ABSTRACT

A college management project aims to streamline academic and administrative processes within an educational institution. Utilizing software, it automates tasks such as student enrollment, Adding subject,updating department, and Uploading notes. This system enhances communication among students, faculty, and administrators, fostering a more efficient learning environment.

Additionally, it helps manage resources, faculty assignments, and campus facilities. By centralizing data, the project simplifies decision-making processes and ensures accurate record-keeping. Ultimately, a well-implemented college management project contributes to the overall effectiveness and organization of the institution, optimizing the educational experience for both students and staff.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1. INTRODUCTION

This document aims at defining overall software requirement for COLLEGE MANAGEMENT SYSTEM. Efforts have been made to define the requirements exhaustively and accurately. The final product will be having only features/functionalities mentioned in this document and assumptions for any additional functionality/feature should not be made by any of the parties involved in developing/testing/implementing /using this product.

College Management System divided in two modules

- Administration
- Faculty
- Student

Administrator Features

- Login
- Add / Delete Faculty
- Add / Delete Student
- Add / Delete Department
- Add / Delete Subject
- Logout

Faculty Features

- Login
- Add / Edit Marks of student
- Upload / Delete Notes
- logout

Student Features

- Login
- View the marks details
- View the uploaded notes
- logout

1.2. PROBLEM STATEMENT

Basically, all school or college has their own websites. Some of them are providing very basic services like admission, notification, infrastructure and availability of courses. Some of them providing extra services like result, online study materials, research facilities and training and placement (only for advertisement purpose). Organization is not designing and developing tools or application according to the requirement of present era students using latest technology with latest services to help students. That is why students visits their school or college .so this a website was one person manages every s moment of college details and student also view their details. Websites only for notification and result purpose.

1.3. MOTIVATION AND OBJECTIVE OF THE PROJECT

Our College Management System application provides an autonomous solution of the paper-based work. This is controlled and monitored by admin. The man power is reduced by using this application. It provides accurate information all the time as faculty member or student needed. But utmost care has been taken to make the system efficient and user friendly.

1.4. PROPOSED SOLUTION AND ADVANTAGES

I have designed the given proposed system in the HTML, CSS & NODEJS to automate the process of COLLEGE. This project will keep track of faculty who adds department, student details, and manages the subject every mark of students are stored in this database. Advantages of using this proposed system are: Efficiency, Elimination of paper based tasks, Control and Security

CHAPTER 2

SYSTEM DESIGN

2.1SCHEMA DIAGRAM

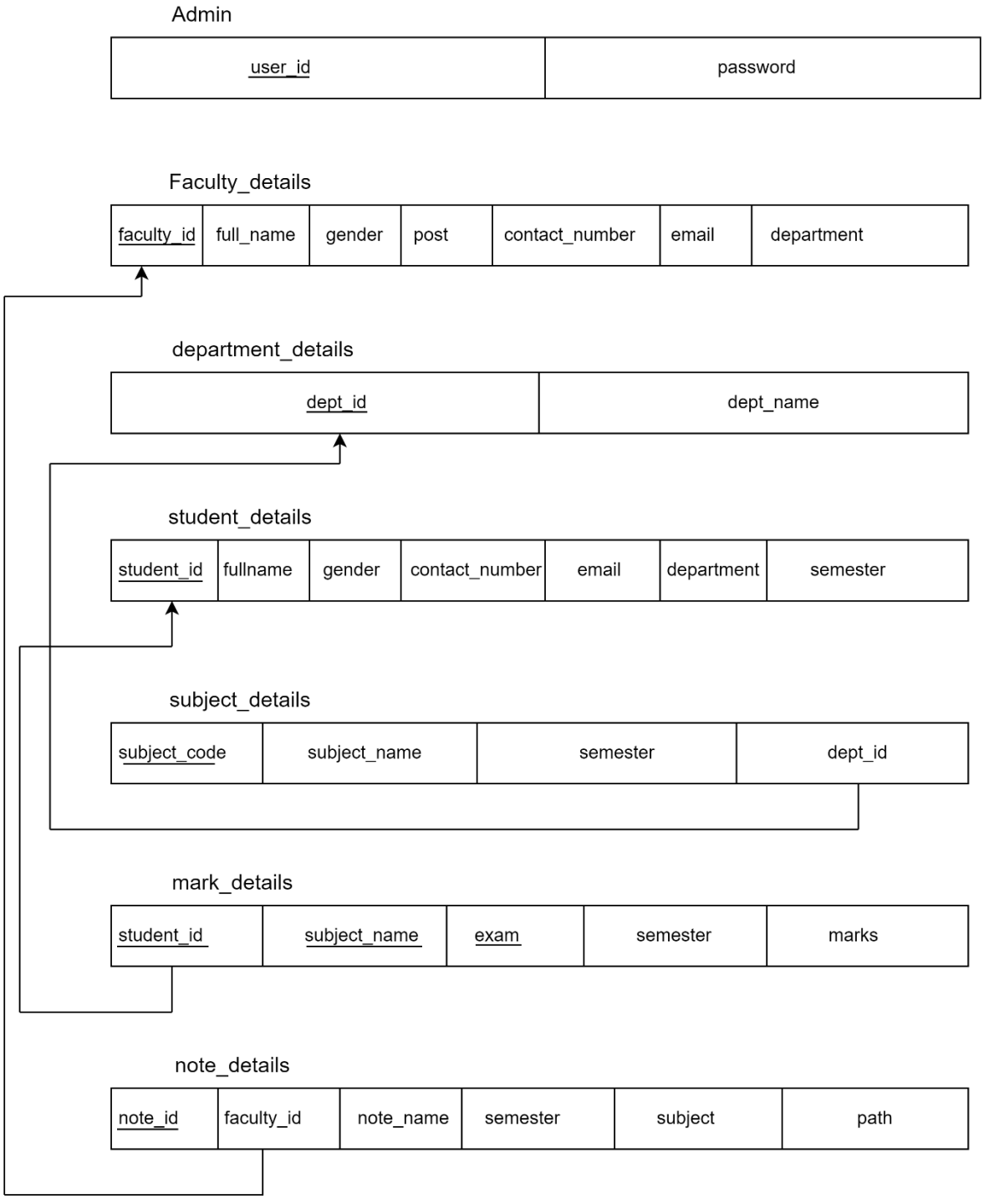

Fig 2.1.1: SCHEMA DIAGRAM

2.2ER DIAGRAM

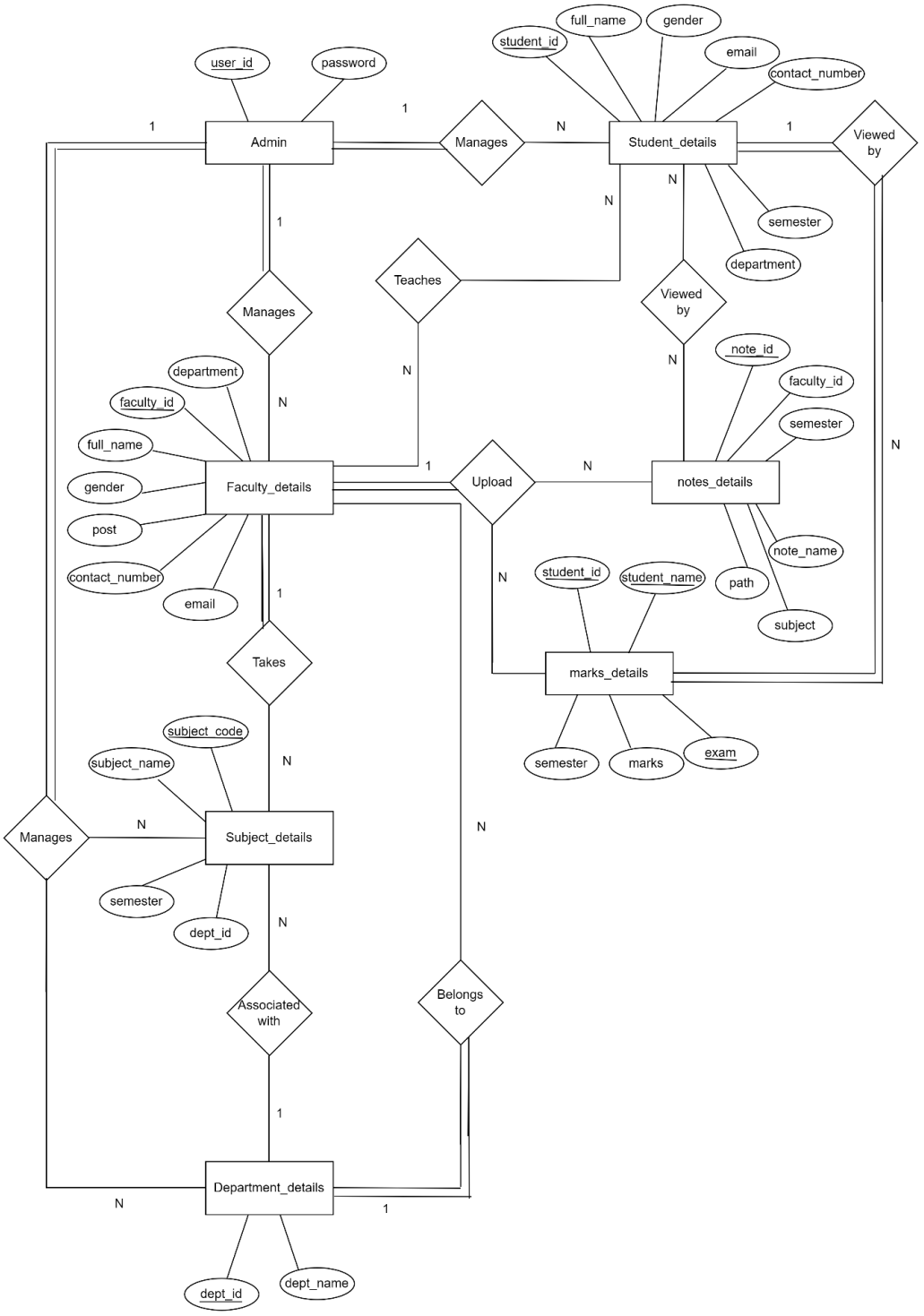

Fig 2.2.1: ER DIAGRAM

CHAPTER 3

IMPLEMENTATION

3.1. HARDWARE SPECIFICATIONS

- High-speed Network Connectivity.

3.2. SOFTWARE SPECIFICATIONS

- Browser
- Visual Studio Code
- MySQL Database
- Node JS

3.3. LANGUAGE USED FOR IMPLEMENTATION

The language used for implementation are as follow:

- Front End: HTML, CSS, JS
- Back End: NodeJS

HTML: -

Hypertext Mark-up Language (HTML) is the standard mark-up language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages.HTML is used to create and save web document. E.g. Notepad/Notepad++

CSS:-

A style sheet language used for describing the presentation of a document written in a markup language such as HTML or XMLCSS can be used for very basic document text styling — for example, for changing the color and size of headings and links. It can be used to create a layout — for example, turning a single column of text into a layout with a main content area and a sidebar for related information.

JAVASCRIPT:-

JavaScript is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else. Today, JavaScript has plenty of frameworks and libraries to simplify complex projects, JavaScript is also utilized on the server-side through platforms like Node.js. It supports

event-driven, asynchronous programming, making it powerful for creating responsive and engaging web applications.

NODE JS:-

Node.js is a runtime environment that allows JavaScript code to be executed on the server side, enabling server-side scripting. It uses the V8 JavaScript engine from Google Chrome and is known for its event-driven, non-blocking I/O model, making it efficient for handling concurrent connections. Node.js is widely used for building scalable network applications, and its package ecosystem, npm, is one of the largest and actively maintained. It's particularly popular for creating web servers and real-time applications.

3.4. OUTPUT TESTING

- While executing mysql database connection code we were not able to make the connection of backend to front end . So to solve this problem we had to create a new mysql user with password. After this the connection was successful.
- The connection was successful but the data entered in front end was not storing in backend, since all the attributes data types in backend were not set to varchar. Therefore, we modified the node js code and mysql query accordingly.
- If we enter wrong password, local host says wrong password

CHAPTER 4

TABLE DESCRIPTION

faculty_details

Field Name	Type	Constraints	Description
faculty_id	varchar(20)	primary key	to store faculty id
full_name	varchar(255)	not null	to store faculty name
gender	varchar(10)	null	to store faculty gender
post	varchar(50)	null	to store faculty post
contact_number	varchar(15)	null	to store faculty phone number
email	varchar(255)	null	to store faculty mail id
department	varchar(50)	null	to store faculty department

department_details

Field Name	Type	Constraints	Description
dept_id	varchar(20)	primary key	to store department id
dept_name	varchar(255)	not null	to store department name

subject_details

Field Name	Type	Constraints	Description
subject_code	varchar(255)	primary key	to store subject id
subject_name	varchar(255)	not null	to store subject name
semester	varchar(255)	not null	to store semester
dept_id	varchar(255)	foreign key	to store department id

student_details

Field Name	Type	Constraints	Description
student_id	varchar(20)	primary key	to store student id
full_name	varchar(255)	not null	to store student name
gender	varchar(10)	not null	to store student gender
contact_number	varchar(15)	not null	to store student phone number
email	varchar(255)	not null	to store student mail id
department	varchar(255)	not null	to store department name
semester	varchar(255)	not null	to store semester

login_detail

Field Name	Type	Constraints	Description
user_id	varchar(20)	primary key	to store user id
password	varchar(255)	not null	to store password

mark_details

Field Name	Type	Constraints	Description
student_id	varchar(20)	primary key, foreign key	to store student id
subject_name	varchar(255)	primary key	to store subject name
semester	varchar(20)	null	to store semester
exam	varchar(10)	primary key	to store exam type
marks	int	null	to store marks

note_details

Field Name	Type	Constraints	Description
note_id	int	primary key, auto increment	to store note id
faculty_id	varchar(20)	foreign key	to store faculty id
note_name	varchar(255)	null	to store note name
semester	varchar(50)	null	to store semester
subject	varchar(255)	null	to store subject name
path	varchar(255)	null	to store path of the file

CHAPTER 5

RESULTS

4.1. SNAPSHOTS

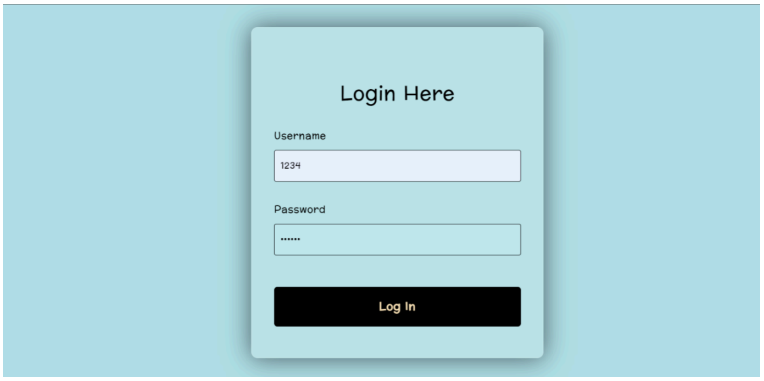

Fig 4.1.1: Login Page

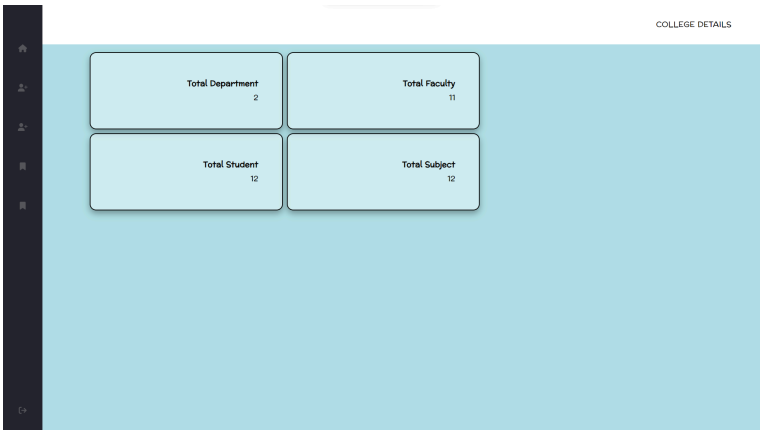

Fig 4.1.2: Admin Homepage

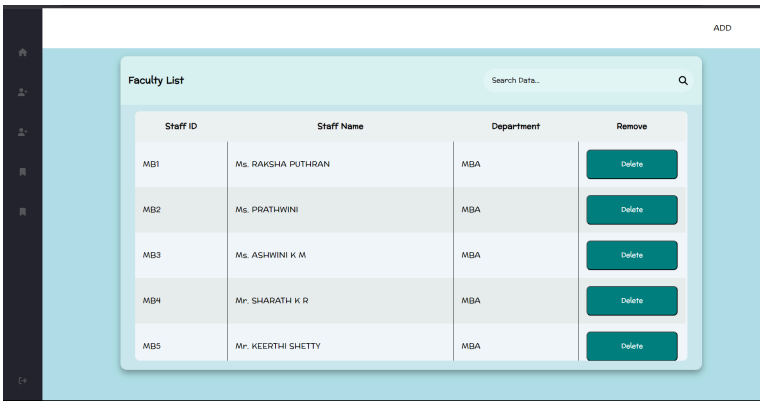

Fig 4.1.3: Faculty List Page

Fig 4.1.4: Add Faculty Page

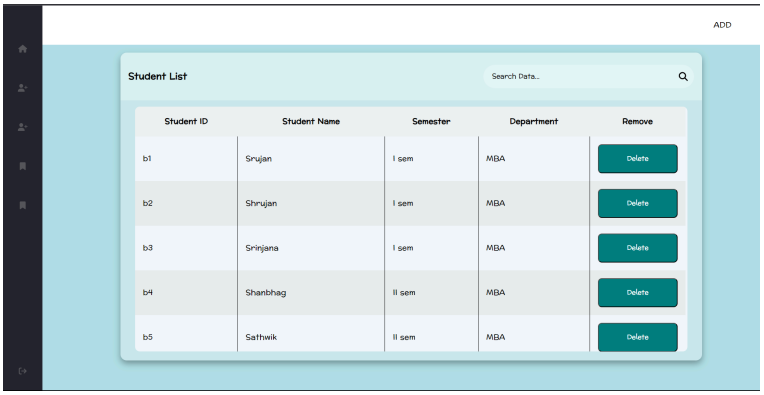

Fig 4.1.5: Student List Page

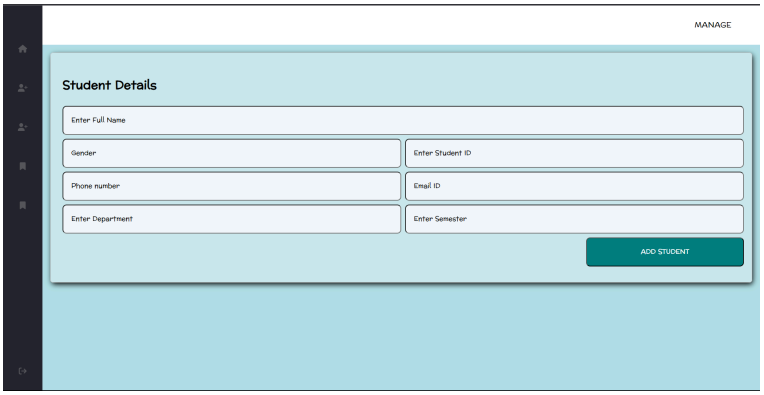

Fig 4.1.6: Add Student Page

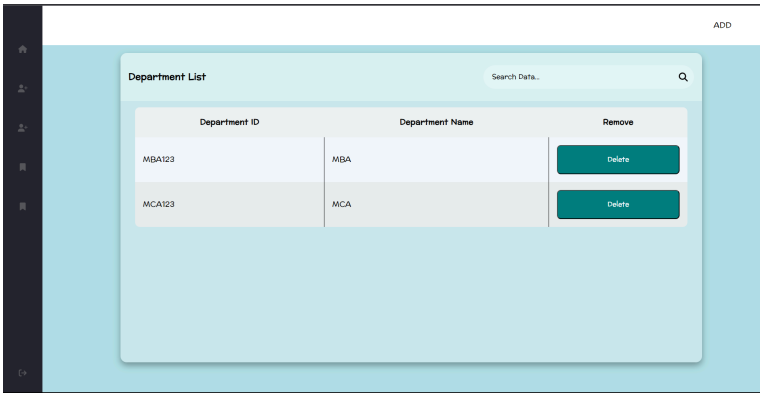

Fig 4.1.7: Department List Page

Fig 4.1.8: Add Department Page

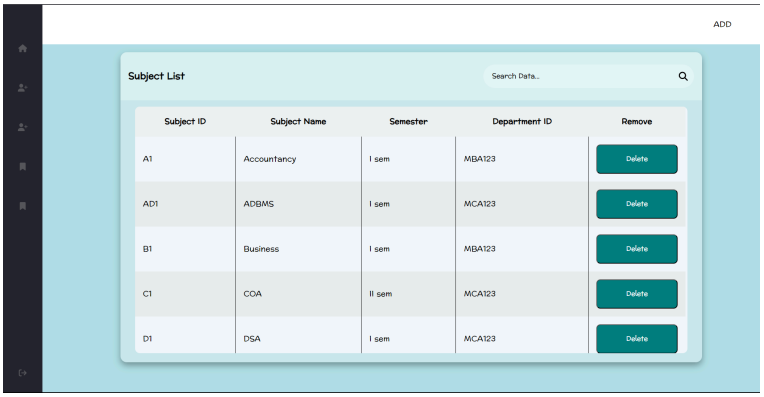

Fig 4.1.9: Subject List Page

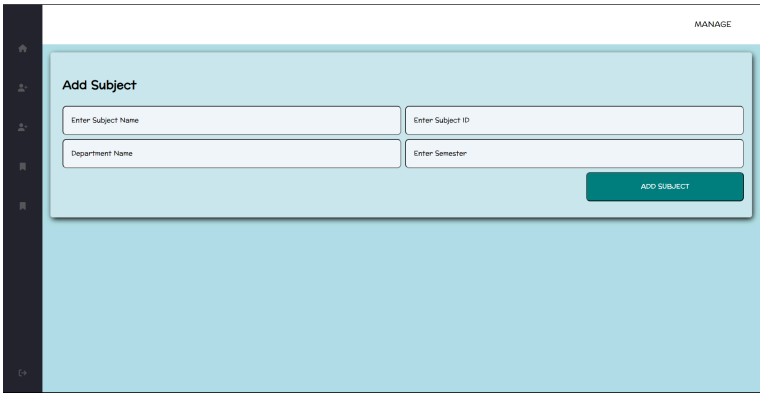

Fig 4.1.10: Add Subject Page

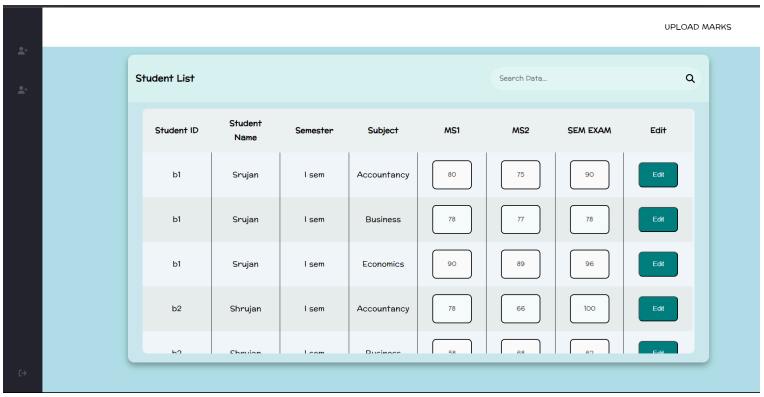

Fig 4.1.11: Edit Marks Page

Fig 4.1.12: Add Marks Page

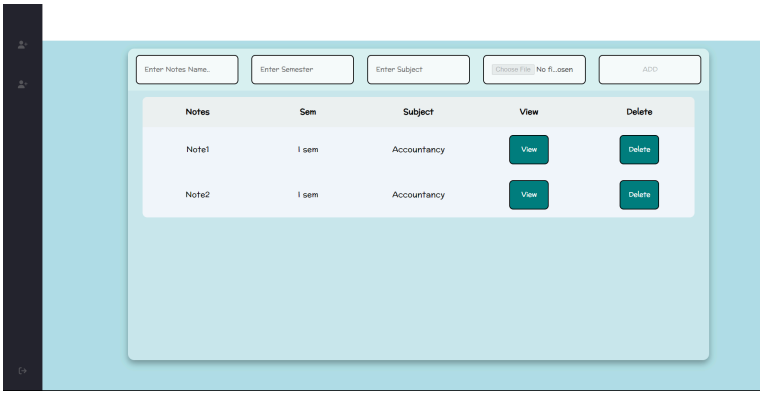

Fig 4.1.13: Add / Delete Notes Page

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

In conclusion, effective college management plays a pivotal role in ensuring the seamless functioning of educational institutions. By implementing streamlined processes, leveraging technology, and fostering transparent communication, colleges can enhance the overall learning experience for both students and staff.

Moving forward, the future of college management lies in embracing advanced technologies like artificial intelligence and machine learning for automation and personalized learning. Improved communication channels, robust data analytics for decision support, and a strong focus on cybersecurity are essential components of future enhancements. Additionally, investing in online learning platforms, sustainability initiatives, and fostering community partnerships will contribute to the adaptability and success of college management systems. Continuous professional development for faculty and staff alongside a commitment to flexibility and student engagement, will position colleges to thrive in the ever-evolving landscape of education.

Looking ahead, we can make college management even better in simple ways. We can use new technologies like smart computers to help with tasks, and we can improve how we talk to each other using phones and apps. Also, we can look at important information about students and teachers to make better choices. In the future, we should use more online tools for learning and do our best to take care of the environment. It's also a good idea to keep learning and training our teachers so they can teach us even better. By working together and being open to new ideas, we can make sure college is a great place for everyone.

CHAPTER 7

REFERENCES

- Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017 Pearson
- Herbert Schild: JAVA The Complete Reference, 9th Edition, Tata McGraw Hill, 2007.
- Jim Keogh: J2EE The Complete Reference, McGraw Hill, 2007