platform using the MERN stack

Project Objectives:

- 1. Simplify User Experience:** Create an intuitive and user-friendly interface that enables administrators, teachers, and students to easily navigate the system, manage academic data, and access results with minimal steps and effort.
- 2. Efficient Restaurant Management:** Provide teachers with a streamlined platform for inputting grades, updating academic records, and generating result reports, ensuring accurate and efficient result management across the institution.
- 3. Seamless Order Processing and Tracking:** Allow students to access their academic results in real time, promoting transparency and enabling them to track their progress instantly after grades are entered by teachers.
- 4. Secure Payment Integration:** Implement strong security measures, such as role-based authentication and data encryption, to ensure that student records and personal information are protected and accessible only by authorized users.

TOOL DESCRIPTION

The Student Result Management System (SRMS) is built using the MERN stack—MongoDB, Express.js, React, and Node.js. MongoDB provides efficient data storage, while Express.js and Node.js handle server-side operations. React is used to create a dynamic and responsive user interface. Together, they offer a seamless experience for all users.

User Interface:

The user interface (UI) is designed to be intuitive and responsive. Built with React, each tailored to the user's needs. The UI ensures easy navigation and efficient access to academic data.

1. Faculty and Student Login Interface:

- **Faculty Login:** Allows faculty members to log in securely to manage and input student results.
- **Student Login:** Students log in using their registration number (RegNo) to access their individual results.

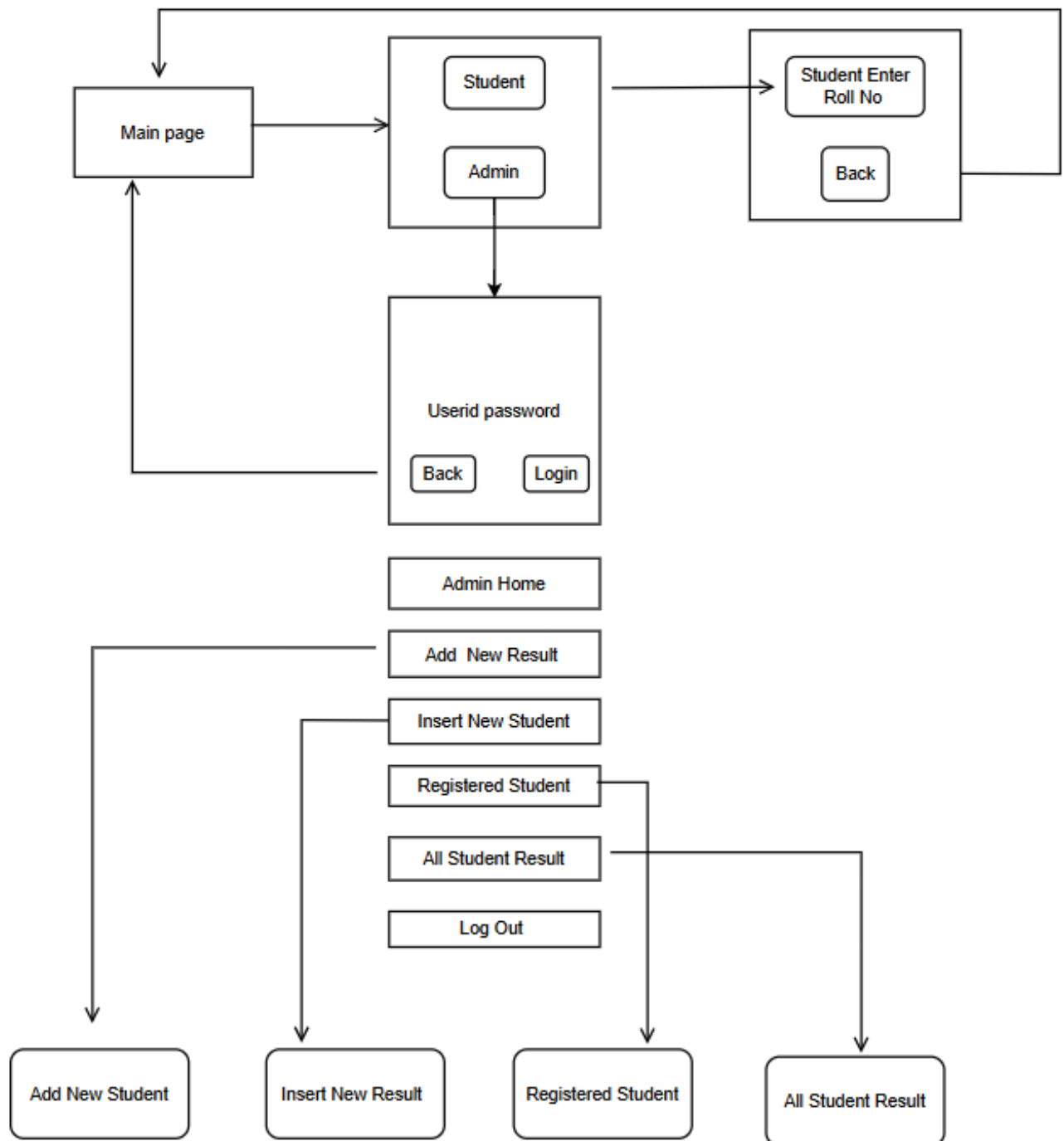
2. Dashboard Interface:

- **Admin Dashboard:** Provides an overview of system activity and management tools.
- **Side Panel Features:**
 - **Classes:** Create and manage classes.
 - **Students:** Manage student records.
 - **Results:** Faculty can create and manage student results.

3. Student Interface:

- **Student Login:** Students access their academic results by logging in with their registration number.

BLOCK DIAGRAM



OPERATIONS

In the Student Result Management System (SRMS), various essential operations are implemented to manage users, classes, students, and results. Below are the main operations and data storage requirements for the system:

1. Store Student Information:

- **Registration Number (RegNo):** A unique identifier is assigned to each student for login and result tracking.
- **Personal Information:** Stores the student's name, class, and other relevant details for academic purposes.
- **Subjects and Performance Data:** Maintains subject-wise grades and overall results for generating detailed academic reports.

2. Store Class Information:

- **Class Name and ID:** Each class is assigned a unique identifier and contains details about the associated students.
- **Subject Details:** Stores the list of subjects for each class, facilitating result entry and management.

3. Store Result Information:

- **Result ID:** A unique ID is assigned to each result entry for secure and organized tracking.
- **Student RegNo:** Links each result to a specific student for accurate record-keeping and access.
- **Grades and Scores:** Stores individual grades for each subject and calculates overall performance for academic progress tracking .

MODULE DESCRIPTION

The Student Result Management System (SRMS) is built using a modular architecture, where each module is designed to handle a specific function independently, ensuring scalability and maintainability. The Class Management Module organizes classes and subjects, while the Student Management Module handles the creation and storage of student records, linking them to their respective classes. The Result Management Module enables faculty to input and manage subject-wise grades, calculate final scores, and generate detailed academic reports.

1. User Authentication Module:

Functionality: Handles user registration, login, and secure authentication for Admin, Faculty, and Student roles.

Features:

- Secure login functionality with unique identifiers (RegNo for students, email/ID for faculty/admin).
- Password encryption and storage to ensure the safety of user data.
- Role-based access control ensuring only authorized users (Admin, Faculty, Student) can access their respective sections.
- Password reset functionality for secure access recovery.

2. Admin Management Module:

Functionality: Manages administrative functions like student registration, class management, and result oversight.

Features:

- Admins can create and manage student profiles, including registration numbers and class assignments.
- Admins can assign students to specific subjects and manage class-related details.
- Admin access to system reports and overall result monitoring, ensuring smooth operation of the platform.
- Secure admin login with role-based access to critical functions like system settings and user management.

3. Faculty Management Module:

Functionality: Allows faculty members to manage and input academic results for students.

Features:

- Faculty login with secure authentication, allowing access to their assigned subjects and student records.
- Faculty can input, update, and manage subject-specific grades for students.
- Ability to view and manage results, including generating reports for students in their classes.
- Allows faculty to track the overall performance of students in their assigned courses.

4. Student Access Module:

Functionality: Provides students with secure access to their academic records and results.

Features:

- Secure student login using a unique Registration Number (RegNo).
- Displays subject-wise grades and overall academic performance.
- Allows students to track and view their academic progress over time.
- Ensures transparency by allowing students to easily access their results and academic reports.

IMPLEMENTATION

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Student Result Management</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body class="container mt-5">
<h1 class="text-center">Student Result Management System</h1>
<h2>Add Student</h2>
<form id="addStudentForm">
<div class="mb-3">
<label for="name" class="form-label">Name</label>
<input type="text" class="form-control" id="name" required>
</div>
<div class="mb-3">
<label for="email" class="form-label">Email</label>
<input type="email" class="form-control" id="email" required>
</div>
<div class="mb-3">
<label for="class" class="form-label">Class</label>
<input type="text" class="form-control" id="class" required>
</div>
<button type="submit" class="btn btn-primary">Add Student</button>
</form>
<h2 class="mt-5">View Results</h2>
<form id="viewResultsForm">
<div class="mb-3">
<label for="studentId" class="form-label">Student ID</label>
<input type="number" class="form-control" id="studentId" required>
</div>
<button type="submit" class="btn btn-secondary">View Results</button>
</form>
<table class="table table-bordered mt-3" id="resultsTable">
<table>
```

App.py:

```
from flask import Flask, request, jsonify
from flask_cors import CORS
import pymysql
app = Flask(__name__)
CORS(app)
db = pymysql.connect(host="localhost", user="root", password="",
database="StudentResultDB")
@app.route('/add-student', methods=['POST'])
def add_student():
    data = request.json
    try:
        with db.cursor() as cursor:
            sql = "INSERT INTO Students (name, email, class) VALUES (%s, %s, %s)"
            cursor.execute(sql, (data['name'], data['email'], data['class']))
            db.commit()
        return jsonify({"message": "Student added successfully"}), 201
    except Exception as e:
        return jsonify({"error": str(e)}), 400
@app.route('/results/<int:student_id>', methods=['GET'])
def get_results(student_id):
    try:
        with db.cursor() as cursor:
            sql = "SELECT Students.name, Subjects.name AS subject, Results.marks,
Results.grade FROM Results INNER JOIN Students ON Results.student_id =
Students.student_id INNER JOIN Subjects ON Results.subject_id = Subjects.subject_id
WHERE Students.student_id = %s"
            cursor.execute(sql, (student_id,))
            results = cursor.fetchall()
            return jsonify(results), 200
    except Exception as e:
        return jsonify({"error": str(e)}), 400
@app.route('/add-result', methods=['POST'])
def add_result():
    data = request.json
    try:
        with db.cursor() as cursor:
            sql = "INSERT INTO Results (student_id, subject_id, marks, grade) VALUES
(%s, %s, %s, %s)"
            cursor.execute(sql, (data['student_id'], data['subject_id'], data['marks'],
data['grade']))
            db.commit()
            return jsonify({"message": "Result added successfully"}), 201
    except Exception as e:
        return jsonify({"error": str(e)}), 400
if __name__ == "__main__":
```

Styles.css

```
/* General Body Styling */
body {
  font-family: Arial, sans-serif;
  background-color: #f8f9fa;
  color: #333;
  margin: 0;
  padding: 0;
}

/* Header Styling */
h1 {
  font-size: 2.5rem;
  margin-bottom: 2rem;
  color: #007bff;
  text-align: center;
}

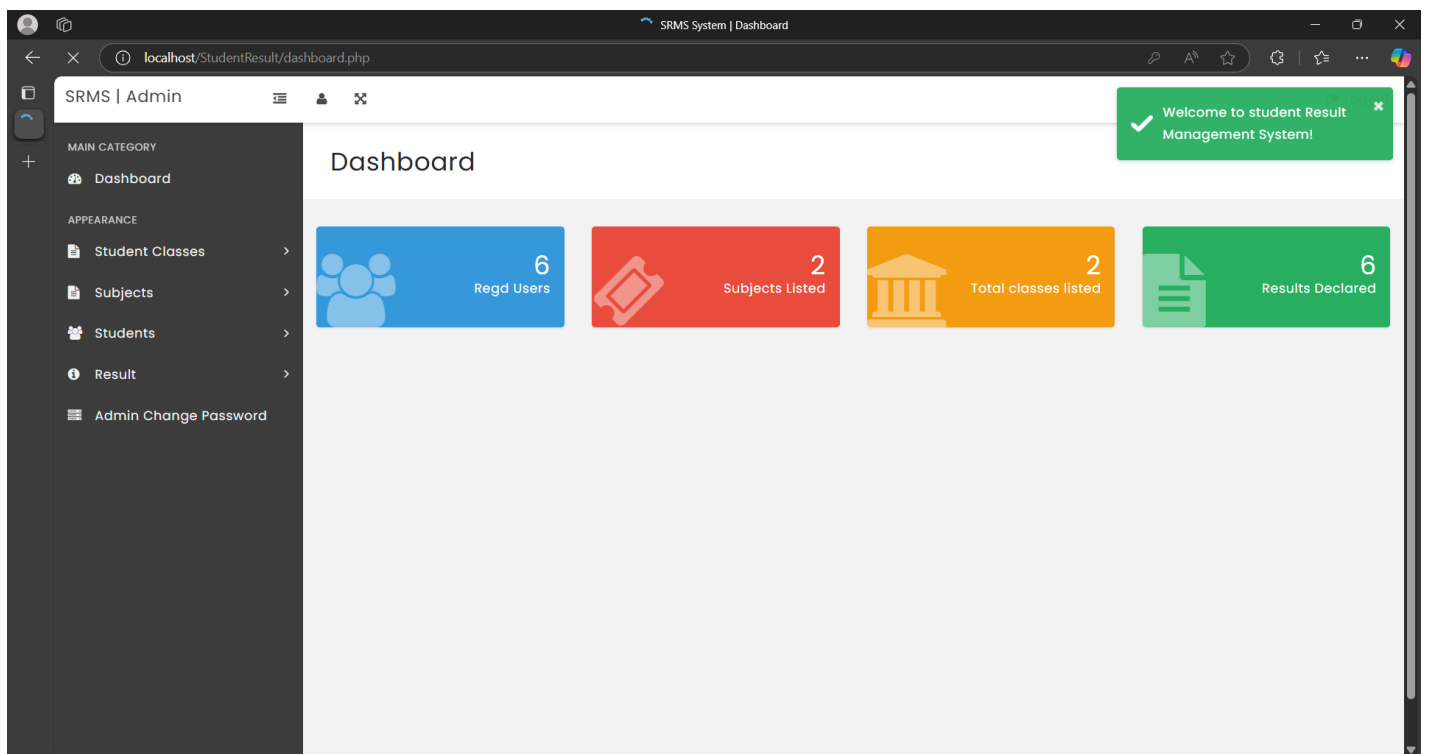
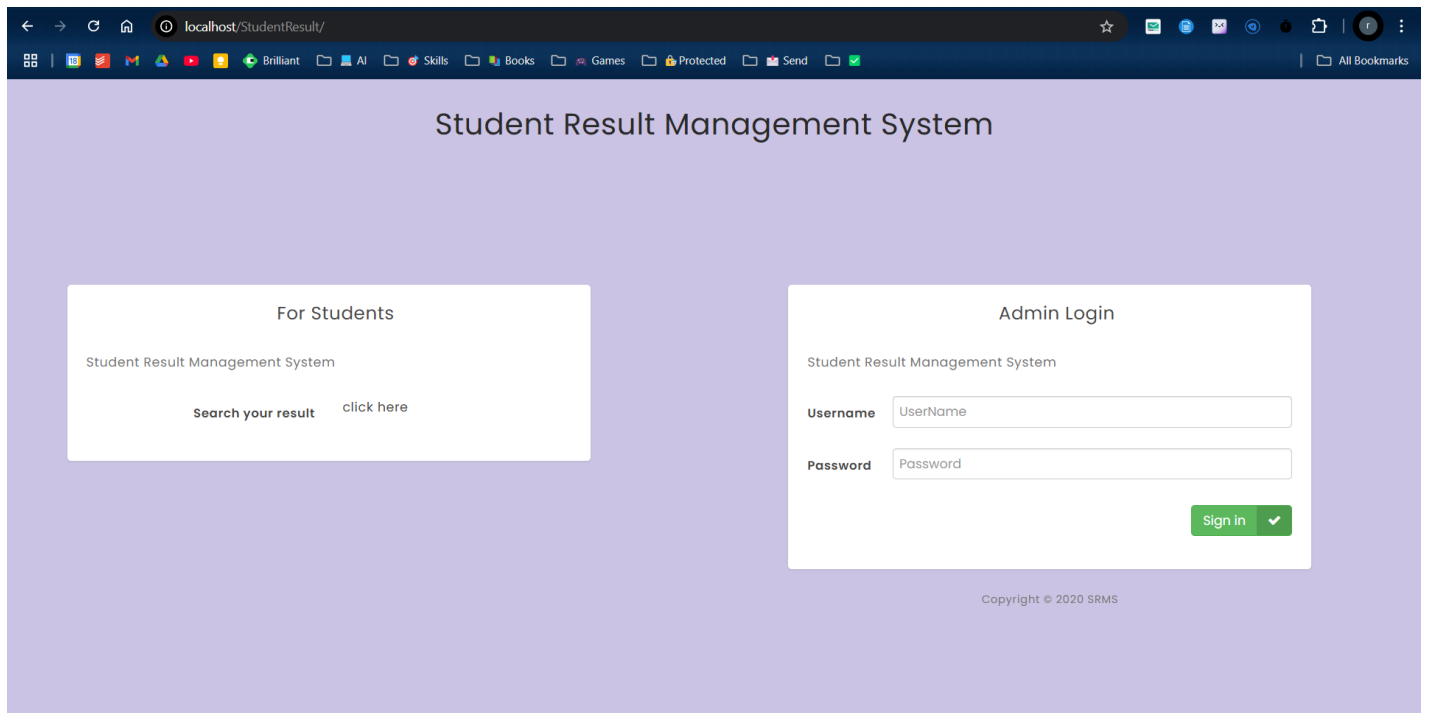
h2 {
  font-size: 1.75rem;
  color: #343a40;
  margin-top: 2rem;
}

/* Form Styling */
form {
  background: #ffffff;
  border: 1px solid #ced4da;
  border-radius: 5px;
  padding: 20px;
  box-shadow: 0px 2px 4px rgba(0, 0, 0, 0.1);
  margin-bottom: 2rem;
}

form label {
  font-weight: bold;
}

form .form-control {
  border-radius: 4px;
  border: 1px solid #ced4da;
  padding: 10px;
  font-size: 1rem;
}
```


RESULT



SRMS | Admin

MAIN CATEGORY

Dashboard

APPEARANCE

Student Classes

Subjects

Students

Result

Admin Change Password

SRMS Admin Manage Students

localhost/StudentResult/manage-students.php

Logout

Manage Students

Home / Students / Manage Students

View Students Info

Show 10 entries

Search:

#	Student Name	Roll Id	Class	Reg Date	Status	Action
1	NAGELLA RAVITEJA REDDY	0667	Operating System(A)	2024-11-20 10:17:31	Active	
2	NAGELLA RAVITEJA REDDY	0667	programming with java(A)	2024-11-20 10:17:56	Active	
3	UDATHA SREE HARSHA VARDHAN	1145	Operating System(A)	2024-11-20 10:18:48	Active	
4	YEDDULA DIVYASRI	0138	Operating System(A)	2024-11-20 10:19:52	Active	
5	YEDDULA DIVYASRI	0138	programming with java(A)	2024-11-20 10:20:31	Active	
6	UDATHA SREE HARSHA VARDHAN	1145	programming with java(A)	2024-11-20 10:40:36	Active	
#	Student Name	Roll Id	Class	Reg Date	Status	Action

Showing 1 to 6 of 6 entries

Previous

1

Next

Student Result Management System

localhost/StudentResult/result.php

Result Management System

Student Name : NAGELLA RAVITEJA REDDY

Student Roll Id : 0667

Student Class: Operating System(A)

#	Subject	Marks
1	Operating System with Raspberry PI 5	75
Total Marks		75 out of 100
Percentage		75 %
Download Result		Download

Back to Home

CONCLUSION

The Student Result Management System has been developed to create an efficient, reliable, and user-friendly platform for managing student academic records. By utilizing modern web technologies and a modular architecture, the system ensures that Admins, Faculty, and Students can easily access and manage the data relevant to them. The system's flexibility and scalability make it an ideal solution for educational institutions of various sizes, addressing key challenges in academic record-keeping and student performance tracking.

Throughout the development of this system, we have achieved several key objectives:

- Implemented secure user authentication to protect student and faculty data.
- Created a streamlined interface for students to easily access and track their academic performance.
- Developed a faculty interface that allows easy input and management of student grades.
- Designed an admin interface that supports efficient management of students, classes, and results.

The SRMS not only meets the immediate needs of educational institutions but is also designed with future enhancements in mind. The modular approach enables features like integrated feedback systems, automated performance analysis, and enhanced reporting to be added seamlessly in the future.

Overall, this project has provided valuable hands-on experience in full-stack web development, especially in the context of building secure and user-focused systems for educational purposes. The **SRMS** demonstrates our ability to design and implement a solution that prioritizes usability, security, and scalability, offering a solid foundation for further improvements in the management of student data and academic performance and results.

References:

1. Beasy, Kim, Sherridan Emery, and Joseph Crawford. "Drowning in the shallows: An Australian study of the PhD experience of wellbeing." *Teaching in Higher Education* 26.4 (2021): 602-618.
2. Walia, Er Saurabh, and Er Satinderjit Kaur Gill. "A framework for web based student record management system using PHP." *International Journal of Computer Science and Mobile Computing* 3.8 (2014): 24-33.
3. Al-Hawari, Feras, et al. "The software engineering of a three-tier web-based student information system (MyGJU)." *Computer Applications in Engineering Education* 25.2 (2017): 242-263.
4. Jasmi, Jamaluddin, et al. "An analysis model for an integrated student activities management system for higher education during RMO/CMCO/PASCA COVID-19 period in Malaysia." *Procedia Computer Science* 179 (2021): 798-803.
5. Jasmi, Jamaluddin, et al. "An analysis model for an integrated student activities management system for higher education during RMO/CMCO/PASCA COVID-19 period in Malaysia." *Procedia Computer Science* 179 (2021): 798-803.
6. Daim, Tugrul, et al. "Exploring student information management system adoption post pandemic: Case of Turkish higher education." *Technology in Society* 77 (2024): 102557.
7. Al-Hawari, F. "MyGJU student view and its online and preventive registration flow." *International Journal of Applied Engineering Research* 12.1 (2017): 119-133.
8. Dewi, Meta Amalya, and Rafi Irham. "Penerapan Agile Scrum Pada Pengembangan Aplikasi Bimbingan Daring Skripsi Mahasiswa." *Jurnal SISKOM-KB (Sistem Komputer dan Kecerdasan Buatan)* 4.2 (2021): 40-45.
9. Sihotang, Ellen Theresia, and Hariadi Yutanto. "Tata Kelola Organisasi Mahasiswa Melalui Pengembangan Sistem Informasi." *MATRIK: Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer* 21.1 (2021): 99-110.
10. Suryandani, Farid. "Pengembangan Sistem Informasi Akademik Berbasis Web Sebagai Sistem Pengolahan Nilai Siswa Di Smk Negeri 1 Kudus." (2016).
11. Sabri, Shaimaa Q., Akeela M. Ahmad, and Maiwan B. Abdulrazaq. "Design and implementation of student and alumni web portal." *Science Journal of University of Zakho* 5.3 (2017): 272-277.
12. Patnaik, S., Kumari Singh, K., Ranjan, R., & Kumari, N. (2016). College management system. *International Research Journal of Engineering and Technology (IRJET)*, 3(5), 659-661.
13. Aziz, Azlan Abdul. "Design architecture of an integrated student activities management system for higher education." *Turkish Journal of Computer and Mathematics Education (TURCOMAT)* 12.5 (2021): 1676-1683.
14. Renaningtias, Nurul, and Dyah Apriliani. "Penerapan metode prototype pada pengembangan sistem informasi tugas akhir mahasiswa." *Rekursif: Jurnal Informatika* 9.1 (2021).
15. Jain, Samkeet, et al. "Smart university-student information management system." 2017 *International Conference On Smart Technologies For Smart Nation (SmartTechCon)*. IEEE, 2017.