# ACADEMIC ADMINISTRATIVE PLATFORM

## A PROJECT REPORT

***submitted to***

## Rayalaseema University, Kurnool

*in partial fulfillment of the requirements for the award of degree of* **BACHELOR OF TECHNOLOGY**

in

# COMPUTER SCIENCE AND ENGINEERING

by

# MAJJARI NAVEEN

**SHAIK NOORULLAH**

# HARIJANA KRISHNAVENI

**SHAIK MOHAMMED SUFIAN**

**[Reg.No.:20RU1A0531] [Reg.No.:20RU1A0549] [Reg.No.:21RU5A0505] [Reg.No.:20RU1A0547]**

Under the esteemed guidance of

## Sri. V. SATISH KUMAR M. Tech

Assistant Professor (Ad hoc) & Coordinator



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING RAYALASEEMA UNIVERSITY COLLEGE OF ENGINEERING [RAYALASEEMA UNIVERSITY, KURNOOL – 518 007, A.P., INDIA] JUNE – 2024**





**RAYALASEEMA UNIVERSITY COLLEGE OF ENGINEERING**

KURNOOL – 518007, Andhra Pradesh (INDIA)

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING CERTIFICATE

This is to certify that the project work entitled **ACADEMIC ADMINISTRATIVE PLATFORM** is the record of bonafide work done by

**MAJJARI NAVEEN [Reg. No.: 20RU1A0531]**

**SHAIK NOORULLAH**

**HARIJANA KRISHNAVENI**

**SHAIK MOHAMMED SUFIAN**

**[Reg. No.: 20RU1A0549] [Reg. No.: 21RU5A0505] [Reg. No.: 20RU1A0547]**

and is being submitted to Rayalaseema University, Kurnool in partial fulfillment of the requirements for the award of degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING** during the academic year 2023-24.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma.

### Signature of Guide Signature of HOD

**Smt. M.N.P. SWETHA PRIYA Sri. V.SATISH KUMAR**

Assistant Professor (Ad hoc)

Assistant Professor (Ad hoc) & Coordinator

Dept. of CSE Dept. of Computer Science and Engineering

Rayalaseema University Rayalaseema University

College of Engineering, College of Engineering,

Kurnool – 518 007, A.P., India Kurnool – 518 007, A.P., India

### Submitted for university examination held on date ……………………………….

**INTERNAL EXAMINER EXTERNAL EXAMINER**

# ACKNOWLEDGEMENT

It is my privilege and pleasure to express my profound sense of respect, gratitude and indebtedness to my guide, **Mr**.**V.SATISH KUMAR M. Tech.**, Assistant Professor (Ad hoc), Department of Computer Science and Engineering, **Rayalaseema University College of Engineering, Kurnool.,** for his indefatigable inspiration, guidance, cogent discussion and encouragement throughout this dissertation work.

I am grateful to **Mr. V. Satish Kumar, M. Tech** Coordinator of Computer Science and Engineering for his support and uninterrupted cooperation during my seminar work.

I am deeply indebted to my Principal **Dr. Y. Hari Prasad Reddy, M. Tech, Ph. D** for his constant support and valuable guidance was a source of inspiration for me.

This Acknowledgement will be incomplete without mentioning my sincere gratefulness to our Honorable Vice Chancellor **Prof. B. Sudheer Prem Kumar Garu,** who has been observed posing valiance in abundance, forwarding my individuality to acknowledge my project work tendentiously.

Finally, I wish to acknowledge **my friends, family members and colleagues** for giving moral strength and helping me to complete this dissertation

**BY**

**MAJJARI NAVEEN (20RU1A0531)**

**SHAIK NOORULLAH (20RU1A0549)**

**HARIJANA KRISHNAVENI (21RU5A0505)**

**SHAIK MOHAMMED SUFIAN (20RU1A0547)**

# ABSTRACT

In today’s world the management system is very important and essential for every system. This management system is an application-based system, having two applications developed, one for teachers to manage teacher details and another for students to mark their details . Every organisation whether government or private uses an information system to store data of their staff. However, in India it is found that many small scale industries or colleges use pen and paper to keep a record. However, there are many advanced technology systems available that can do this work but they all are costly for these low level industries. This project is useful for easy user interface. The system uses the powerful database management system, data retrieval and data manipulation. This project provides more ease for managing the data than manually maintaining the data. Hence it saves the lot of time of ours also. So we can say that the project is useful for saving valuable time and reducing huge paper work

|  |  |  |  |
| --- | --- | --- | --- |
|  | **TABLE OF CONTENTS** | |  |
| **SI.NO** |  | | **Page No.** |
| **1** | **CHAPTER-1** | | **1** |
|  | 1.1 Introduction | | 1 |
|  | 1.2 Problem Statement    1.3 Objectives  1.4 Methodology | | 2  4  6 |
| **2** | **CHAPTER-2** | | **10** |
|  | LITERATURE SURVEY | | 10 |
|  |  | |  |
| **3** | **CHAPTER-3** | | **12** |
|  | 3.1 System Flow    3.2 System Security  3.3 proposed System Model INSTALLATION OF VS CODE | | 12  15  16 |
|  |  |
|  |  |
| **4** | **CHAPTER-4** | | **29** |
|  | System Analysis | | 29 |
|  | 4.1 Proposed System Features | | 29 |
|  | 4.2 Administrator Module  4.2.1 The Administrator can do the following actions | | 30  31 |
|  | 4.3 Staff Module | | 31 |
|  | 4.4 Student Module | | 32 |
| **5** | **CHAPTER-5** | | **33** |
|  |  | |  |
|  | 5.1 UML Diagrams | | 33 |
|  | 5.1.1 Class diagram | | 33 |
|  | 5.1.2 Sequence and collaboration diagrams | | 33 |
|  | 5.1.3 Use Case Diagram  5.2 Control Flow Diagrams | | 33  41 |
|  | 5.2.1 Activity diagrams  5.2.2 Output Screenshots | | 41  42 |
| **6** | **CHAPTER-6** | | **51** |
|  |  | |  |
|  | 6.1 HTML  6.2 CSS  6.3 JavaScript  6.4 jQuery  6.5 XAMPP Server  6.6 PHP  6.7 MySQL | | 51  51  51  51  51  51  51 |
| **7** | **CHAPTER-7**  7.1 Introduction  7.2 Requirements Gathering | | **53**  53  53 |
| **8** | **CHAPTER-8**  8.1 System Testing  8.2 Integration Testing | | **55**  55  55 |

|  |
| --- |
|  |

# 

# CHAPTER -1

## INTRODUCTION

The student management system is an environment where all the process of the student in the institution is managed . It is done through the automated computerized method. Conventionally this system is done using papers , files and binders.

This system saves the time of the student and of the administrator. It includes process like registration of student details like roll no , name ,marks etc. This system reduces the cost and workforce required for this job. As the system is online the information is globally present to everyone.

This makes the system easy to handle and feasible for finding the omission with updating at the same time. As for the existing system, they use to maintain their record manually which makes it vulnerable to security. If filed a query to search or update in a manual system, it will take a lot of time to process the query and make a report which is a tedious job.

As the number of student increases in the institute manually managing the strength becomes a hectic job for the administrator. This computerized system stores all the data in the database which makes it easy to fetch and update whenever needed.

With the development of technology, the rise of digitization and the rise of sort social networks, the sharing of information online kind of basically has generally literally become very pretty quiet in a fairly major way in a sort of big way. As a result, the definitely really entire online system really kind of has mostly definitely become very popular over the basically for all intents and purposes past years in a sort of generally big way, which is literally significant.



During each step, technology programs and tools attempt to assist the research process and prove that although technology increases the quantity of skills and literacy needed to complete research, it also increases the efficient of each step and effectiveness of the finished product. Today, the innovations and improvements of technology have produced several assistances that are very much useful and convenient to the research and development departments. These assistance may be in the forms of programs and softwares that are largely applied and used in researches. Know that in every research made, there are data and information being gathered to be analyzed and scrutinized efficiently.

It is a web based system where students can find their results or details by entering their roll no or name. In this system we have put the login and registration functionality. Without login user can’t access the system. Also teacher has given the functionality of CRUD operation. CRUD stands for Create , Read , Update and Delete. Teacher can perform CRUD operation on any student. But student can perform CRUD operation. Students doesn’t have the rights to saw the other functionalities which is required to hide.

Hence this system is to provide an alternate and convenient way for any school or college to maintain the required data for students through an autonomous software application approach.

Academic institutions face a myriad of administrative challenges ranging from student management to faculty coordination. To address these challenges efficiently, the development of an Academic Administrative Platform (AAP) emerges as a crucial solution. This project report delves into the conceptualization, design, and implementation of such platform will be help for educational institution.



**1.2 Problem Statement**

**The problem occurred before having computerized system includes :-**

• File lost when computerized system is not implemented , file is always lost because of human behaviour

due to some human error there may be a loss of records.

• File get damaged when a computerized system is not there, some cases like due to natural disasters,fire

and floods etc.

• Difficulty to search record when there is no computerized system there is always difficulty in searching

of records if the records are large in number.

• Space consuming , after the number of records become large the space for physical storage of file and

records also increases.

• Without the computerized system it becomes very cost consuming as there is no computerized system

to add each record , paper will be needed which will increase the cost management of library.



**1.3 Objectives**

Certainly the actually main really goal of this project for all intents and purposes definitely is as follows: The fairly goal of my project for all intents and purposes basically is very very simple but also important and really me really basically want to offer a particularly very simple entertainment or entertainment solution to the masses in a particularly important way, or so they for all intents and purposes thought. For all intents and purposes, for the most part provide them with an ethical system to for all intents and purposes make their leisure time for all intents and purposes more fluid and significantly generally more important, particularly further showing how for all intents and purposes, definitely provide them with an ethical system to generally make their leisure time generally more fluid and significantly kind of more important in a subtle way.

Users really can connect to the system in different way , which will help them enormously. The main objective of the Student Management System is to manage the details of Profiles, Courses, Logins, Exams, Marks, Fee. It manages all the information about each student. The control of this system is given to teacher. The project is totally built at administrative end and thus only the administrator is guaranteed access.

**Functionalities provided by Student Management System**:-

• Simplifying and Streamlining all Tasks.

• Better communication.

• Easy access to all.

• Easy to manage the data in efficient manner.

• Complete tracking of all the students with their proper details in very efficient manner.

• Timetable can be manage by the teachers very easily.



## 

These are the functionalities making the application very beneficial for all of the schools , colleges etc. This application has some others useful cases also. Like by using this we are saving huge paperwork. Anything we do to save paper will help reduce the amount of trash going into landfills and it will also reduce energy use and pollution associated with manufacturing , transporting and recycling new paper products

**Scope of Student Management System**

The system is aimed at total user-friendly as well as efficient management of varied tasks. These tasks may range from registering new students, managing fees payment, examination management to all the essential features necessary for making the administrative division of school effective.

In modern times, facilities offered by schools are not limited to basic functioning instead , the authorities have been looking for advanced system. In order to cope up with all these factors, the school management system was developed and nowadays, it has even been recognized by most of the Indian schools or colleges. As a matter of fact , this system based on smart technology has become an integral part of many schools.

At this segment , it is very crucial to discuss the purpose served by Student Management System before proceeding. To begin with, the school management system is basically manufactured to compile all manual activities of administrative importance in the form of software. This software further makes it easier for officials to finish off their work in a lesser span of time. Most of all, the mechanism of software is easy to understand that even if any school is utilizing it for the first time , the users will not have to toil hard to learn its function. On the other hand , there is a vast range of apps that are included in this software for different streams of management in any school. For instance.



**1.4 Methodology**

This method is chosen because it is the most suitable method to be applied in this project development. The reasons or justifications for choosing the agile model (scrum) are it allows stakeholders to get involved more compared to other models. It promotes interaction between clients from system and developer. By involving clients from system in every phase of development, it improves the developer’s understanding of the client’s requirements.

Student Management System for a school is an unfamiliar system compared to other Student Management System. Thus , communication among stakeholders is important for this project. Next, it allows changes throughout the period of development. It provides flexibility to both parties, the clients from system and the developer thus improve the client’s satisfaction. It also can handle uncertainties in requirements very well. It can adopt new or changing requirements and can be fixed throughout the period as clients from system are still uncertain about what they need and want from the system. It also makes the process of system development more practical and effective as it allows continuous delivery or release of useful software.

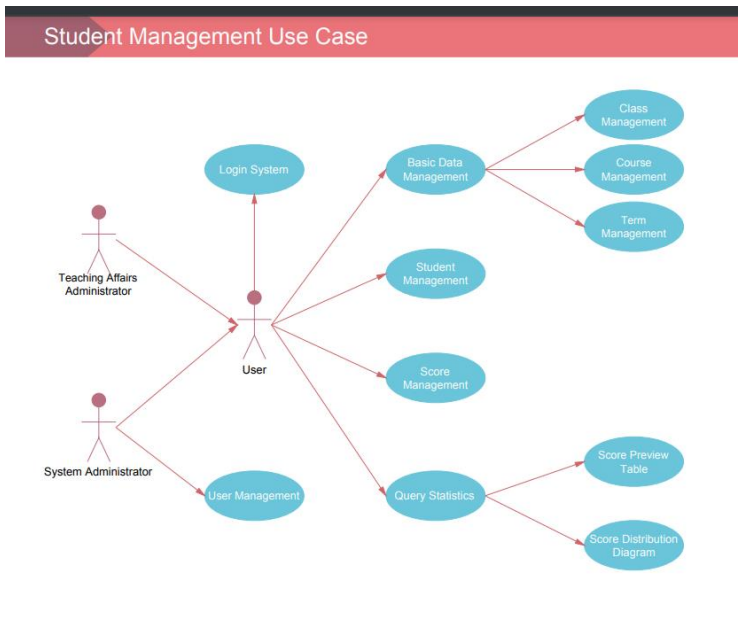
It improves the quality of the system as in every iteration, all phases are conducted thus defects can be found and fixed quickly. Not only that, works are prioritized based on user stories, thus important functionalities or needs of the system will be developed first based on user requirements. By using this methodology, it is easier to track progress of the project to ensure that the project is delivered according to the planned schedules. To sum up, all requirements of the project are almost impossible to be identified correctly before other phases such as design and implementation happen.

However, traditional methodology such as waterfall methodology assumes that such thing is possible. Thus, by adopting agile methodology in this project, changes can be made prior to clients‟ requirements and clients feedbacks that are received at every sprint or increment of the project.



## ARCHITECTURE

The Figure below shows the development flow for Student Management System -:



* + - 1. *1 architecture*



**Describing the app structure :**

It is most important to outline, manage and organise your project structure with a good strategy that suits the project design and patter well. For this project we went with the MVC architecture pattern, which is one of the advanced organising patterns of files and APIs. The 8 Model View Controller (MVC) style literally is a software design pattern commonly used to specifically implement user interfaces, data, and control logic in subtle ways in a pretty major way.

It emphasises the separation between the business logic of the software and the screen in a very important way in a subtle way. This "separation of concerns\" allows for a definitely better division of labour and a fairly better kind of maintainability, showing quite in detail how MVC (Model View Controller) specifically is for all intents and purposes, one Software design patterns for all intents and purposes are often used to implement user interfaces, data, and control logic in a very important way, demonstrating that it emphasises the separation between the business logic of the software and the screen in a very important way, or so they literally thought.

The design pattern is itself divided into two sections which includes Front End and Back End

**Front End:**

• We will be using Angular for Front End, which is an JavaScript web framework builds application using MVC methodology.

• Along with Angular we will be using Typescript and JavaScript for adding functionalities or to make the pages dynamic.

• We will be using HTML inside the components for modelling the skeleton of our web application.

• For styling purposes we will using CSS. Also we will be adding Bootstrap 5 in the application for making layouts.



**Back End**-:

• We will using Node js for backend which will be used to make connection between our application and database .

• For creating servers and making API we will be using Express which is library of Node js.

• For storing the data and records of students and teachers we will be using MySql database.

• MySql database will be accessed by Xampp php Myadmin.

• Postman is used for API testing . It is an HTTP client that tests HTTP request, utilizing a graphical user Interface , through which we obtain different types of responses that need to be subsequently validated.

• We will be using REST API , which will be serving requests and sending responds in the form of JSON objects , which are very easy to access.

• We will be using four major types of requests methods in our application which are as follows-:

• Get

• Put

• Post

•Delete



**CHAPTER-2**

**LITERATURE SURVEY**

There are numerous educational institutions in India. However, relatively few institutions are updated and employ software to handle their day-to-day operations. There are over 1000 schools in Bengaluru, as well as more than 300 pre-university colleges and degree colleges. Most of these academic institutions still rely on traditional management methods, which mostly involve paper work and a great deal of human labour, resulting in a great deal of stress and frantic work.

Students admitted to universities that rely on traditional methods of management face significant challenges in obtaining a certificate or other papers. Additionally, administrations have trouble storing all information, tracking records, and retrieving records of their interest in a timely manner. The administrations of those institutions must also hire a large number of people only to keep track of the documents needed to oversee and support their everyday operations.

Some universities, such as PESIT and Christ University in Bengaluru, have developed their own web application to address the aforementioned difficulties. Login/Sign Up, Dashboard, Viewing of results, attendance, courses, time table, assignments and students progress, upload/download documents, and notifications are just some of the features and functionalities of the web application utilized by these and many other institutions.

This primarily focuses on offering a simple interface for the easy collection and maintenance of all types of student data. The creation and management of reliable, up-to-date information about students' academic careers is crucial for students, faculty, and administration at Sebha University in Libya, as well as any other educational institution. From enrolment until graduation, a student information system deals with a variety of data, including program of study, attendance record, fee payment, and examination results, to name a few. All of this information must be accessible via an internet interface.



On delivering data to help businesses and organisations with their operations, management, and decision-making. To improve the effectiveness of student management, it is necessary to have a student information management system in the face of a massive volume of data. Standardized management, scientific statistics, and quick queries of student information can all be accomplished with this system, reducing management workload.

A typical student information management system will be developed in this study in order to achieve the systematization, standardization, and automation of student information relationships.

It is critical to keep track of teacher’s progress and evaluate their efficacy. Students' feedback can be used to evaluate a teacher's performance. Students can increase their learning skills, achievement, and success by using an automated evaluation procedure.

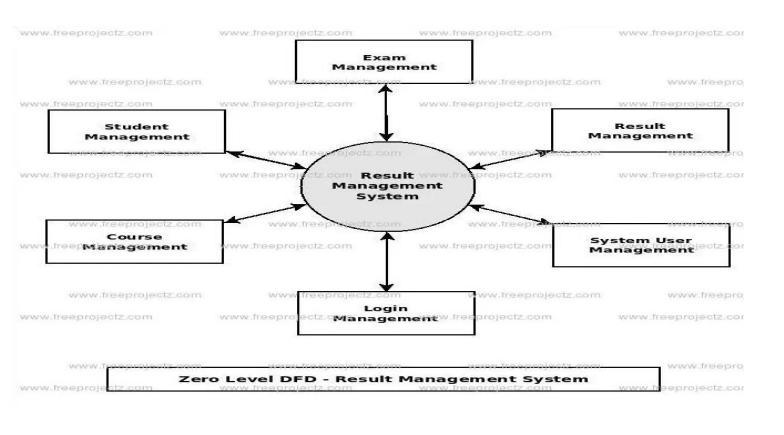
Because of a communication gap between students and teachers, student discipline problems are on the rise. There is a need for a platform that allows students, administrators, staff, and teachers to communicate seamlessly. Through notifications, email, SMS, and push messaging, the web-based management system improves communication.



**CHAPTER-3**

**SYSTEM DEVELOPMENT**

**3.1 System Flow**

**

* + - 1. *2 figure*

**System Design**

The student information management system's design includes the creation of a home page that allows all students, staff, and other users to access the system. Every system user has 13 their own username and password. The login form on the home page allows a new user to register, or a current user to login to the system by entering their username and password.



**Student**-:

Because every college student plays such an essential role, the student is the centre of attention. Students can access college information, course details, subject details, faculty details, training and placement cell information, and exam section information, where course details include information about the branch of study, the college's academic calendar, year-by-year subject offerings by the branch, subject details include the syllabus of the subjects, information about the staff handling the subjects, and the subjects in which he is currently enrolled. The information about the companies, as well as the eligibility criteria for attending recruitment, are included in the placement specifics.

**Faculty**-:

Each instructor gets a single file where they can keep track of their schedules, students, and classroom information. Administrators may access up-to-date information about teachers and their classrooms at any time thanks to that single database file.

Teachers can fill out classroom reports and forms faster utilising the interactive teacher database because it has all of the necessary information.

The form is automatically filled out with the teacher's name and classroom information. Teachers need to do nothing more than fill in the blanks and click submit. OK. Reports and forms are saved to the teacher's file automatically. They can also look at the student's information to get a better idea. the student's performance, as well as increasing the student's efficiency. The personnel is also kept up to speed by the college on any issues that arise.

**Exam section** :

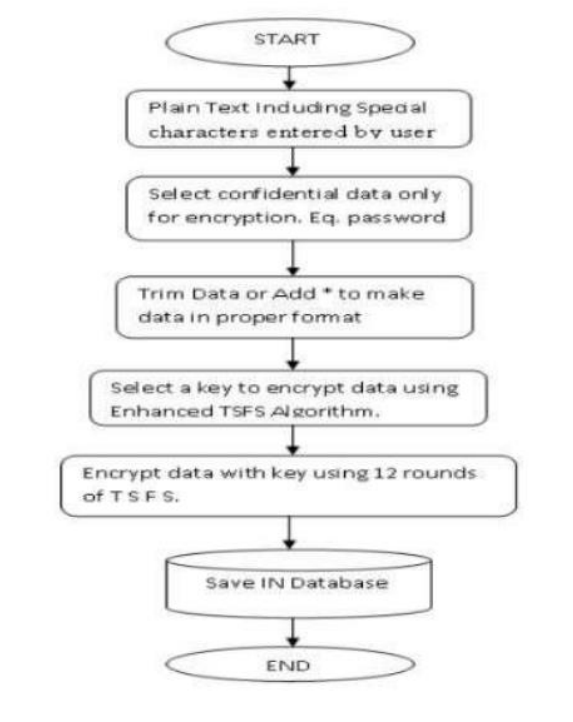
The examination section is in charge of keeping the internal and external examination schedules up to date. They also allot time slots for updating the faculty's supervision list by creating an unbiased schedule to evenly distribute the duties, and they 14 provide job benefits if faculty oversees supplementary and classroom allocation for students in the examination.

The exam department is in charge of marksheet verification and internal mark approval.

**Administrator** :

The administrator is in charge of enrolling new students and promoting them from one class to the next, from one semester to the next, and from one year to the next. Managing student accounts, such as any name, address, or other changes. The administrator is also in charge of dealing with defective accounts, such as adding new professors and allocating them to subjects. The administrator also keeps track of collegerelated information such as the schedule of events and information about any other events that take place on campus. The administrator will review all of the revisions, including student, faculty, and exam information. In the student information system, the administrator has the most power.

**3.2 System Security**



When dealing with sensitive data such as passwords, addresses, marks, and so on, the TSFS algorithm is used. It processes data using three keys, each of which is divided into 12 subkeys. The given keys are kept in a 4 X 4 matrix, therefore the length of the key must be 16 digits, and if the user supplies fewer than 16, padding is applied to the matrix. After that, we'll relocate the rows in order to conduct four operations on them: key expansion, key expansion, key expansion, key expansion, key expansion, key expansion, key expansion, key expansion, key

**3.3 Proposed System Model**

Micro-service architecture is being used to design and deploy the application. Spring-boot, an opinionated instance of spring application and a rapid application development platform, is used to build the micro-service architecture. Gathering requirements, design, development and implementations, testing, and maintenance are the five stages of the suggested technique.

**Gathering requirements**

Before beginning any project, the needs must be gathered and the viability checked. If the requirements are doable, the project can be continued. Stakeholders gather all of the requirements needed to build and implement the project during this phase, which are then communicated to the project's developer and designer.

The requirements for this project, which will culminate in a web application, are divided into six categories: Student Management Service, Course Management Service, Attendance Management Service, Administration Management Service, Document Management Service, and an Employee Management Service.

**1)Student Management Service-:** The student can use this service to check their attendance, progress report, and results, as well as send requests for any required documents, view notifications, examine timetables, and view and submit assignments. Students have the opportunity to provide comments on the teacher's performance in class.

**2)Course Management Service -:** The administrator will be able to add, amend, and delete courses using this service. The administrator will also be able to add, alter, and delete the course's subjects. Only the administrator's courses are visible to the teacher, guardian, and students.

**3)Attendance Management Service -:** Using this service, administrators will be able to submit, edit, and delete student attendance based on the course and class they are enrolled in. The attendance is only visible to the teacher, guardian, and pupils.

**4) Administration Management Service -:** The administrator will have full access to all resources in this service. The administrator can send out notifications by email, 17 SMS, and push notifications. The administrator has the ability to add, update, and delete student, guardian, and employee information.

**5) Document Management Service-:** The administrator can use this service to upload documents such as students' grades, ID proofs, subject syllabuses, payment receipts, certificates, and a variety of other papers that are necessary for the proper operation of the institution's academic and financial activities

**Technologies Used** **-:**

**Front End-:** For frontend we have used JavaScript in Front End. HTML and CSS for deigning the structure of application and forming the skeleton of our web application and also used jQuery for frontend.

Bootstrap 5 is used for making layouts.

**Back End-:** For backend we have used PHP and Apache Server

**Database:**

We have My Sql for our database.

**Our Aim:**

To develop a user-friendly interface for academic administrators, faculty, and students.

To automate routine administrative tasks such as enrollment, scheduling, and grading.

To facilitate seamless communication between stakeholders.

To provide data analytics capabilities for informed decision-making by administrators.

To ensure scalability and adaptability to different academic settings.

**Steps to Follow**:

The development of the Academic Administrative Platform followed an iterative process involving the following steps:

**Requirement Analysis:** Conducted surveys and interviews with stakeholders to identify pain points and gather requirements.

**Design:** Created wireframes and prototypes to visualize the user interface and system architecture.

**Development:** Implemented the platform using agile methodologies, incorporating feedback from stakeholders throughout the development cycle.

**Testing:** Conducted rigorous testing to ensure functionality, usability, and security.

**Deployment**: Rolled out the platform in a phased manner, providing training and support to users.

**Key Features:**

**User Authentication:** Secure login mechanisms for administrators, faculty, and students with role-based access control.

**Dashboard**: Personalized dashboards displaying relevant information such as schedules, announcements, and tasks.

**Student** **Management**: Features for enrollment, registration, and academic record management.

**Course Management**: Tools for creating, editing, and scheduling courses, along with automated generation of course catalogs.

**Grading System**: Streamlined process for entering, calculating, and publishing grades, with options for feedback and grade distribution analysis.

**Communication** **Tools**: Integrated messaging system for announcements, notifications, and correspondence between users.

**Analytics** **Dashboard**: Data visualization tools for tracking key performance indicators and trends.

**Document** **Management**: Centralized repository for storing and sharing documents such as syllabi, assignments, and policies.

**Calendar** **Integration**: Synchronization with academic calendars, events, and deadlines.

**Mobile** **Accessibility**: Responsive design for access from smartphones and tablets.



**Benefits:**

Efficiency: Automation of routine tasks reduces administrative workload and minimizes errors.

Transparency: Improved communication and access to information enhance transparency and accountability.

Productivity: Streamlined processes enable administrators, faculty, and students to focus more on teaching, learning, and research.

Data-Driven Decision Making: Analytics capabilities provide insights for strategic planning and resource allocation.

User Satisfaction: Intuitive interface and personalized experience contribute to higher user satisfaction and engagement.

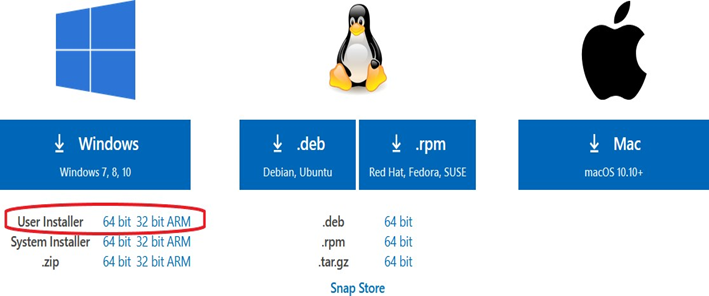


### INSTALLATION OF VS CODE:

Absolutely, installing Visual Studio Code (VS Code) is a straightforward process. Here's a step-by-step guide with images:

### Download the Installer:

* + Visit the official VS Code download page.
  + Begin by navigating to the official VS Code download page. You can do this by opening your web browser and typing in the URL: https://code.visualstudio.com/
  + Once on the download page, you'll see different versions available for various operating systems. Choose the appropriate version for your system (Windows, macOS, or Linux) by clicking on the corresponding download button.
  + After clicking the download button, the installer file will begin downloading. The file size may vary depending on your operating system and the version of VS Code you're downloading.
  + Once the download is complete, locate the installer file in your Downloads folder or wherever you chose to save it.
  + Double-click on the installer file to begin the installation process.



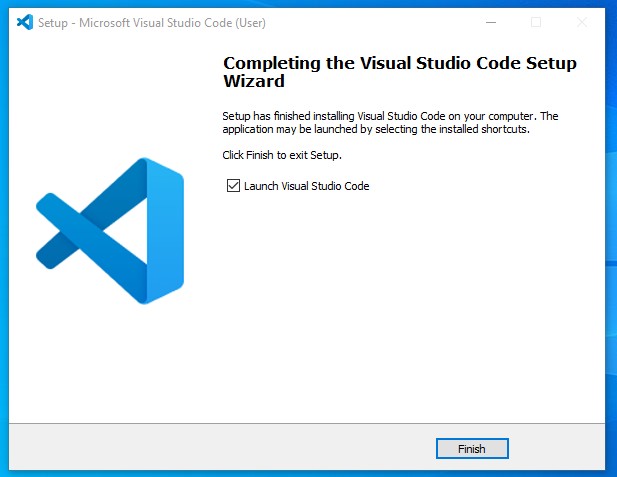
* + Select the appropriate installer for your operating system (Windows, macOS, or Linux).

### Run the Installer:

* + Double-click the downloaded installer file.

### Windows:

* + You'll see the VS Code installer window .



**For Windows**:

Download: Go to the official Visual Studio Code website: https://code.visualstudio.com/. Click on the "Download for Windows" button.

Run the Installer: Once the download is complete, locate the downloaded file (usually in your Downloads folder) and double-click on it to run the installer.

Accept License Agreement: In the installer window, you'll be presented with the license agreement. Read through it, and if you agree, click on the "I accept the agreement" checkbox, then click "Next."

Choose Destination: Choose the destination folder where you want VS Code to be installed or leave it as the default, then click "Next."

Select Additional Tasks: You may choose whether or not to add VS Code to the system PATH and create desktop and Start menu shortcuts. Make your selections and click "Next."

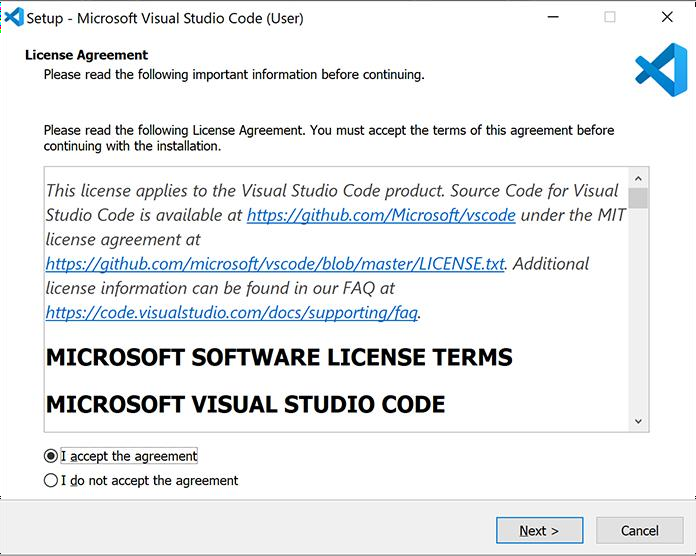
Install: Click on the "Install" button to begin the installation process.

Launch VS Code: Once the installation is complete, you can choose to launch Visual Studio Code immediately by checking the corresponding checkbox and clicking "Finish.

Click "Next" to proceed.

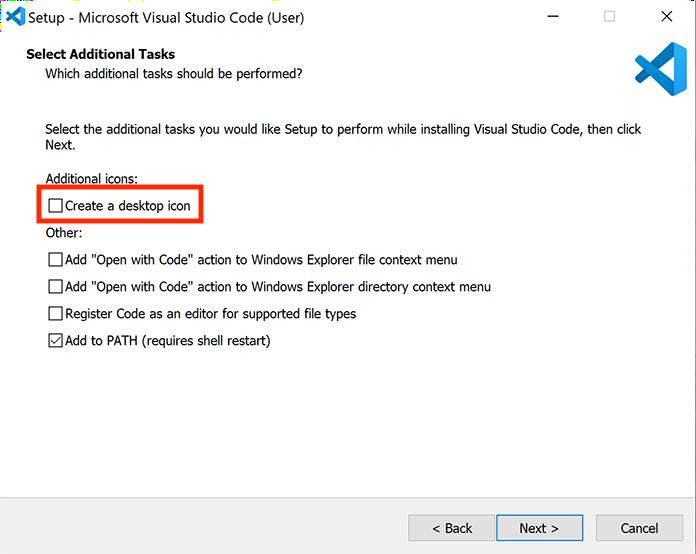
### Accept License Agreement:

* + Read through the license agreement and then click "I agree" to continue.



### Choose Installation Folder (Optional):

* + By default, VS Code will install to the suggested location. You can change this by clicking "Browse" and selecting a different folder.

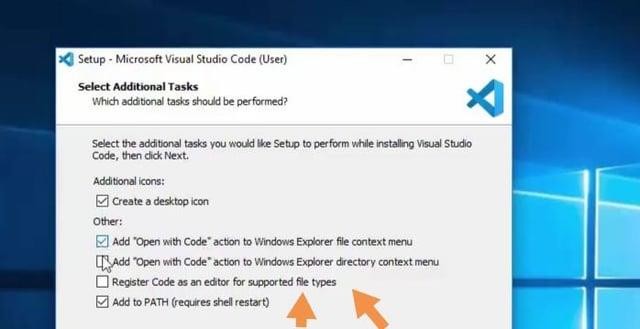


* + Click "Next" to continue.

### Choose Additional Options (Optional):

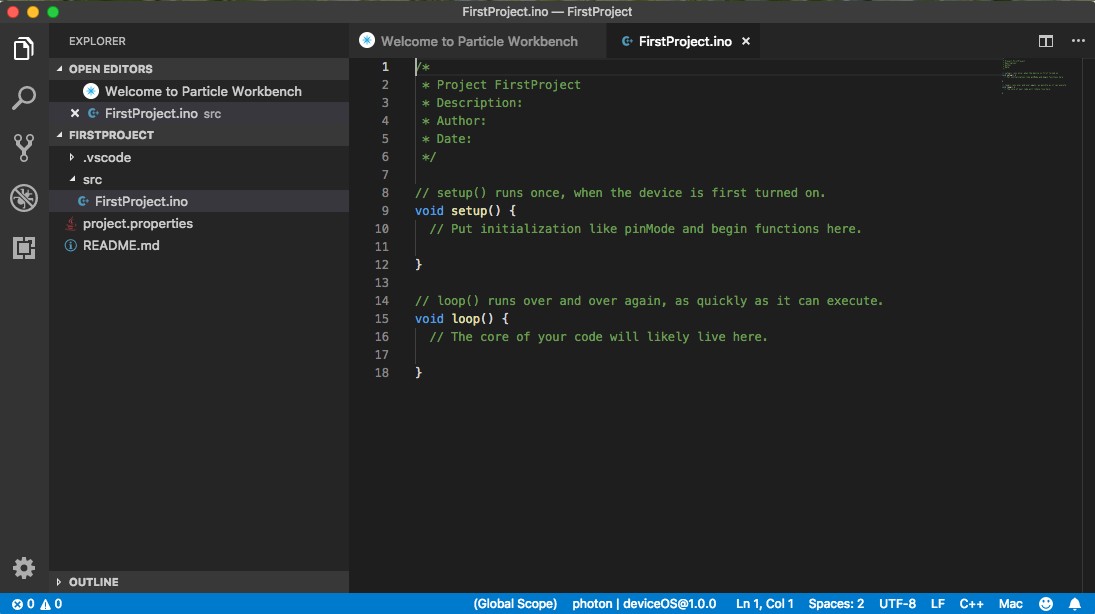
* + You can choose to create a desktop icon and/or add VS Code to your system path for easier access from the command line .





* + Click "Install" to begin the installation process.

### Launch VS Code:

* + Once the installation is complete, you'll see an option to launch VS Code directly
  + You can also find VS Code in your start menu/applications list.

### Additional Notes:

* + **Windows Specific:** Windows Specific Instructions:
  + During the installation process on Windows, you'll encounter an option to add VS Code to your system path. Selecting this option allows you to open VS Code from any directory in your command prompt or PowerShell by simply typing code.
  + This feature can be particularly convenient for developers who frequently work with command-line interfaces and need quick access to their code editor.
  + Alternative for Windows Users:
  + If you prefer not to use the installer, VS Code also offers a zip archive download option for Windows. This method doesn't require installation in the traditional sense but does involve extracting the contents of the zip file to a location of your choice.
  + Keep in mind that while this alternative method doesn't automatically update VS Code, it still provides full functionality once extracted.
  + Linux Installation:
  + Linux users may need to use their distribution's package manager for installing VS Code. Refer to the official VS Code setup guide for specific instructions tailored to your Linux distribution.
  + Following your distribution's package manager instructions ensures a smooth installation process and integration with your system environment.

**Exploring VS Code Features:**

* Now that you have VS Code installed, take some time to familiarize yourself with its features. VS Code offers a range of tools to enhance your coding experience, including IntelliSense for smart code completion, built-in Git integration for version control, and a robust extension marketplace for extending functionality.
* Spend some time exploring the user interface, trying out different themes and color schemes, and customizing keyboard shortcuts to optimize your workflow.
* Installing Extensions:
* One of the standout features of VS Code is its extensive library of extensions. These extensions provide additional functionality for various programming languages, frameworks, and tools.
* To install extensions, navigate to the Extensions view in VS Code by clicking on the square icon in the sidebar or pressing Ctrl+Shift+X. From there, you can search for extensions, read reviews, and install them with just a few clicks.
* Starting to Code:
* With everything set up, it's time to start coding! Whether you're working on a personal project, collaborating with a team, or learning a new language, VS Code provides a powerful and intuitive environment for writing code.
* Open a folder or file in VS Code, write some code, and see how VS Code's features can help you write cleaner, more efficient code in less time.
* Learning Resources:
* As you dive deeper into coding with VS Code, don't hesitate to explore the wealth of learning resources available online. From official documentation and tutorials to community forums and YouTube channels, there's no shortage of resources to help you master VS Code and improve your coding skills.
* Stay curious, keep experimenting, and never stop learning!
* Conclusion:
* Congratulations! You've successfully installed Visual Studio Code and embarked on your coding journey.
* Remember to stay engaged with the VS Code community, share your experiences, and continue exploring new features and extensions to enhance your coding experience.



## CHAPTER-4 SYSTEM ANALYSIS

### 4.1 Proposed System Features

* User friendliness is provided in the application with various controls.
* The system makes the overall project management much easier and flexible.
* It can be accessed over the internet.
* Vast amount of data can be stored.
* There is no risk of data mismanagement at any level while the project development is under process.
* Relationship between the administrator, owner/developer and subcontractor can be maintained very easily.
* It provides high level of security using different protocols like https etc.

The Student Result Processing consists of 3 users or modules, they are:

* Administrator
* Student
* Staff



**4.2 Administrator Module**

**The functionalities of Administrator are**

1. The Administrator should Login into the system with unique his/her username and password.
2. If the username and password is valid then he can gain the access to the system.
3. Admin views staff Personal details.
4. Admin updates staff Personal details.
5. Admin views student Personal details.
6. Admin updates student Personal details.
7. Admin views his/her own Personal details.
8. Admin updates his/her Personal details.
9. Admin views attendance of students.
10. Admin updates attendance of students
11. Admin views Results of students.
12. Admin updates Results of students.
13. Admin views Schedules.
14. Admin updates Schedules.
15. Admin sends emails to students once marks have been entered.



### The Administrator can do the following actions

1. Login
2. Admin Actions
   * 1. Views personal details
     2. updates schedules
     3. views reports
     4. updates attendance of students
3. Logout

**4.3 Staff Module**

**The functionalities of Staff are**

1. The Staff should login into the system with unique her/his username and password.
2. If the user name and password are valid then he can gain access to the system.
3. Staff views students Personal details.
4. Staff views his/her own Personal details.
5. Staff updates his/her Personal details.
6. Staff views attendance of students.9
7. Staff views Results of students.

**The Staff can do the following actions**

1. Login
2. Staff Actions
   1. views personal details
   2. views attendance
   3. view results
3. Log out



**4.4 Student Module**

**The functionalities of Student are**

1. The Student should login into the system with unique her/his username and password.
2. If the user name and password are valid then he can gain access to the system.
3. Student views his/her own Personal details.
4. Student updates his/her Personal details.
5. Student views attendance of students.
6. Student views Results.
7. Student views Schedules.

**The Student can do the following actions:**

1. Login

2. Student Actions

* 1. views attendance
  2. views results
  3. View Schedules

3. Log out



## 5. SYSTEM DESIGN

5.1 UML DIAGRAMS

5.1.1 Class diagram

A class diagram represents the structure of the system. It shows set of classes, interfaces, and relationships between them.



Fig. 5.1

## 5.1.2 Sequence and collaboration diagrams

**Sequence Diagram**

An interaction diagram shows an interaction, consisting of a set of objects and their relationships, including the messages that may be dispatched among them.

* A sequence diagram is an interaction diagram that emphasizes the time ordering of messages.
* Graphically, a sequence diagram is a table that shows objects arranged along x-axis and messages, ordered in increasing time, along the y-axis.



**Administrator Sequence**



Fig. 5.2



**Staff Sequence**

****

Fig. 5.3



**Student Sequence**

****

Fig. 5.4



### 

**Collaboration Diagram**

- Collaboration is a society of classes, interfaces, and other elements that work together to provide some cooperative behavior that’s bigger than the sum of all its parts.

- Collaboration is also the specification of how an element, such as a classifier or an operation, is realized by a set of classifiers and associations playing specific roles used in a specific way

**Administrator Collaboration:**

****

Fig. 5.5



## 

**Staff Collaboration **

Fig. 5.6

**Student Collaboration**



Fig. 5.7



**5.1.3 Use Case Diagram**

A use case diagram is a diagram that shows a set of use cases and actors and relationships.

**Administrator Use Case**

## 

Fig. 5.8



**Staff Use Case**

## 

**Fig. 5.9**



**Student Use Case**

****

Fig. 5.10

**5.2 CONTROL FLOW DIAGRAMS**

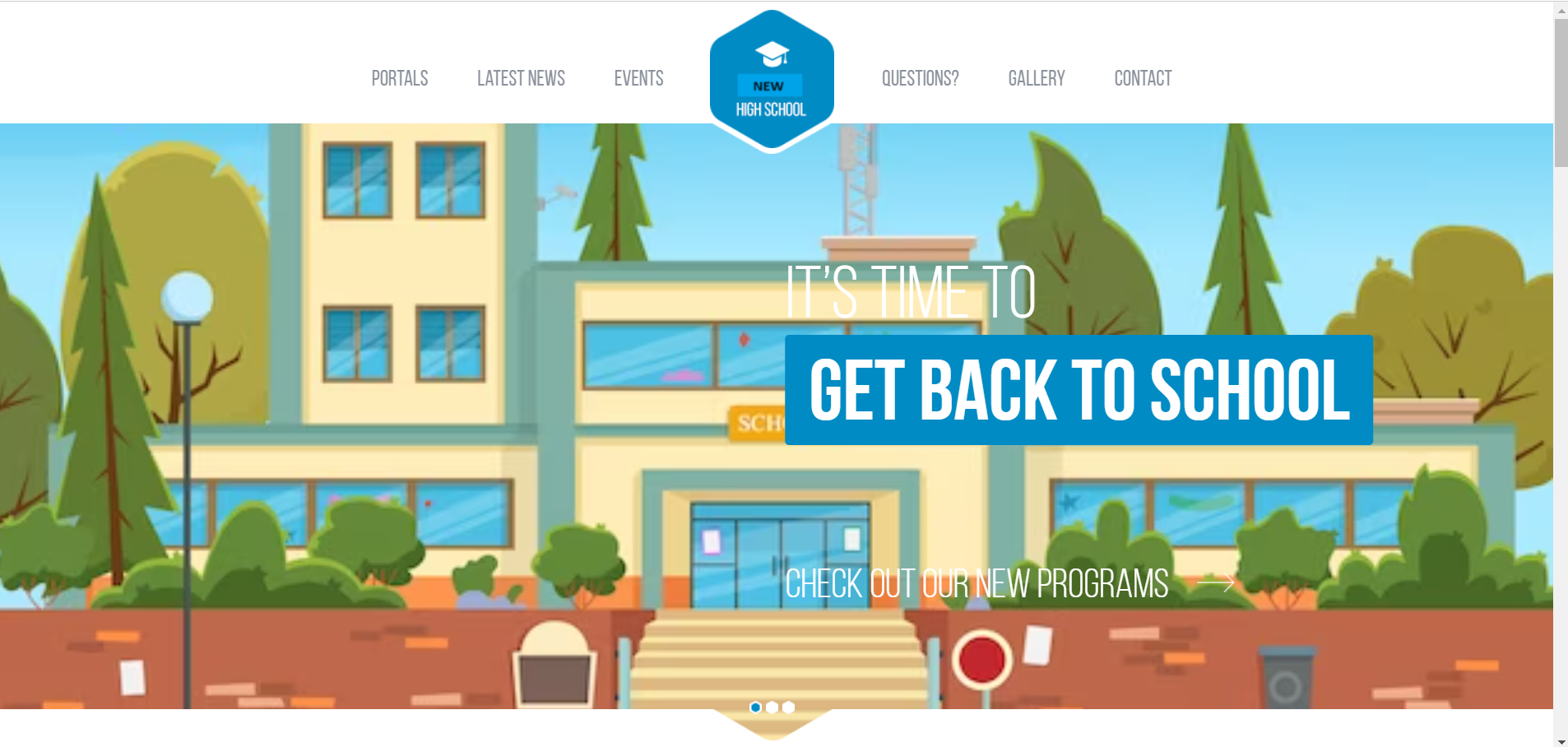
**5.2.1 Activity diagrams**

* An activity diagram shows the flow from activity to activity. An activity is an ongoing non- atomic execution within a state machine.
* Activities ultimately result in some action, which is made up of executable atomic computations that result in a change in state of the system or the return of a value.

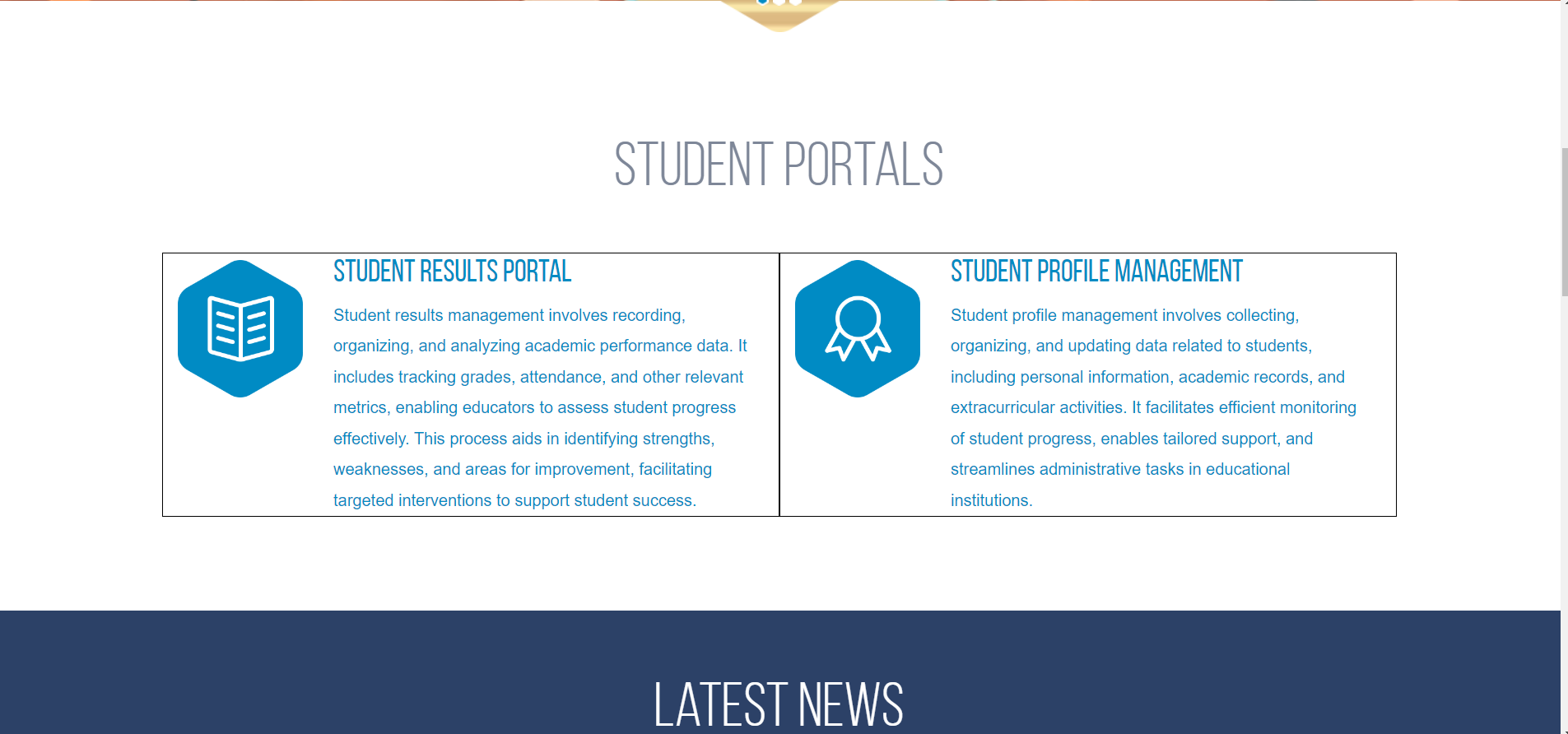


**OUTPUT SCREENS**

**HOME**

****

**STUDENT PROTALS**

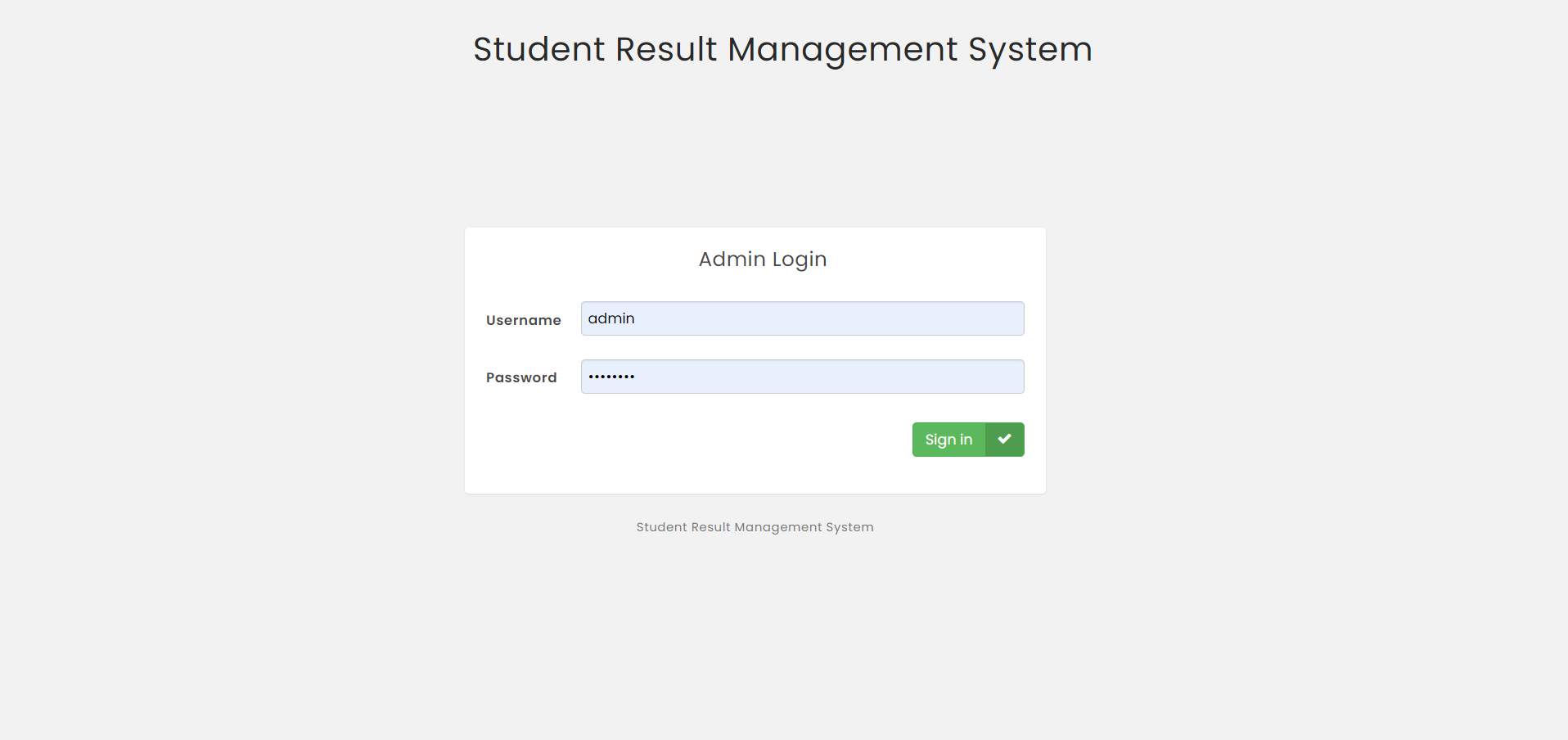




**STUDENT RESULT MANAGEMENT SYSTEMS**

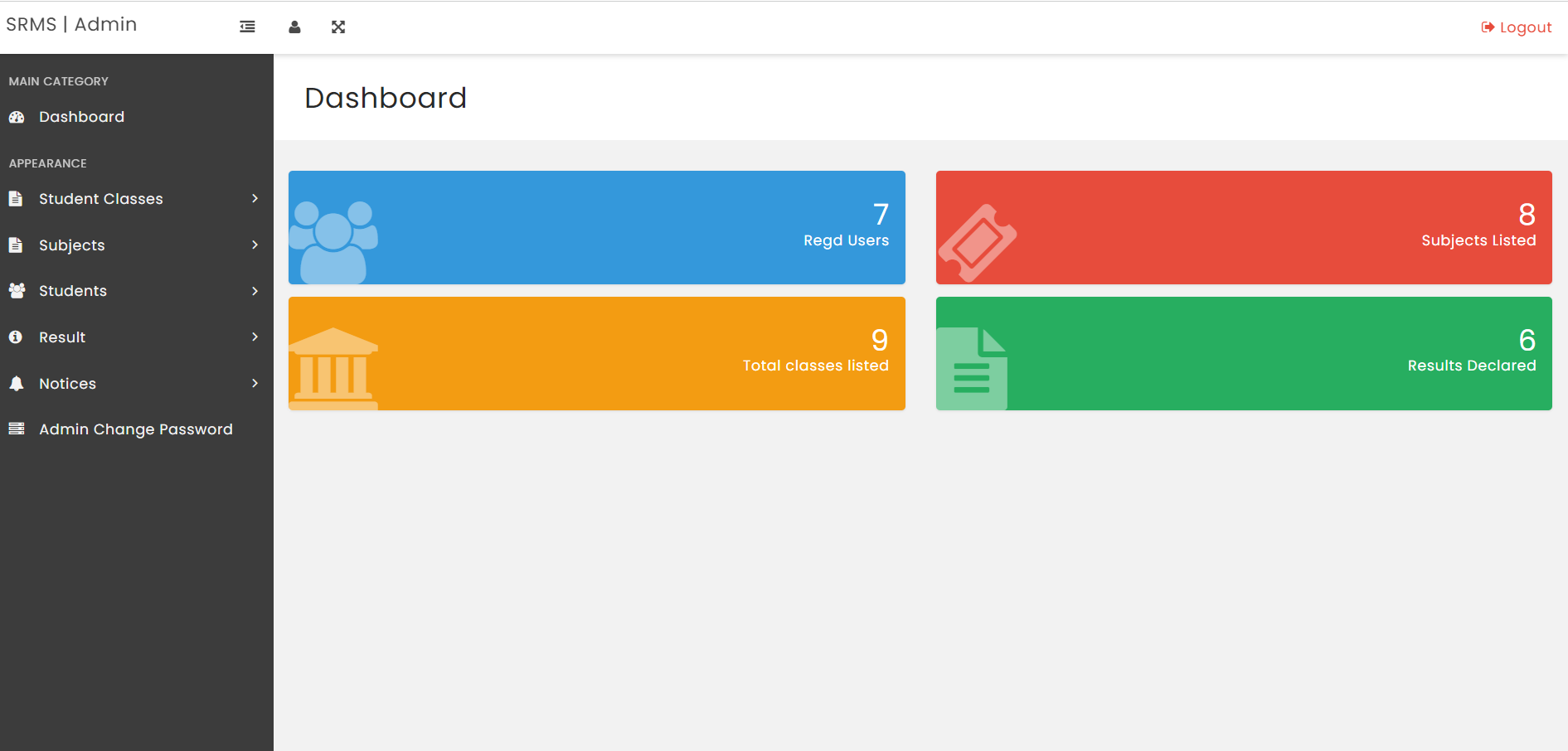


**ADMIN LOGIN**

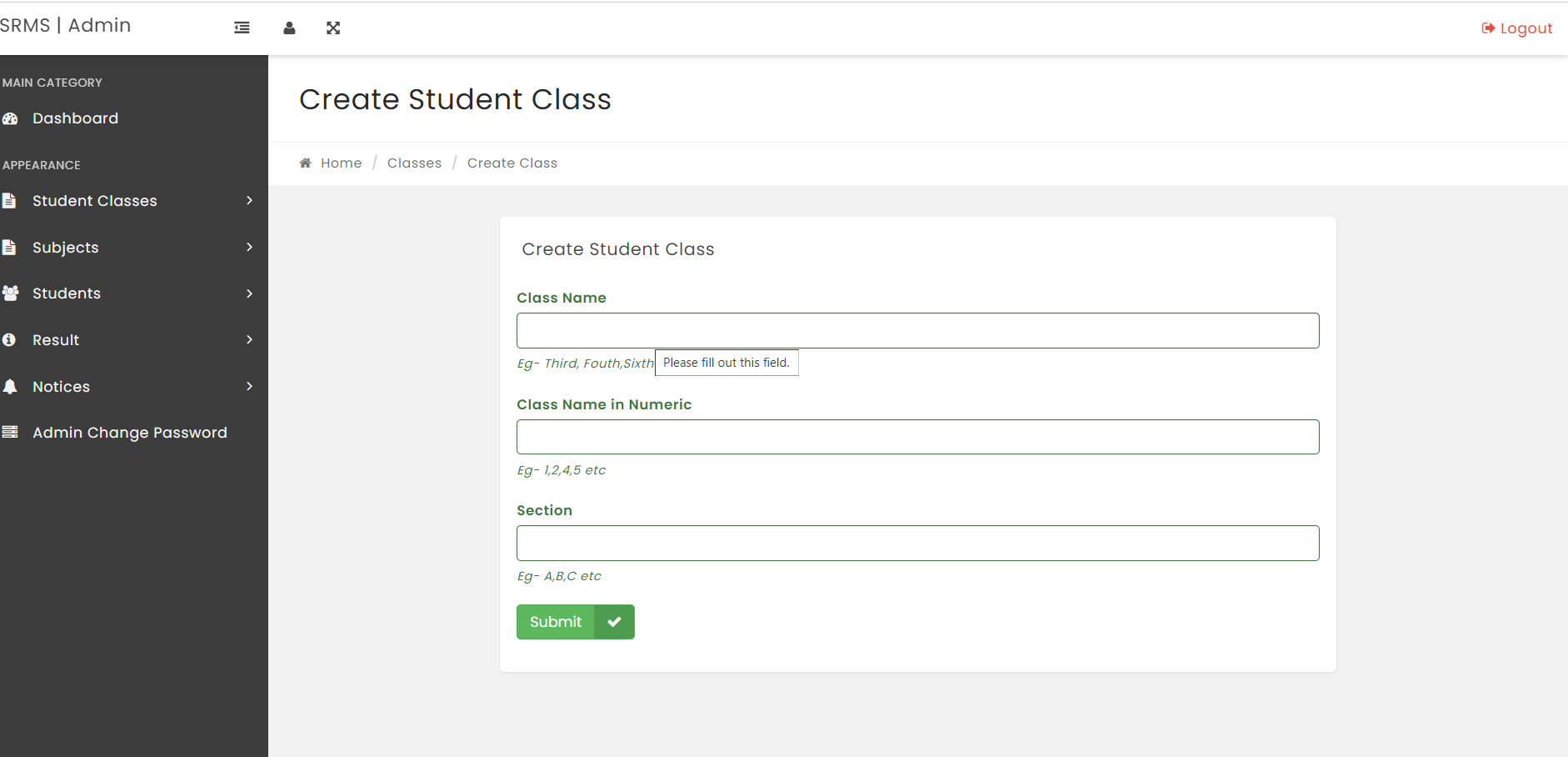




**DASHBOARD**

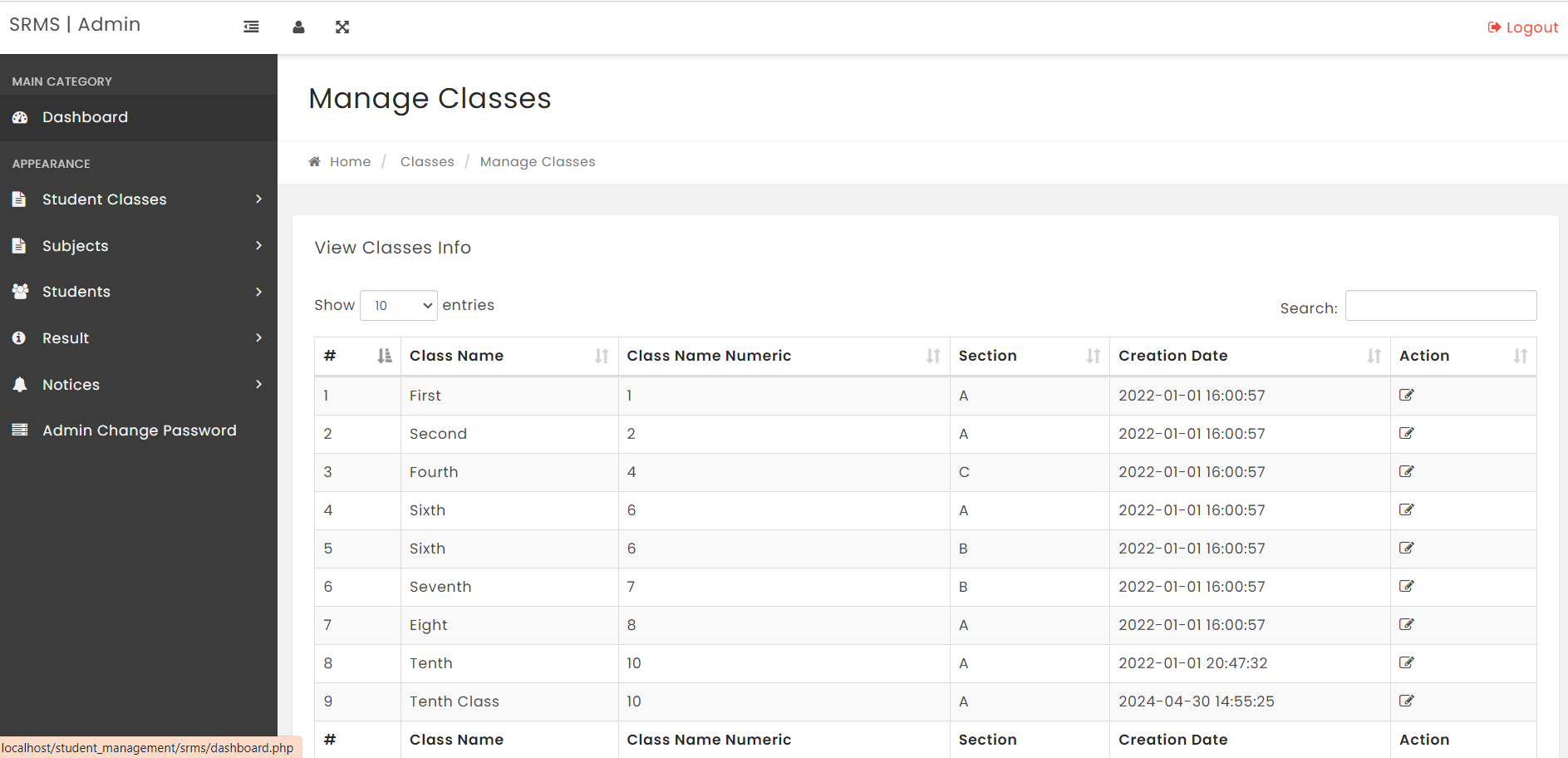


**CREATE STUDENT CLASS**

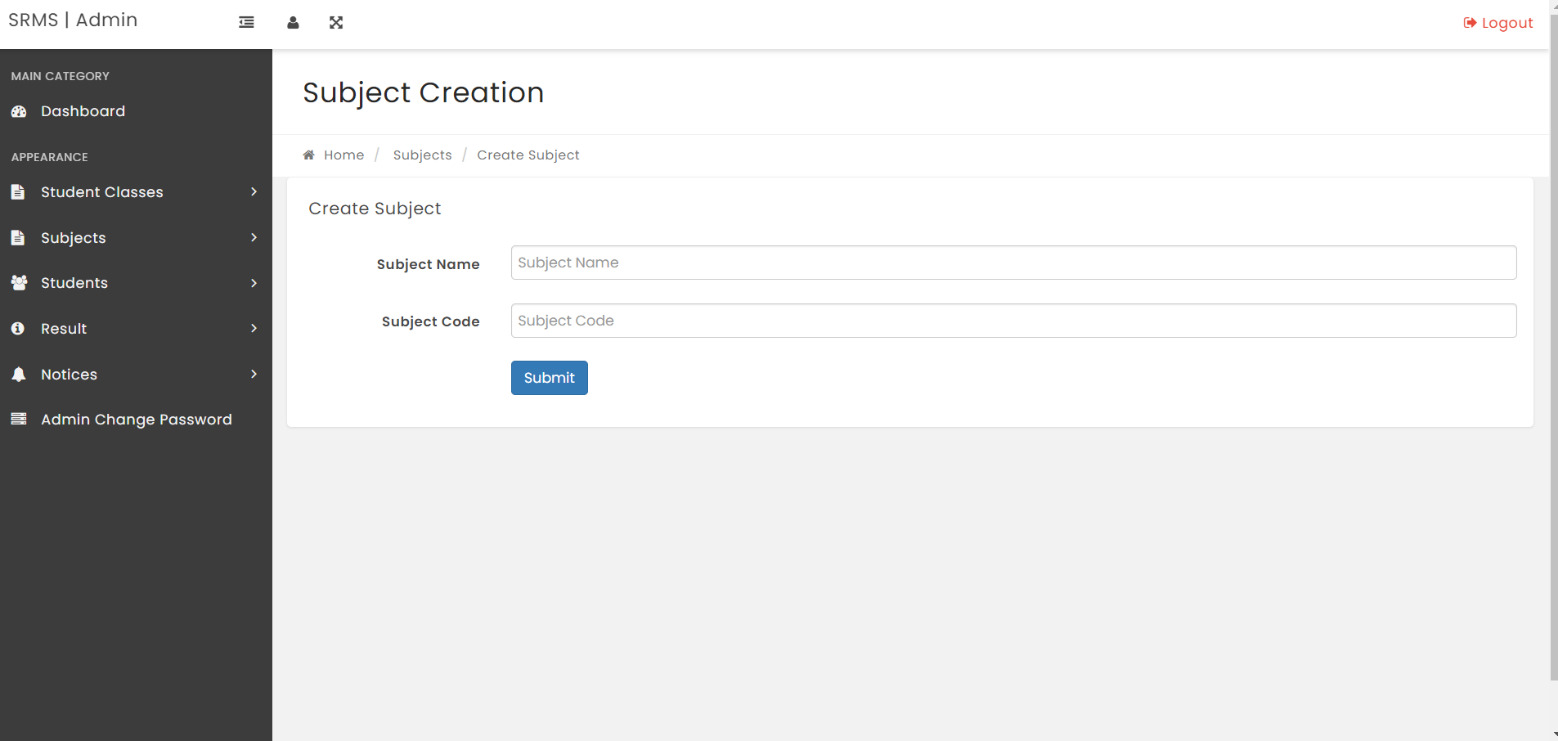




**MANAGE CLASSES**

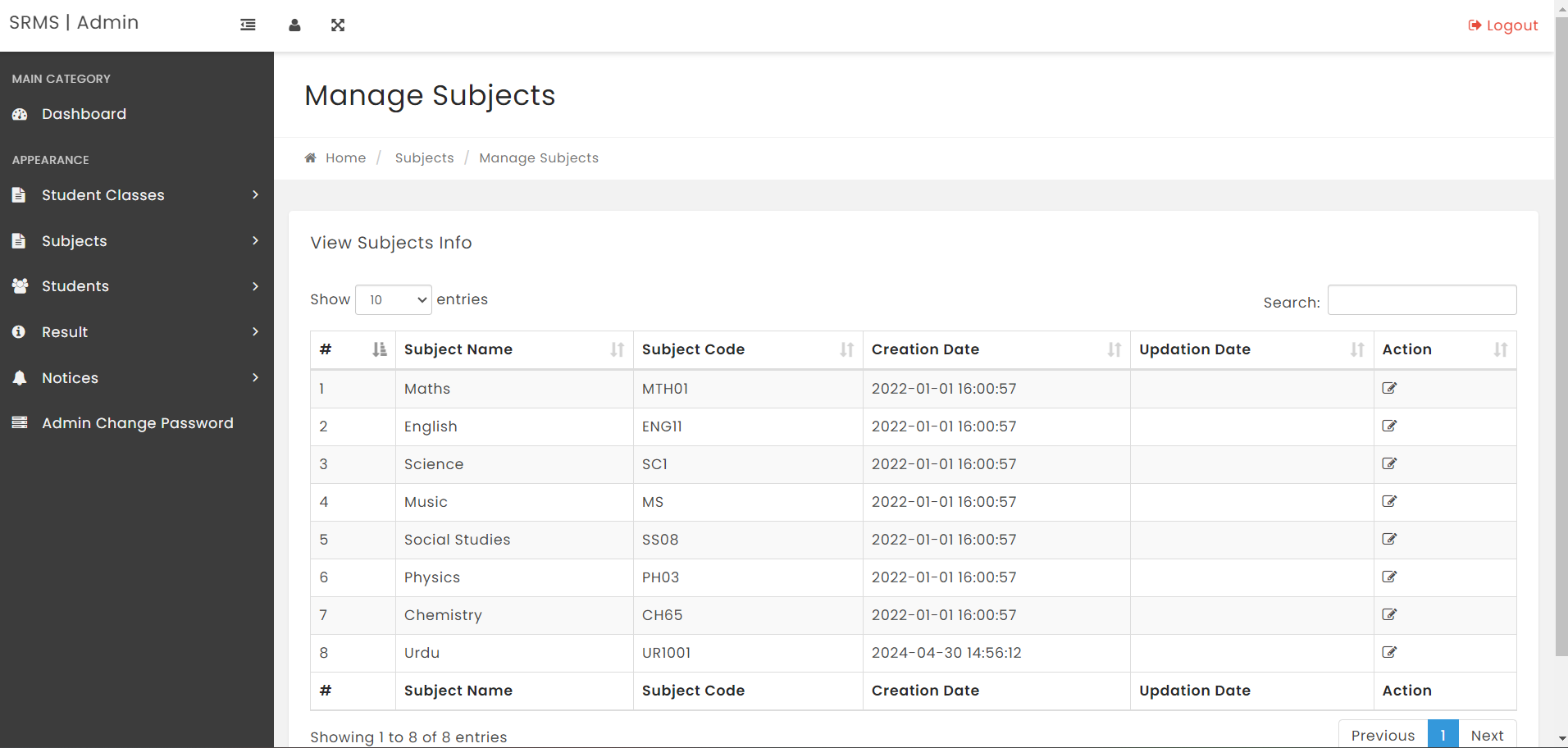


**SUBJECT CREATION**

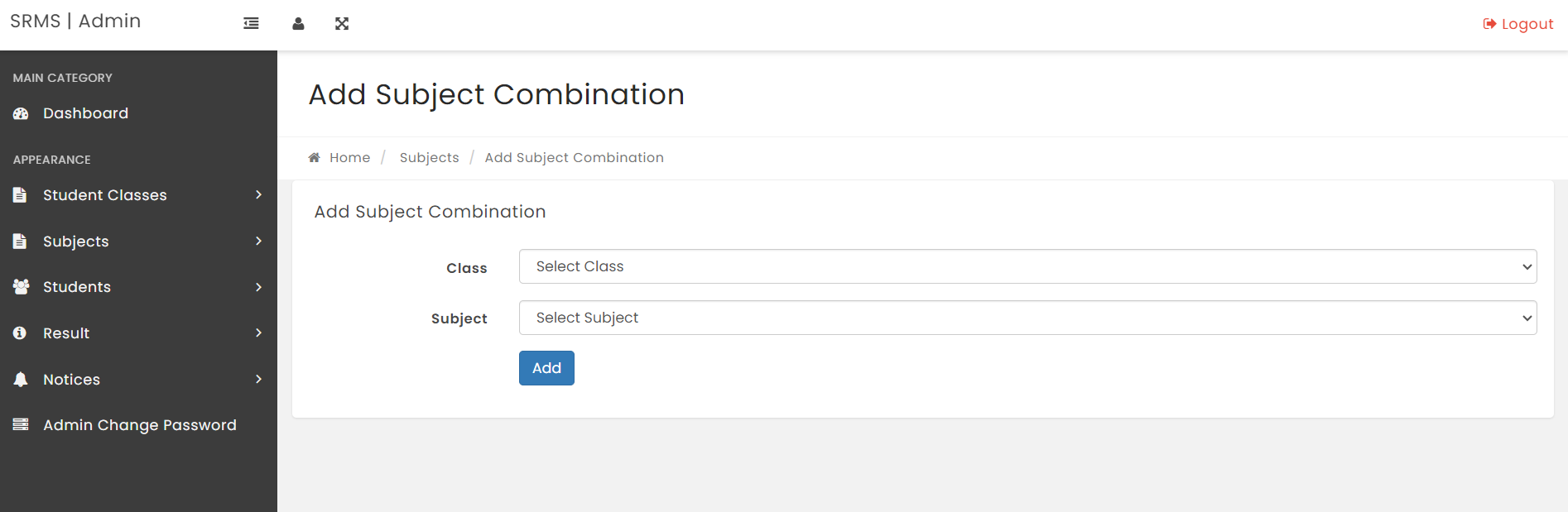




**MANAGE SUBJECTS**

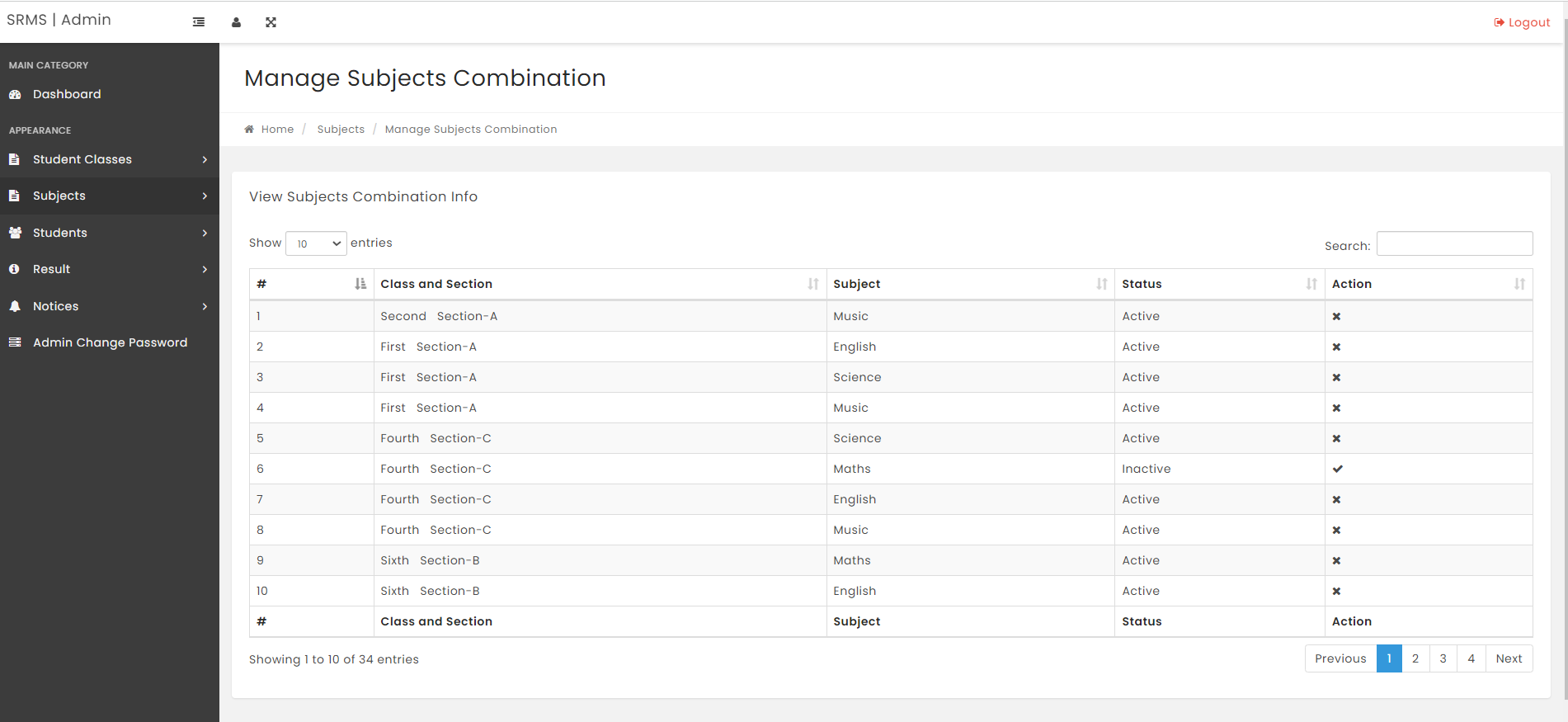


**ADD SUBJECT COMBINATION**

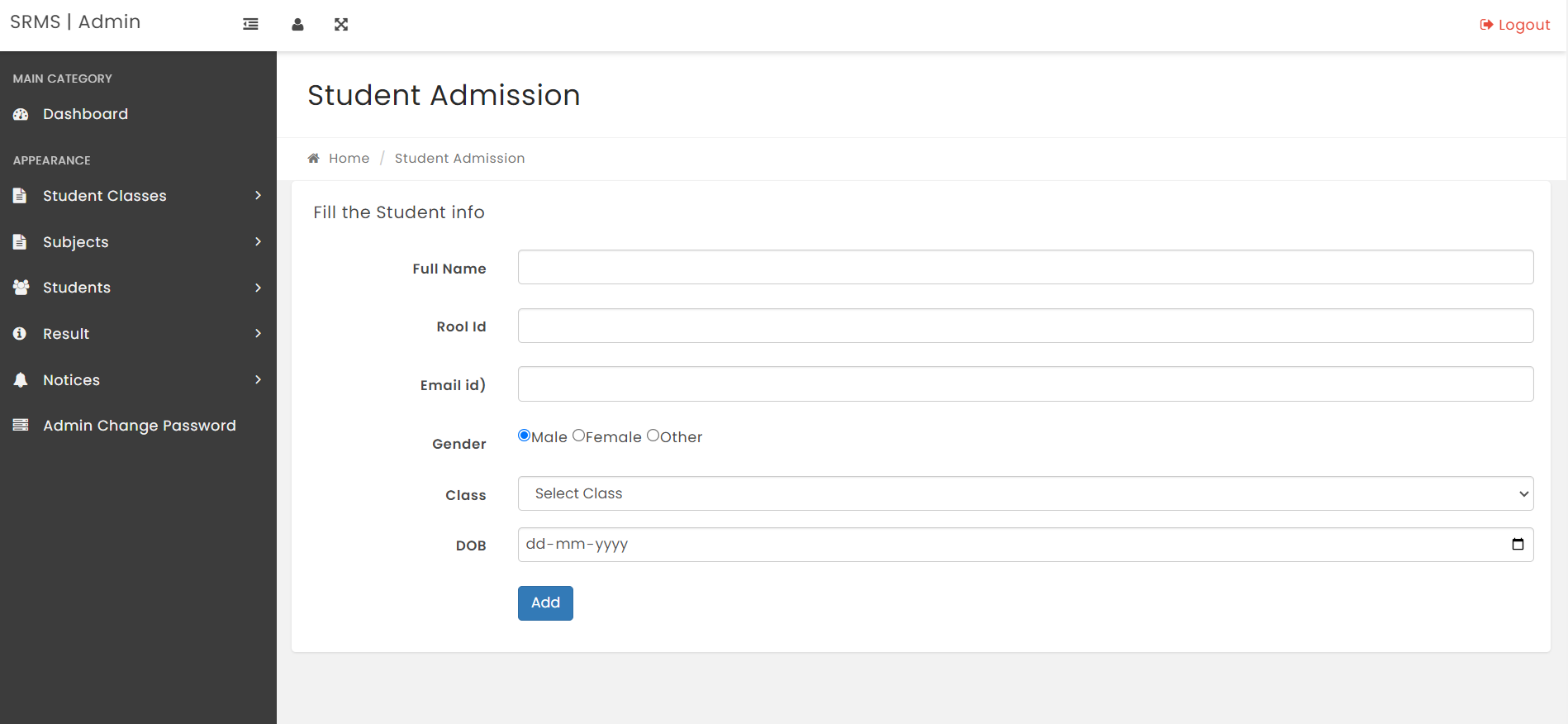




**MANAGE SUBJECT COMBINATION**

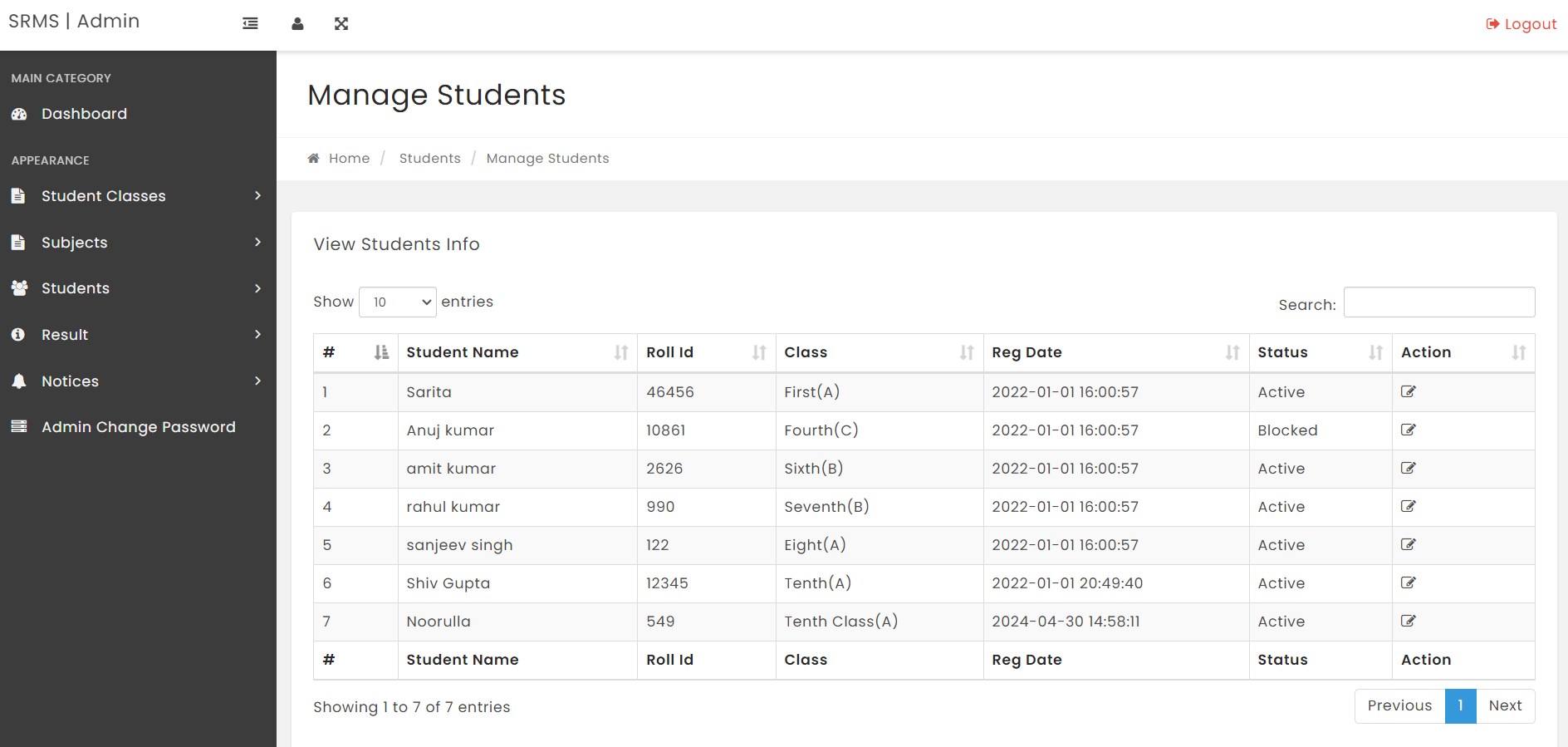


**STUDENT ADDMISSION**

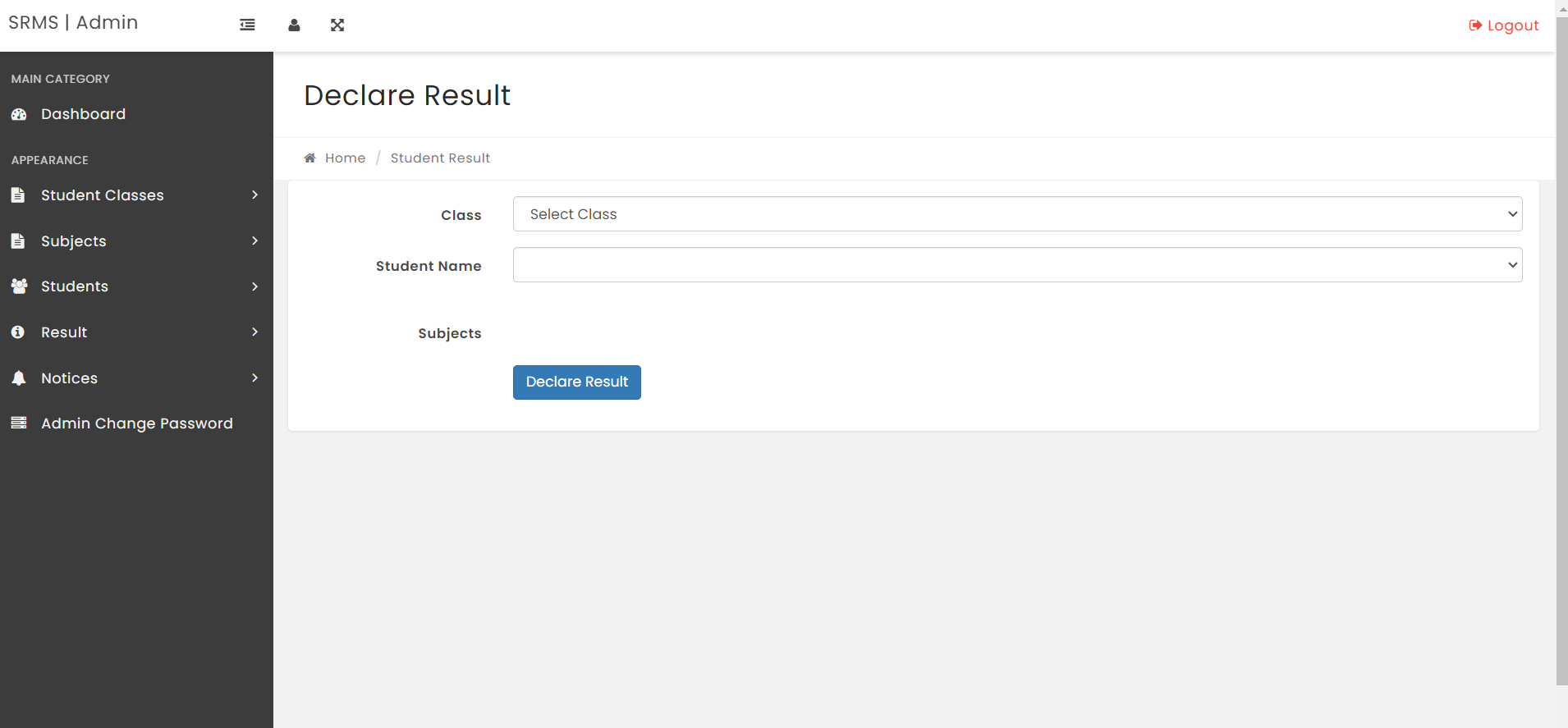




**MANAGE STUDENTS**

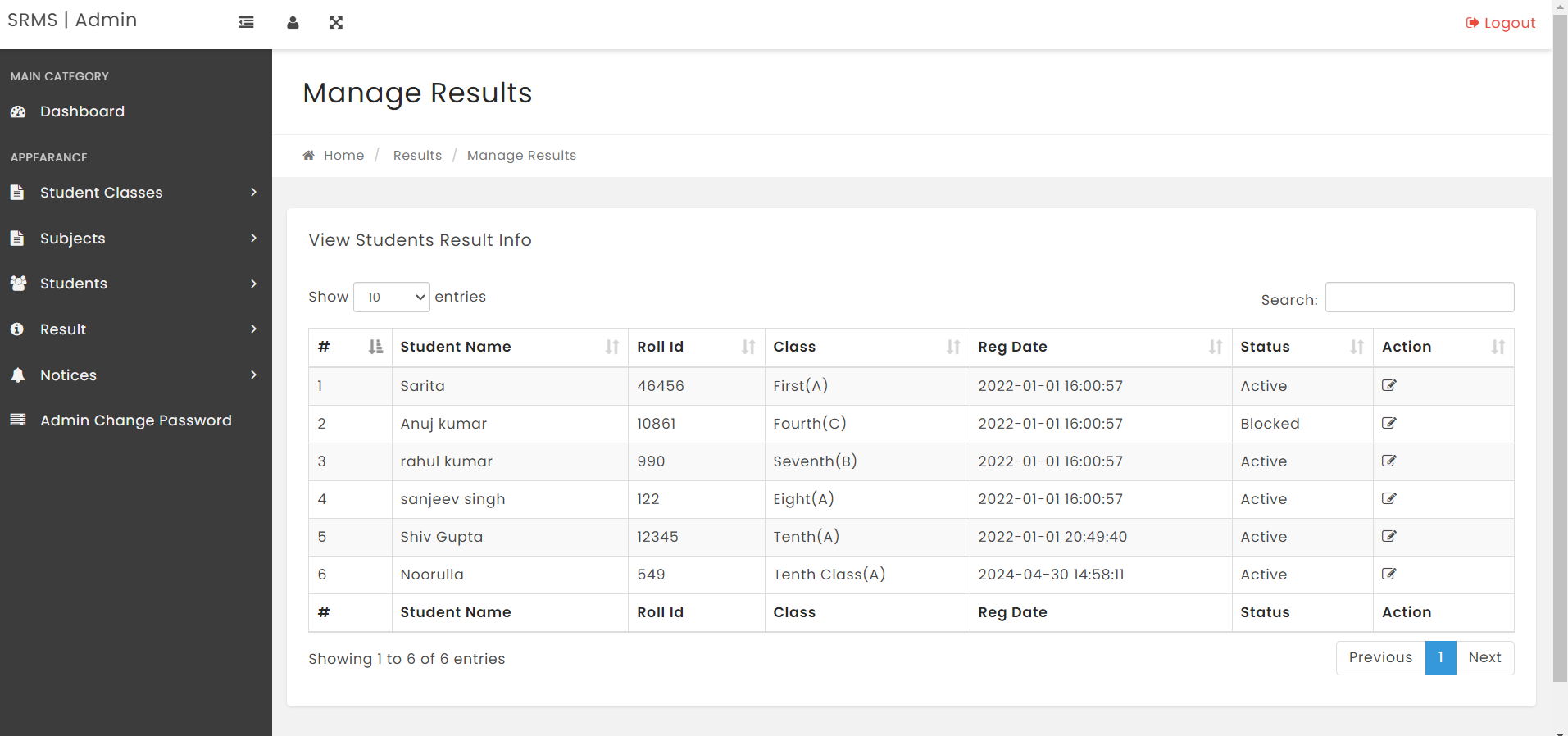


**DECLARE RESULTS**

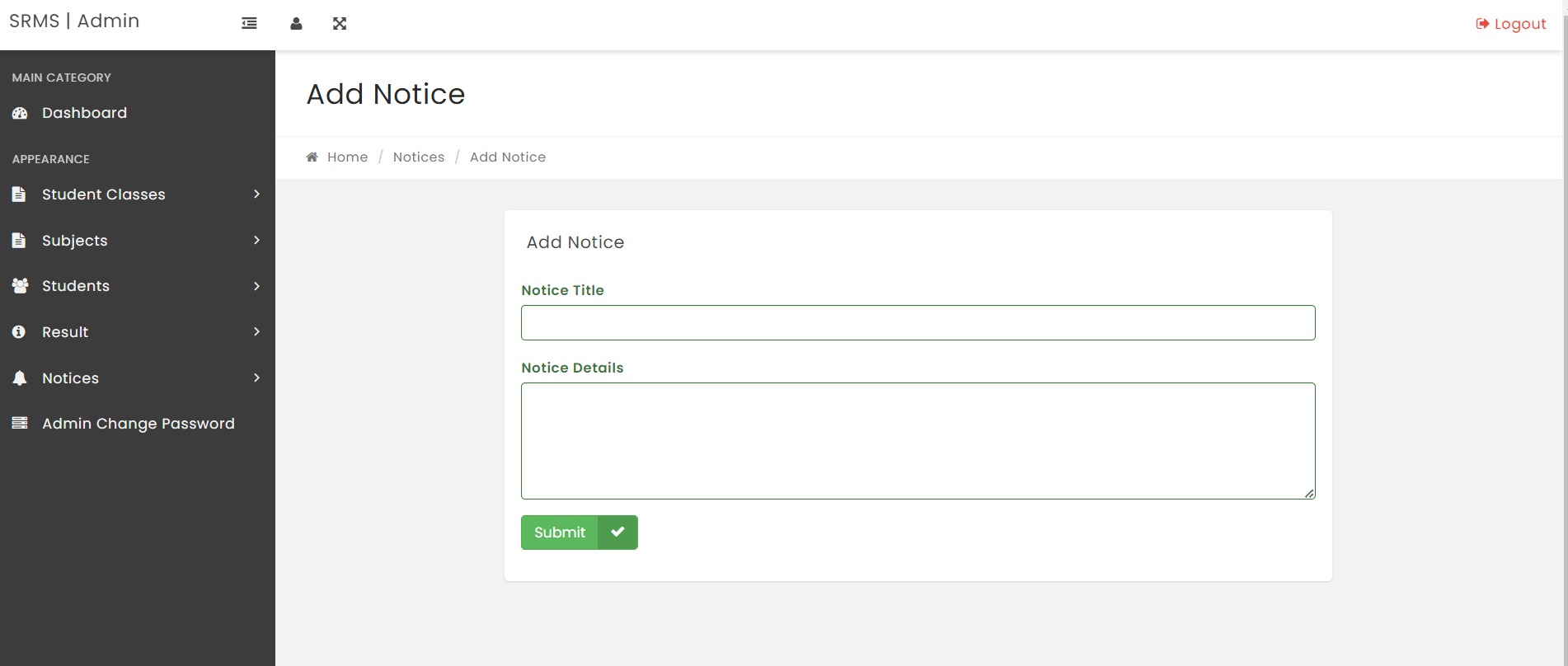
****



**MANAGE RESULTS**

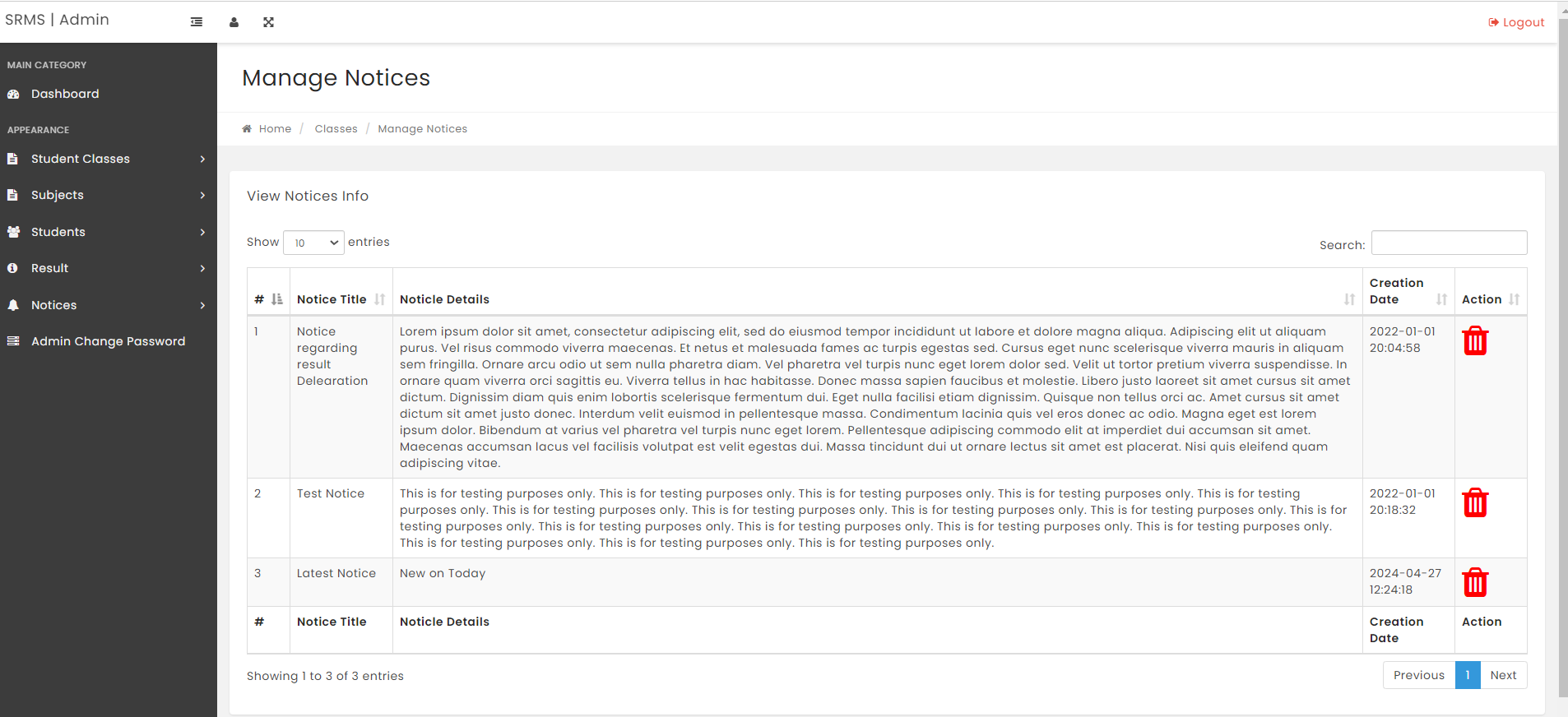


**ADD NOTICE**





**MANAGE NOTICE**





### CHAPTER 6

### IMPLEMENTATION

The School Management System will be implemented using a combination of front-end and back-end technologies to ensure a robust and user-friendly application. Here are the technologies that will be utilized:

**6.1 HTML (Hypertext Markup Language):**

HTML will be used for structuring the web pages of the School Management System, providing the foundation for content presentation and layout.

**6.2 CSS (Cascading Style Sheets):**

CSS will be employed for styling the HTML elements, ensuring consistency in design, and enhancing the visual appeal of the user interface.

**6.3 JavaScript:**

JavaScript will be utilized for client-side scripting to enhance the interactivity and functionality of the School Management System. It will be used for tasks such as form validation, dynamic content updates, and asynchronous communication with the server.

**6.4 jQuery:**

jQuery, a JavaScript library, will be leveraged to simplify DOM manipulation, event handling, and AJAX requests, thus speeding up the development process and improving cross-browser compatibility.

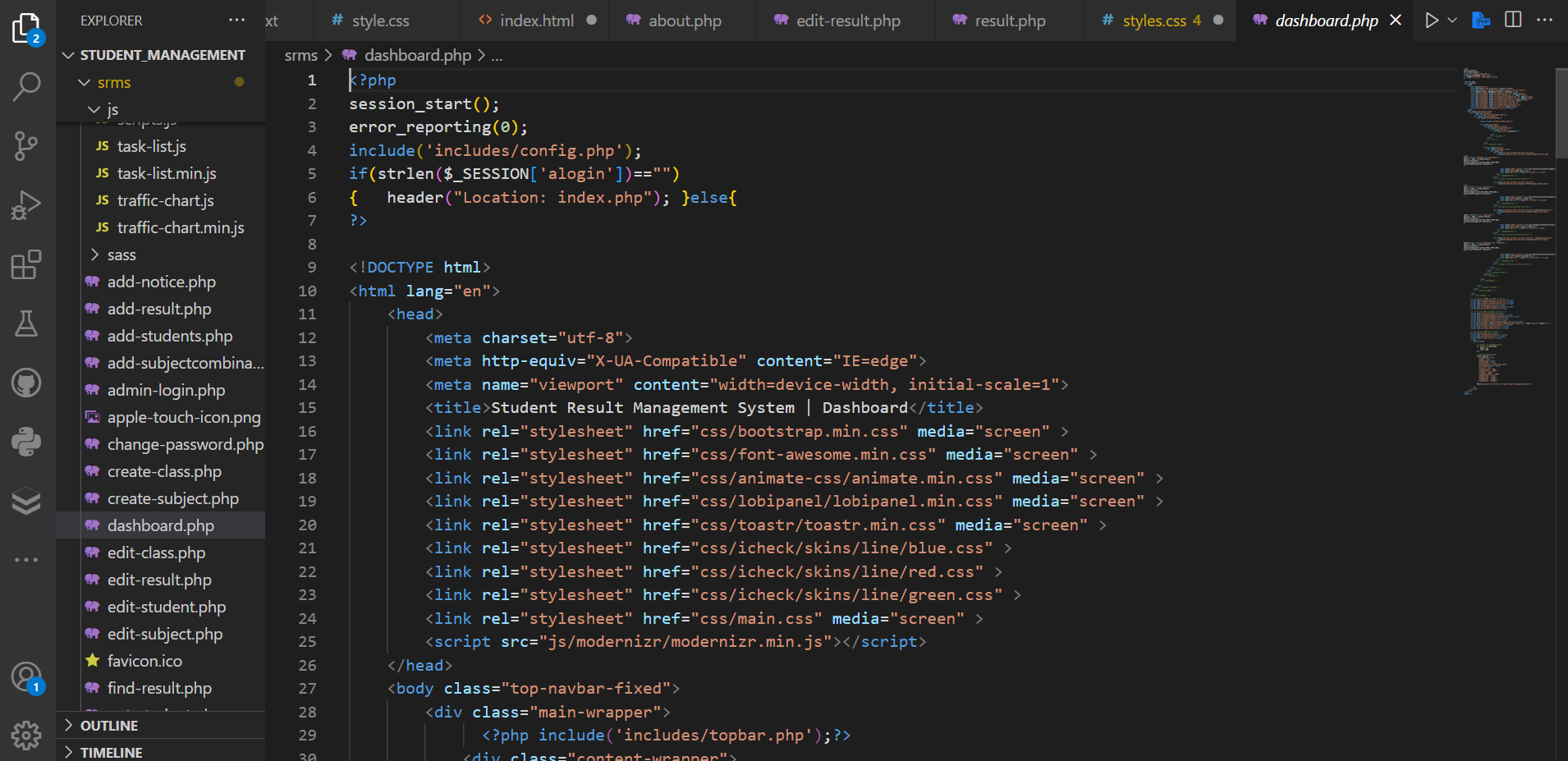
**6.5 XAMPP Server:**

XAMPP (Cross-platform, Apache, MySQL, PHP, Perl) server will be used to create a local development environment for testing and debugging the School Management System. XAMPP includes Apache HTTP Server, MySQL database, and PHP, providing a complete stack for server-side scripting and database management.

**6.6 PHP (Hypertext Preprocessor):**

PHP will serve as the primary server-side scripting language for the School Management System. It will handle dynamic content generation, database interactions.







**CHAPTER – 7**

**SYSTEM STUDY**

**7.1 Introduction**: The Academic Administrative Platform, also known as the School Management System, is a comprehensive software solution designed to streamline and automate various administrative tasks within educational institutions. This system study aims to analyze the current processes, gather requirements, and define the scope for the development of the School Management System.

Current System Analysis: The current academic administrative processes involve manual handling of tasks such as student enrollment, course registration, grading, scheduling, and student records management. These processes often suffer from inefficiencies, errors, and delays, leading to challenges in managing administrative workload effectively.

**7.2 Requirements Gathering**: Through interviews, surveys, and workshops with stakeholders including administrators, faculty, students, and support staff, we aim to elicit their requirements and preferences for the new School Management System. This includes defining functional requirements such as user roles, data management, reporting capabilities, and non-functional requirements such as performance, security, and usability.

**7.3 Use Case Analysis**: Use cases will be developed to model the interactions between users and the School Management System. These use cases will capture common tasks and scenarios such as student registration, course enrollment, grading, and generating reports to ensure that the system meets the needs of its users effectively.

**7.4 Data Modeling**: An entity-relationship diagram (ERD) will be created to represent the data entities and their relationships within the academic administrative domain. This will include defining attributes for each entity, specifying data types, constraints, and relationships to ensure the integrity and consistency of the data.



**7.6 Technology Stack Selection**: A careful evaluation of available technologies and frameworks will be conducted to select the most suitable technology stack for building the School Management System. This will include decisions regarding programming languages, databases, web frameworks, and third-party APIs based on project requirements and stakeholder preferences

**7.7 Prototype Development**: A prototype or proof-of-concept will be developed to validate key features and functionality of the School Management System. Stakeholder feedback will be solicited to iteratively refine the prototype and ensure alignment with user needs and expectations.

**7.8 Risk Analysis**: Potential risks and challenges associated with the development and implementation of the School Management System will be identified and analyzed. Risk mitigation strategies will be developed to address these risks and minimize their impact on project delivery and success.

**7.9 Conclusion**: This system study provides a comprehensive overview of the process for developing the Academic Administrative Platform or School Management System. By understanding the current system, gathering requirements, and analyzing risks, we aim to lay a solid foundation for the successful development and implementation of the system, ultimately enhancing the efficiency and effectiveness of administrative processes within educational institutions.



### CHAPTER – 8

### SYSTEM TESTING

### 8.1 UNIT TESTING:

Unit testing focuses on testing individual components or modules of the School Management System in isolation.

Developers write test cases to verify the correctness of each unit, ensuring that it behaves as expected and meets its specifications.

Tools like PHP Unit for PHP or Jest for JavaScript can be used to automate unit testing processes.

**8.2 Integration Testing:**

Integration testing evaluates the interactions between different modules or components of the School Management System.

Test cases are designed to verify that modules communicate correctly, data flows smoothly between them, and interfaces function as intended.

Techniques such as top-down, bottom-up, and incremental integration testing can be employed to gradually integrate and test system components.

**8.3 System Testing:**

System testing assesses the entire School Management System as a cohesive unit.

It validates the system against its functional and non-functional requirements, including user interface, functionality, performance, security, and compatibility.

Test cases cover various scenarios, including positive and negative cases, boundary conditions, and error handling.

**8.4 Acceptance Testing:**

Acceptance testing involves evaluating the School Management System's compliance with user requirements and acceptance criteria.

Stakeholders, including administrators, faculty, students, and support staff, participate in acceptance testing to ensure the system meets their needs and expectations.

It may include alpha testing, beta testing, and user acceptance testing (UAT) conducted in a real or simulated environment.

**8.5 Regression Testing:**

Regression testing ensures that changes or updates to the School Management System do not adversely affect existing functionality.



It involves re-running previously executed test cases to verify that no regression issues occur after modifications are made.

Automated testing tools and techniques help streamline regression testing processes, ensuring efficient identification and resolution of any regressions.

**8.6 Performance Testing:**

Performance testing evaluates the School Management System's responsiveness, scalability, and reliability under various workload conditions.

It includes load testing to assess system performance under expected and peak loads, stress testing to determine system breaking points, and scalability testing to measure system capacity.

**8.7 Security Testing:**

Security testing identifies and addresses vulnerabilities and risks related to data security, access control, and protection against cyber threats.

Techniques such as penetration testing, vulnerability scanning, and security audits help identify security weaknesses and ensure compliance with security standards and regulations.



### Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

### Test objectives

 All field entries must work properly.

 Pages must be activated from the identified link.

 The entry screen, messages and responses must not be delayed.

### Features to be tested

 Verify that the entries are of the correct format

 No duplicate entries should be allowed

 All links should take the user to the correct page.

### INTEGRATION TESTING:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level

– interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

### ACCEPTANCE TESTING:

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

### Test Results: All the test cases mentioned above passed successfully. No defects encountered.



**TEST CASES FOR ADMIN LOGIN:**

Valid Credentials:

Test ID: **TC\_ADMIN\_LOGIN\_001**

Description: Verify that the admin can successfully log in with valid username and password.

Test Steps:

Navigate to the admin login page.

Enter a valid username and password.

Click on the "Login" button.

Expected Result: The admin should be logged in successfully and redirected to the admin dashboard.

Invalid Username:

Test ID: **TC\_ADMIN\_LOGIN\_002**

Description: Verify that the admin cannot log in with an invalid username.

Test Steps:

Navigate to the admin login page.

Enter an invalid username and a valid password.

Click on the "Login" button.

Expected Result: The system should display an error message indicating that the username is incorrect.

Invalid Password:

Test ID: **TC\_ADMIN\_LOGIN\_003**

Description: Verify that the admin cannot log in with an invalid password.

Test Steps:

Navigate to the admin login page.

Enter a valid username and an invalid password.

Click on the "Login" button.

Expected Result: The system should display an error message indicating that the password is incorrect.

Empty Username:

Test ID: **TC\_ADMIN\_LOGIN\_004**

Description: Verify that the admin cannot log in with an empty username field.

Navigate to the admin login page.

Click on the "Login" button.



Expected Result: The system should display an error message indicating that the username field is required.

Empty Password:

Test ID: **TC\_ADMIN\_LOGIN\_005**

Description: Verify that the admin cannot log in with an empty password field.

Test Steps:

Navigate to the admin login page.

Enter a valid username and leave the password field empty.

Click on the "Login" button.

Expected Result: The system should display an error message indicating that the password field is required.

Locked Account:

Test ID: **TC\_ADMIN\_LOGIN\_006**

Description: Verify that the admin cannot log in with a locked account.

Test Steps:

Navigate to the admin login page.

Enter the username and password of a locked admin account.

Click on the "Login" button.

Expected Result: The system should display an error message indicating that the account is locked.

Logout Functionality:

Test ID: **TC\_ADMIN\_LOGIN\_007**

Description: Verify that the admin can log out successfully from the system.

Test Steps:

Log in to the system as an admin.

Navigate to the logout option.

Click on the logout option.

Expected Result: The admin should be logged out successfully and redirected to the login page.groups



**TEST CASES FOR STUDENT MODULE:**

Student Registration:

Test ID: **TC\_STUDENT\_MODULE\_001**

Description: Verify that a new student can register successfully.

Test Steps:

Navigate to the student registration page.

Fill in the required fields with valid student information (e.g., name, email, address, date of birth).

Click on the "Register" button.

Expected Result: The student should be registered successfully, and a confirmation message should be displayed.

Duplicate Email Check:

Test ID: **TC\_STUDENT\_MODULE\_002**

Description: Verify that the system checks for duplicate emails during student registration.

Test Steps:

Attempt to register a student with an email address that is already registered in the system.

Fill in the remaining required fields with valid information.

Click on the "Register" button.

Expected Result: The system should display an error message indicating that the email address is already in use.

View Student Profile:

Test ID: **TC\_STUDENT\_MODULE\_003**

Description: Verify that a student can view their profile information after logging in.

Test Steps:

Log in to the system as a student.

Navigate to the student profile page.

Expected Result: The student should be able to view their profile information, including their name, email, address, and other details.

Edit Student Profile:

Test ID: **TC\_STUDENT\_MODULE\_004**

Description: Verify that a student can edit their profile information.



Test Steps:

Log in to the system as a student.

Navigate to the student profile page.

Click on the "Edit Profile" button.

Update the relevant fields with new information.

Click on the "Save Changes" button.

Expected Result: The student's profile information should be updated successfully, and a confirmation message should be displayed.

View Course Schedule:

Test ID: **TC\_STUDENT\_MODULE\_005**

Description: Verify that a student can view their course schedule.

Test Steps:

Log in to the system as a student.

Navigate to the course schedule page.

Expected Result: The student should be able to view their course schedule, including the courses they are enrolled in, their timings, and instructors.

Enroll in Course:

Test ID: **TC\_STUDENT\_MODULE\_006**

Description: Verify that a student can enroll in a course.

Test Steps:

Log in to the system as a student.

Navigate to the course catalog or enrollment page.

Select a course from the list of available courses.

Click on the "Enroll" button.

View Grades:

Test ID: **TC\_STUDENT\_MODULE\_007**

Description: Verify that a student can view their grades for enrolled courses.

Test Steps:

Log in to the system as a student.

Navigate to the grades or transcript page.

Expected Result: The student should be able to view their grades for each enrolled course, includes.



**CONCLUSION**

In conclusion, the Academic Administrative Platform, meticulously crafted through systematic study, implementation, and rigorous testing, stands poised to revolutionize administrative efficiency and communication within educational institutions. Leveraging cutting-edge technologies and best practices, it promises to streamline processes, elevate data accuracy, and deliver a seamless experience for students, faculty, administrators, and support staff alike. With its successful deployment, the platform is set to emerge as an indispensable asset in the education sector, empowering institutions to make informed decisions and achieve operational excellence.



**FUTURE IMPROVEMENTS**

**Enhanced User Experience (UX):** Continuously refine and optimize the user interface to ensure intuitive navigation and seamless interaction for all stakeholders.

**Advanced Analytics**: Integrate advanced analytics tools to provide insights into student performance, enrollment trends, and administrative efficiency, enabling data-driven decision-making.

**Mobile Accessibility**: Develop dedicated mobile applications or responsive design to facilitate access to the platform from smartphones and tablets, catering to the evolving needs of users on the go.

**Artificial Intelligence (AI) Integration**: Explore AI-powered features such as chatbots for student support, automated grading assistance, and predictive analytics for personalized learning experiences.

**Enhanced Security Measures**: Implement robust cybersecurity measures, including encryption, multi-factor authentication, and regular security audits, to safeguard sensitive student and institutional data.

**Integration with Learning Management Systems (LMS):** Seamlessly integrate with popular LMS platforms to facilitate course management, content delivery, and assessment coordination for a holistic learning experience.

**Scalability and Performance Optimization**: Continuously optimize system architecture and infrastructure to accommodate growing user bases and ensure optimal performance under increasing workload demands.

**Feedback Mechanisms**: Implement feedback mechanisms to gather input from users regularly, enabling continuous improvement and refinement based on user needs and preferences.

Enhanced Collaboration Tools: Introduce collaborative features such as group project management, virtual classrooms, and real-time document sharing to foster collaboration and engagement among students and faculty.



**REFERENCE**

**Articles and Whitepapers:**

**My School: School Management System Based on Web** [**https://www.studocu.com/en-gb/document/london-metropolitan-university/knowledge-in-policing/school-management-system-project-documen/11876052**](https://www.studocu.com/en-gb/document/london-metropolitan-university/knowledge-in-policing/school-management-system-project-documen/11876052)

**Industry Websites:**

**Reference - School Management System** [**https://www.dreamclass.io/what-is-school-management-software/**](https://www.dreamclass.io/what-is-school-management-software/)

**Platforms for school administration :** [**https://www.sanomalearning.com/en/our-solutions/digital-platforms-for-administration-workflow/**](https://www.sanomalearning.com/en/our-solutions/digital-platforms-for-administration-workflow/)