NETAJI SUBHAS INSTITUTE OF TECHNOLOGY
DELHI UNIVERSITY

Dwarka Sector-3, Dwarka, Delhi, 110078

COLLEGE MANAGEMENT SYSTEM

A Database Management System Project

(Github Repo)

made by -

VISHAL JHA

BACHELOR OF ENGINEERING
Electronics and Communication
(LinkedIn)

RITIK SINGLA

BACHELOR OF ENGINEERING
Electronics and Communication
(LinkedIn)

ABSTRACT

The purpose of College Management System is to automate the existing manual system by the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

College Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather than concentrating on the record keeping. Thus it will help organizations in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

Using various technologies we have implemented the CMS application that provides the above functionalities. The software was also tested using the various testing methods and results were positive.

INTRODUCTION

1.1 About

The "College Management System" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner. The application is reduced as much as possible to avoid errors while entering the data. It also provides an error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly. College Management System , as described above, can lead to error free, secure, reliable and fast management systems. It can assist the user to concentrate on their other activities rather than concentrating on the record keeping. Thus it will help organizations in better utilization of resources.

1.2 Purpose

The purpose is to design software for a college database which contains up to date or accurate information of the college. That should improve efficiency and flexibility of college record management and to provide a common and or simple platform for everyone to access the student's information.College Automation System consists of different modules such as student, faculty, admin etc. Our main purpose is to create a software which will manage the working of these different modules. The interconnectivity among modules reduces the time to perform different operational tasks.

1.3 Scope

The CMS will permit process impact gathering information about students from the admission cell and give it to staff. And another very important impact is taking classes, conducting exams, maintaining attendance records, fee details, assignments, marks etc.

PROJECT DESCRIPTION

2.1 Perspective

The CMS is a system that can help maintain students and staff information easily and efficiently. A fully integrated web-based ERP will capture and create accurate, consistent and timely relevant data, and assist in intelligent business decision-making. The primary purpose of CMS is to provide mechanisms for automated processing and management of the entire institution. It reduces data error/ data redundancies, ensures that information is managed efficiently and is always up-to-date. Complete student histories for all years, can easily be searched, viewed and reported on the press of a button.

The CMS is a stand alone product and doesn't depend upon the availability of other websites. The system will have an administrator who has full-fledged rights with regards to performing all actions related to control and management of the website and all relevant data.

2.2 Product Features and User Classifications

There are several types of end users for the CMS. They are broadly divided as Students, Staff and the Administrator. Each of these classes have their own set of features

- **ADMIN** who can view and edit the details of any students/staff. Can add/edit departments, courses, classes and time-tables.
- **STAFF** who can view students details, add/update assignments, marks and attendance of a particular student. They can see the time-table of a particular class also.
- **STUDENT** who can update profile/ add solution to assignments and see marks/attendance.

The features that are available to administrator are :

- An administrator can login into the system and perform any of the available operations :
    - add a student
    - view the student details
    - modify student data

    - delete student record
    - add staff
    - view staff details
    - modify staff data
    - delete staff record
    - add department
    - modify department
    - add/modify courses
    - add/modify classes
    - can view the record of all the students and staff of a particular class/ course/ department.

The features that are available to staffs are :

- A staff can login into the system and perform any of the available operations :
    - view his/her personal details (name, dob, age, address, joining date)
    - edit his/her details
    - view the department (s)he belongs to.
    - the classes they teach.
    - generate class report
    - mark attendance of a particular student
    - update the attendance
    - see the time-table of a class.

The features that are available to students are :

- A student can login into the system and perform any of the following operations :
    - view his/her personal details (name, dob, age, address, joining date)
    - edit his/her details
    - view the department (s)he belongs to.
    - the classes they belong
    - view his/her attendance
    - add solution to the assignment
    - view his/her marks

2.3 Operating Environment

The operating env. for CMS application are listed below :

- Operating System : Windows 10
- Database : MySQL
- Frontend : HTML/CSS/JS/Bootstrap
- Backend : Node.js
- Deployment : Heroku

2.4 Constraints

- Users should be aware of the internet.
- Users must have a valid id and password.
- Only Admin, Staff and Students can access.
- Every user must be comfortable using a computer.
- All operations are in English so users must have basic knowledge of English.

INTERFACE REQUIREMENT

3.1 User Interfaces

The User Interfaces are made using HTML / CSS and JS. Each type of end user has different types of UI as per requirements. These are user friendly and anyone can use it. No extra efforts or manual work is needed for marking attendance, assigning assignments, submitting assignments etc.

3.2 Hardware Interfaces

Since neither the mobile application nor the web portal have any designated hardware, it does not have any direct hardware interfaces. Any browser can be used to access the web app. Though some features of js may not work in old versions of browsers such as internet explorer, so it is best recommended to use this on the latest browsers.

3.3 Software Interfaces

The following is a list of software used in the making of the project.

- Operating System: We have chosen Windows operating system for its best support and user-friendliness.
- Backend: We have made our backend using node.js as it offers an easy scalability (both in horizontal as well as vertical directions.)
- Database: Our main motive is to use relational databases and so we are using MySQL as a database for our project. It provides Data Security, On Demand Scalability and comes with a very user friendly MySQL Workbench which can be used to see our data more easily and run our scripts.

SYSTEM DESIGN

4.1 System Design of CMS

Various Design concepts and processes were applied to this project on the basis of which a complete logical system is built which fulfils the given requirements. There were two steps while design analysis :

1. **Primary Design Phase :**

   The system is designed at block level. We first decided the end users that will be using our application. On finalizing that and following the concepts like separation of concerns, the software is divided into individual modules based upon the end users that are functionally independent and incorporates information hiding. There are 3 types of end users (modules) - students, staff and the administrator.

2. **Secondary Design Phase :**

   Each module/block was studied in detail during this phase.

4.2 Admin

The administrator will have access to all the information in the different tables in the database. They will be able to add an entry in any table and also edit them. They will be responsible for maintaining the data. The main aim here is to provide a user friendly environment for the admin so that they can easily adapt themselves with the environment. They will be responsible for the creation/deletion of student / staff / department / course / class information. Only authorized users will be made admin and thus we have protected the routes. They will be provided with search and filter features so that they can access the data efficiently. Different Views of Admin are described below.

- **Dashboard**

  An overview for admin where they can easily navigate to other submodules for performing its task.

- **Profile**

  Admin will be able to see it's own information. They will be able to see a complete overview of total students / staff in the college and various courses / departments.

- **Settings**

  Admin will be able to change it;s own information like name, email, password, add other informations such as address, contact etc.

- **Department**

  The admin is responsible for creating new departments and updating the already existing ones. Here they will be provided with search and filter operations so that they can work efficiently. Necessary information will be needed while adding new departements.

- **Course**

  The admin is responsible for creating new courses and updating the already existing ones. Here they will be provided with search and filter operations so that they can work efficiently. Necessary information will be needed while adding new courses (like which department the course is associated with).

- **Class**

  The admin is responsible for creating new classes and updating the already existing ones. Here they will be provided with search and filter operations so that they can work efficiently. Necessary information will be needed while adding new classes.

- **Student Section**

  The admin will be able to see the complete information of all the students. They are also responsible for adding new students and updating the already existing ones. They will be provided with a user friendly UI to list down all the necessary information of students and if they want they can see the complete details by clicking on the relevant student's name.

- **Staff Section**

  The admin will be able to see the complete information of all the staff. They are also responsible for adding new students and updating the already existing ones. They will be provided with a user friendly UI to list down all the necessary information of staff and if they want they can see the complete details by clicking on the relevant staff's name.

- **Forget Password / Reset Password**

  In case an admin forgets his password or wants to reset it, he can do so by clicking on a reset password button. A reset link will be sent in the mail which will be active for only 20 min.

4.3 Staff

Each teacher belongs to a department and is assigned to classes with a course. Teachers will also have a username and password to login (which will be provided by the admin) . The different views for teachers are described below.

- **Dashboard**

  An overview for staff where they can easily navigate to other submodules for performing tasks.

- **Profile**

  Each staff member will be able to see it's own information. They will be able to see a complete overview of the classes they teach, number of students that belong to those classes etc.

- **Class**

  Teacher can see a list of all the classes he teaches. He can navigate into any class to see the list of students of that class.

- **Attendance**

  The teacher has the ability to add and also edit the attendance of each student. For entering the attendance, they will be given the list of students in each class and they can enter the attendance of the whole class on a day to day basis.

- **Assignments**

  The teacher has the ability to add and also edit the assignments of each class. They can also see the records of students who submitted the assignments and also the time in which they submitted that.

- **Time Table**

  The teacher can see the time-table of the classes he teaches (so that if he wants to take an extra class he can do so without any clashes).

- **Forget Password / Reset Password**

  In case a staff forgets his password or wants to reset it, he can do so by clicking on a reset password button. A reset link will be sent in the mail which will be active for only 20 min

4.3 Students

Each student belongs to a class identified by semester and section. Each class belongs to a department and are assigned a set of courses. Therefore, these courses are common to all students of that class. The students are given a unique username and password to login. Each of them will have a different view. These views are described below.

- **Dashboard**

  An overview for students where they can easily navigate to other submodules for performing tasks.

- **Profile**

  Each student will be able to see it's own information. They will be able to see the department they belong to, current semester registered courses, their personal information etc.

- **Attendance Information**

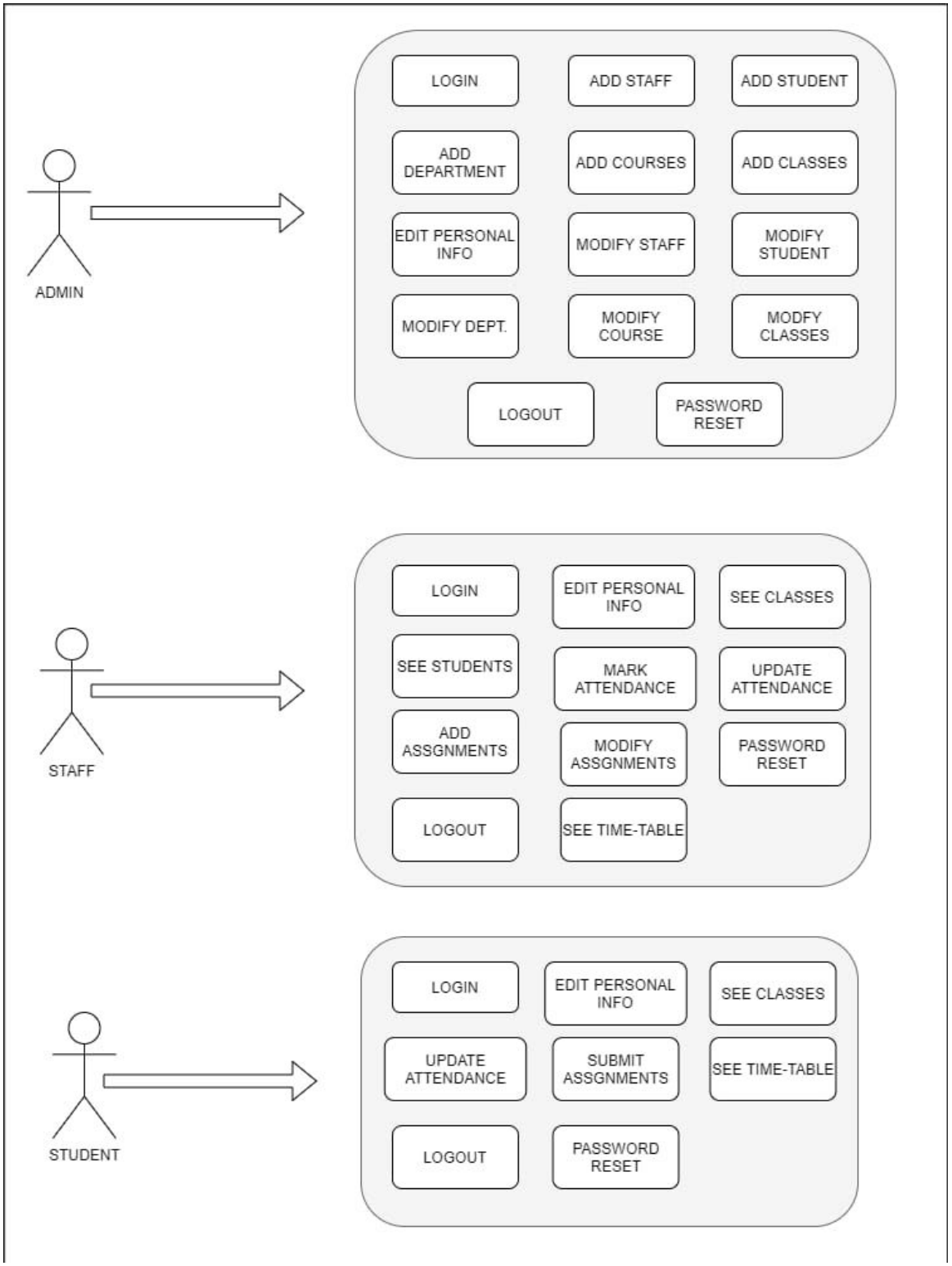
  Attendance for each course will be displayed. This includes the number of attended classes and the attendance percentage. If the attendance percentage is below a specified threshold, say 75%, It will be marked in red otherwise it will be in green.

- **Assignments Information and Submissions**

  Students will have a separate section for viewing the assignments along with their deadlines. He can also submit the assignments there only. The teacher view will get updated once the assignment is done.

4.3 End User Features (Diagram)

This is the use case diagram which depicts the user's interaction with the system. It also shows the relationship between the user and the different use cases in which the user is involved.



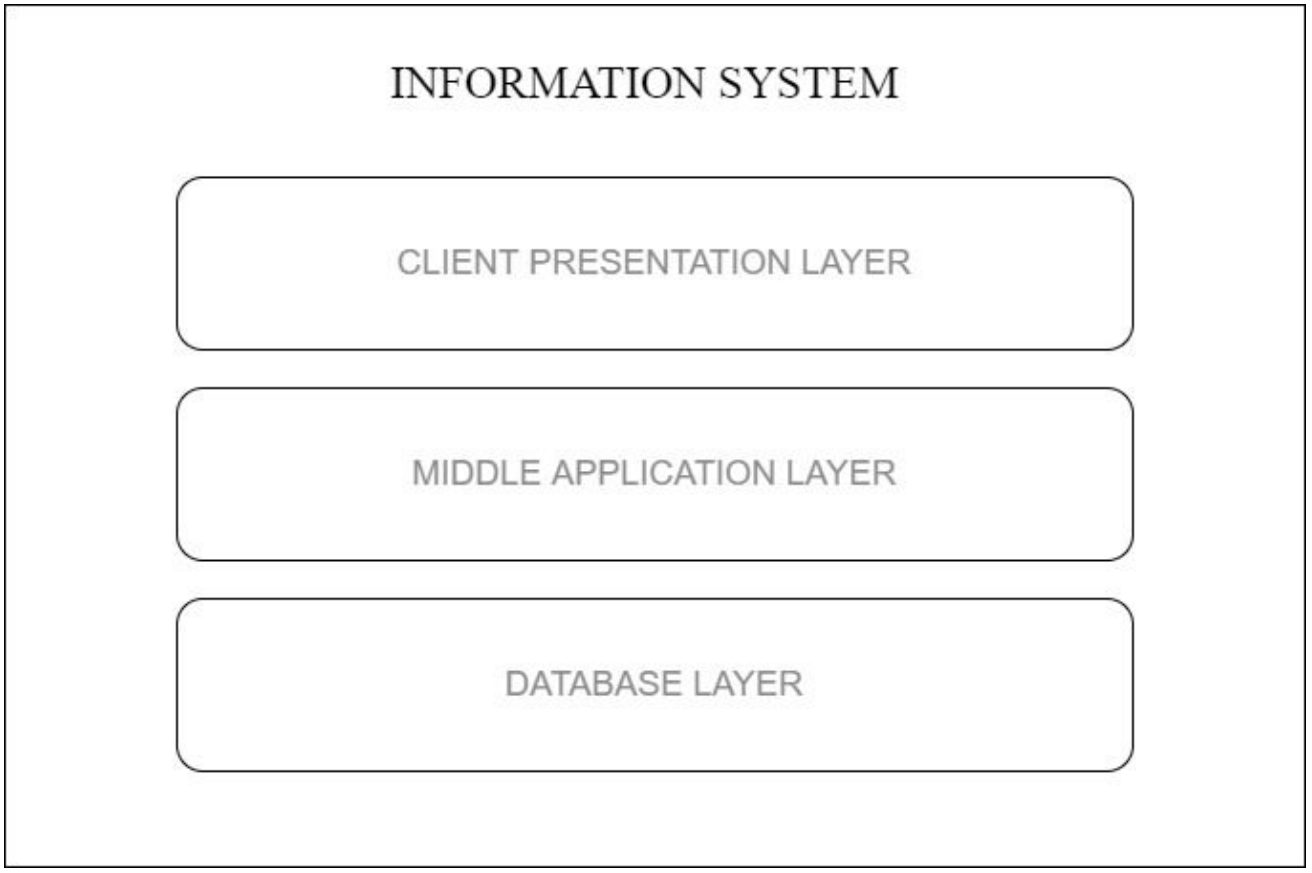
4.4 3 tier Architecture

Generalizing CMS architecture in 3 tiers. The 3 tiers consist of **presentation layer, application logic layer and data layer**. Any Information System needs to communicate with external entities, human users or other computers. Presentation layer allows these entities to interact with the system; it can also be implemented as a GUI interface and can be referred to as the client of the IS. Application layers do more than information delivery, they perform data processing (Business Logic and calculation) behind the results being delivered.

This tier is often referred to as

1. Services
2. Business rules
3. Business logic
4. Servers

The database layer is implemented using a Database Management System which in our case is MySql.



4.5 Database Design and Implementation

We are using MySQL as our database. The main objective of this project is to use Relational Database and hence MySQL is the best choice for that.

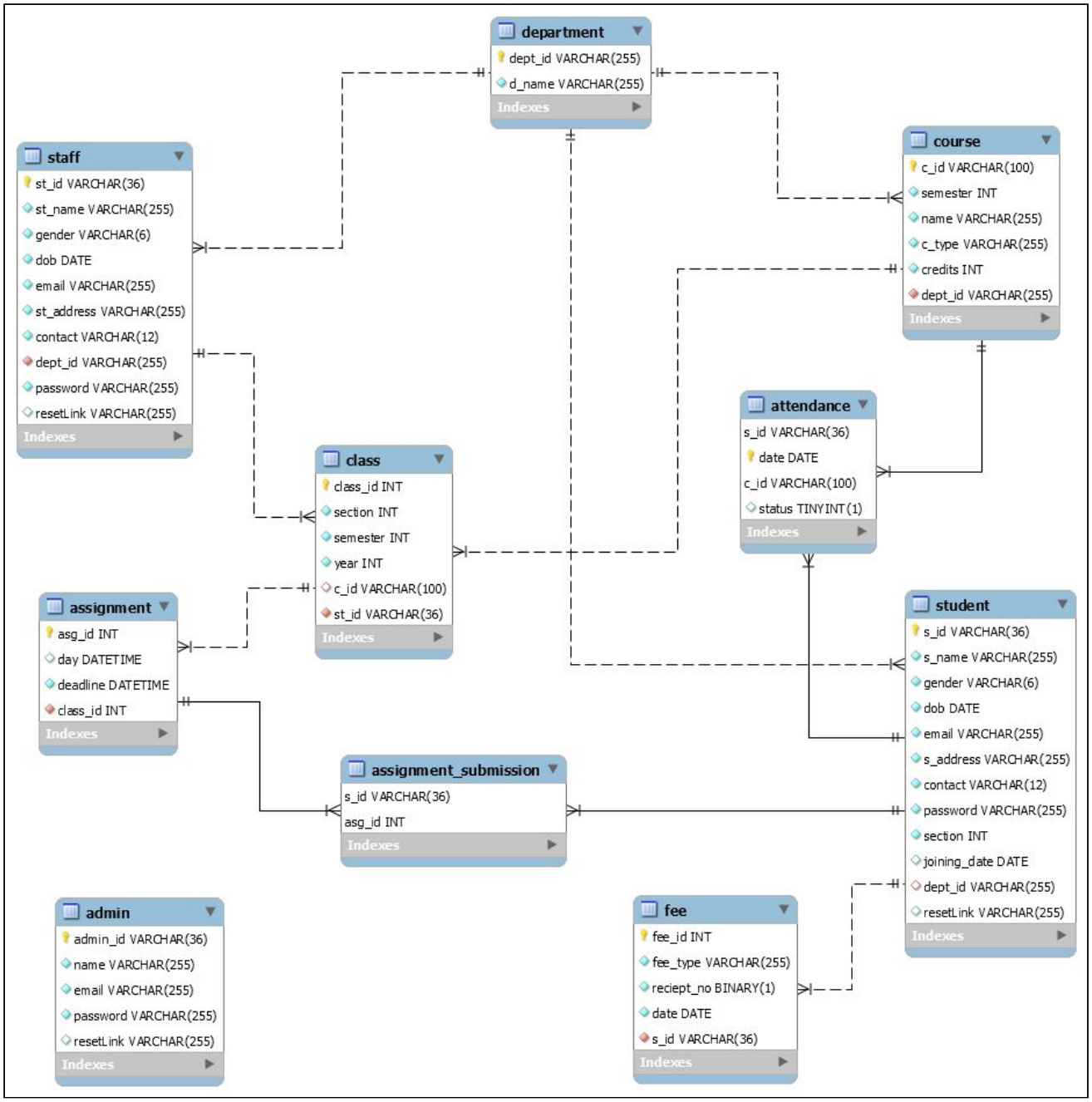
4.5.1 Key Features of Relational Model

- All data are represented as tables (relations)
- Tables consist of rows and columns (tuples)
- Rows are (officially) unordered (i.e., the order in which rows are referenced does not matter) A proper relational table contains no duplicate rows.
- Each table has a primary key, a unique identifier constructed from one or more columns.
- Most primary keys are a single column (e.g., s_id for students)
- A table is linked to another by including the other table's primary key. Such an included column is called a foreign key

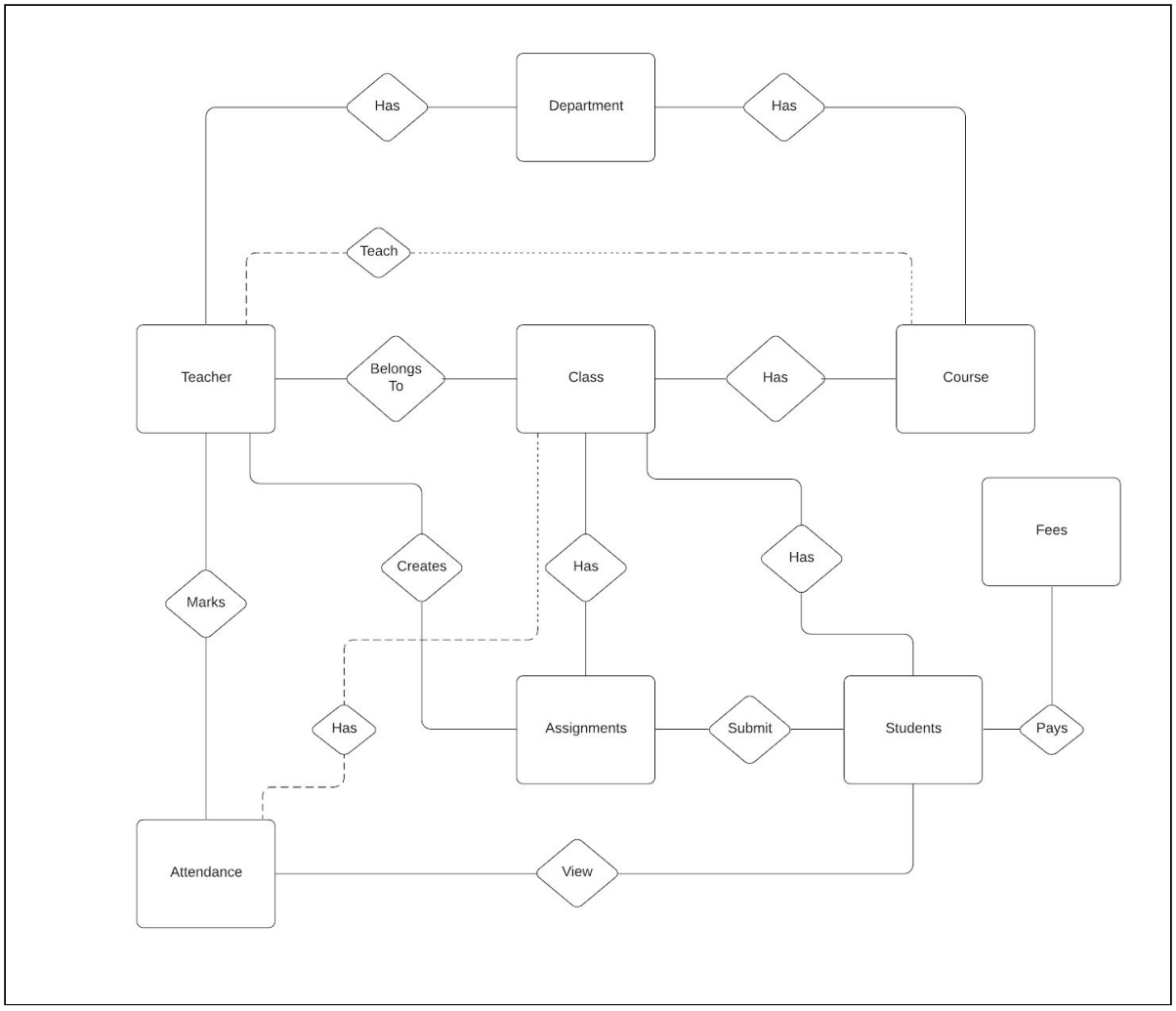
4.5.2 Qualities of our Database Design

- Reflects real-world structure of the problem
- Can represent all expected data over time
- Avoids redundant storage of data items
- Provides efficient access to data
- Supports the maintenance of data integrity over time
- Clean, consistent, and easy to understand
- No data redundancies (Reduced to 3NF).

4.5 Class Diagram



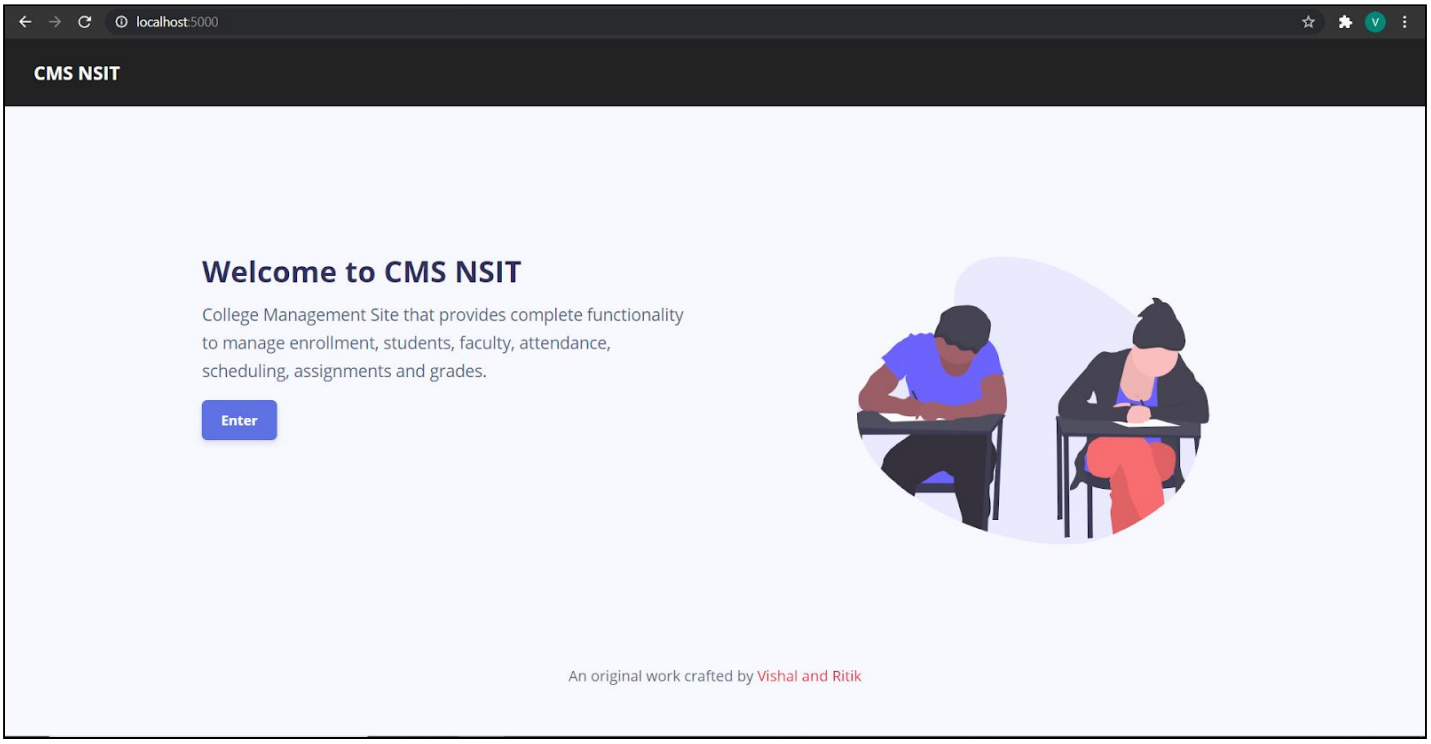
4.5 ER MODEL



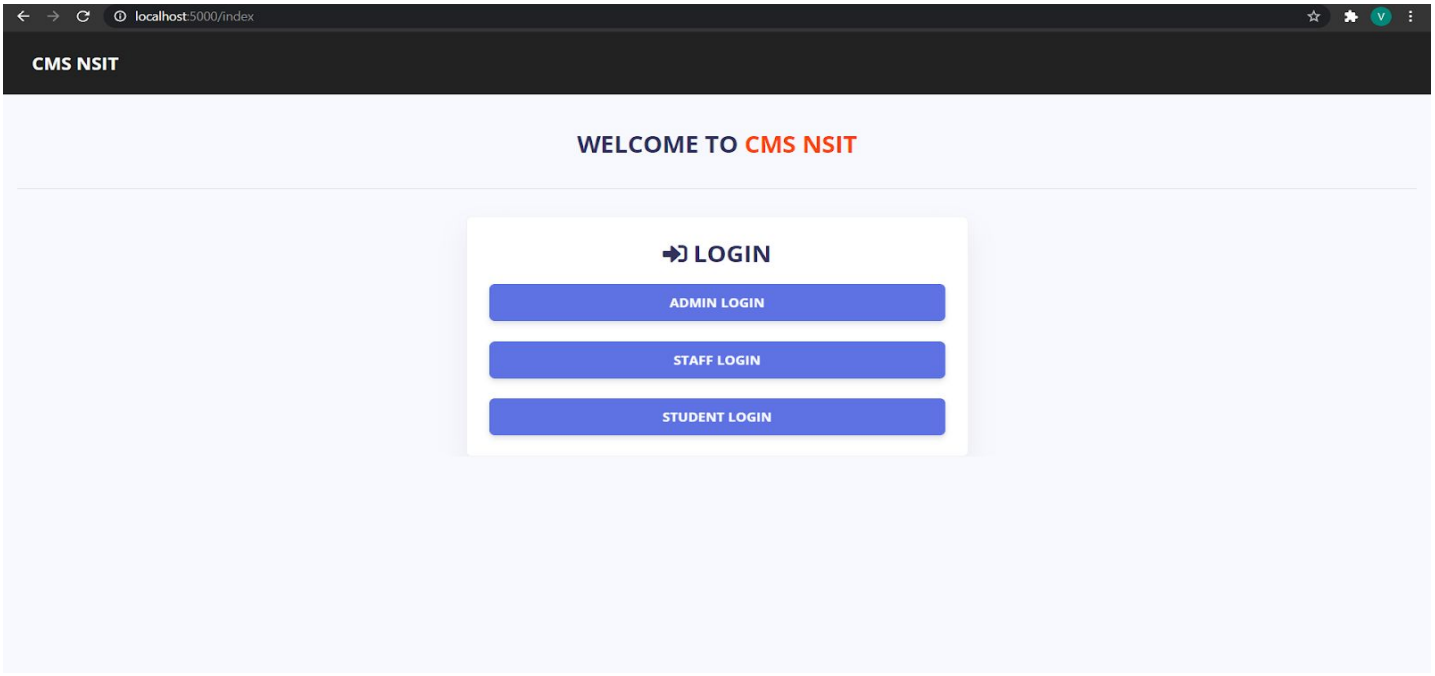
ENTITY RELATIONSHIP DIAGRAM

PROJECT OVERVIEW

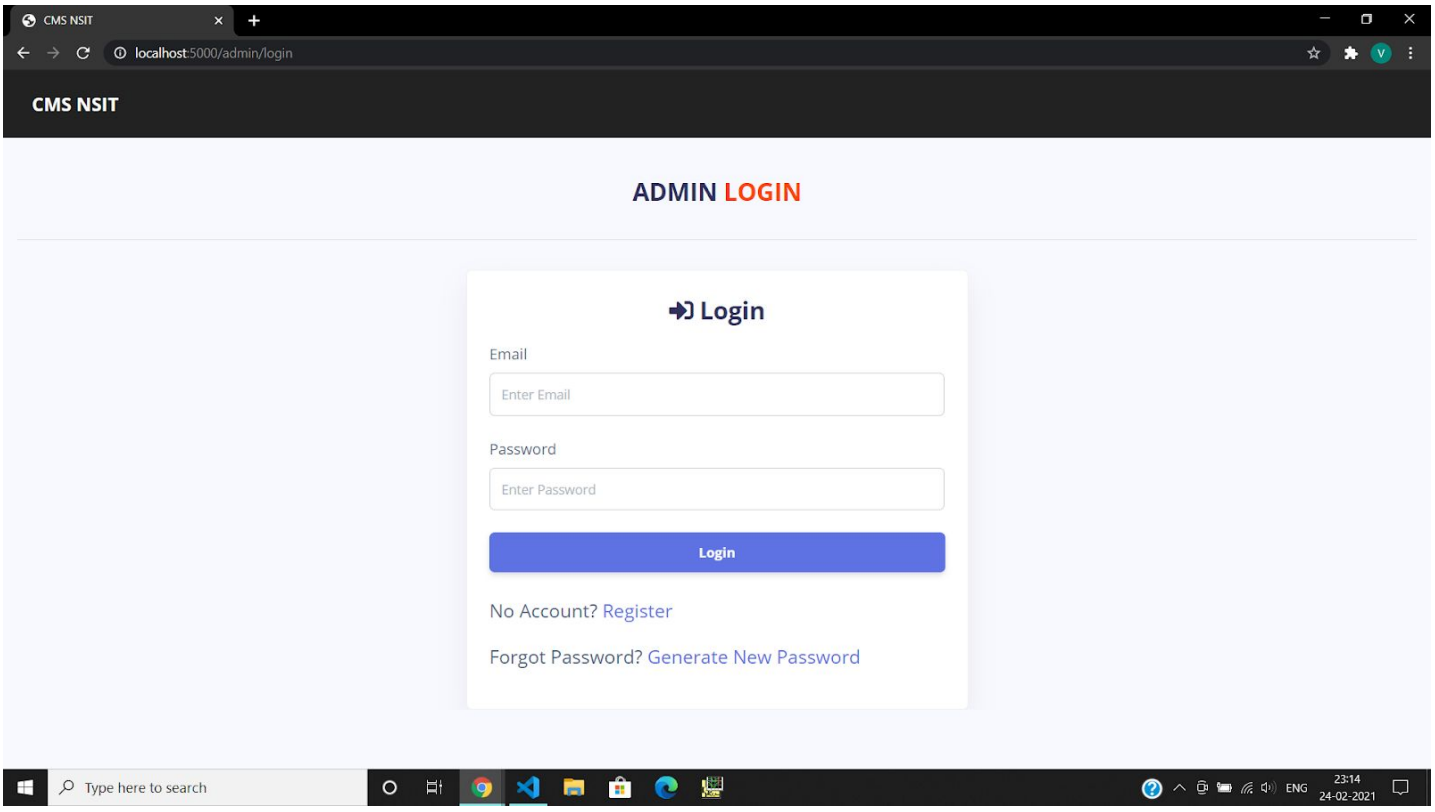
1. LANDING PAGE



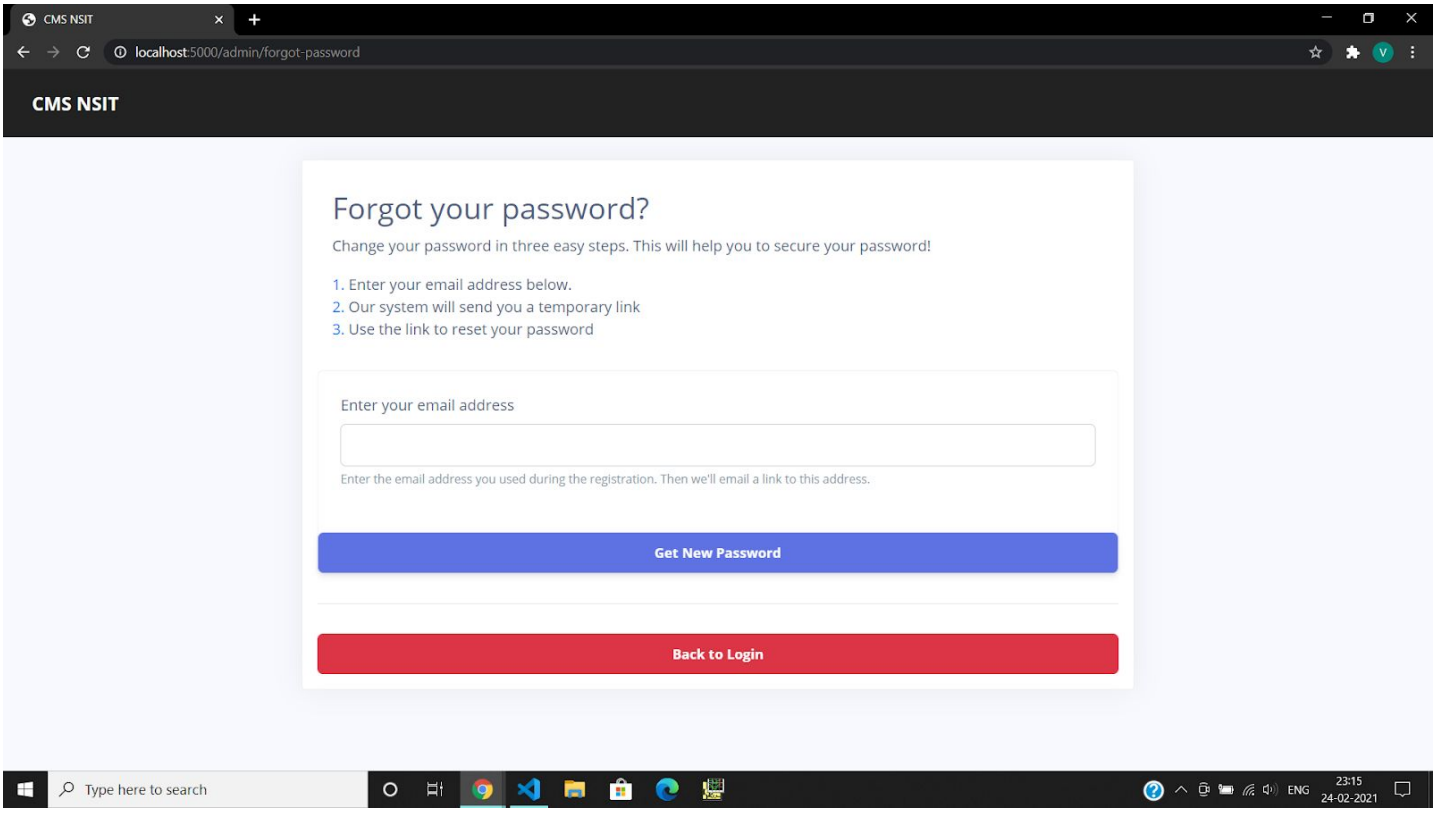
2. LOGIN AREA



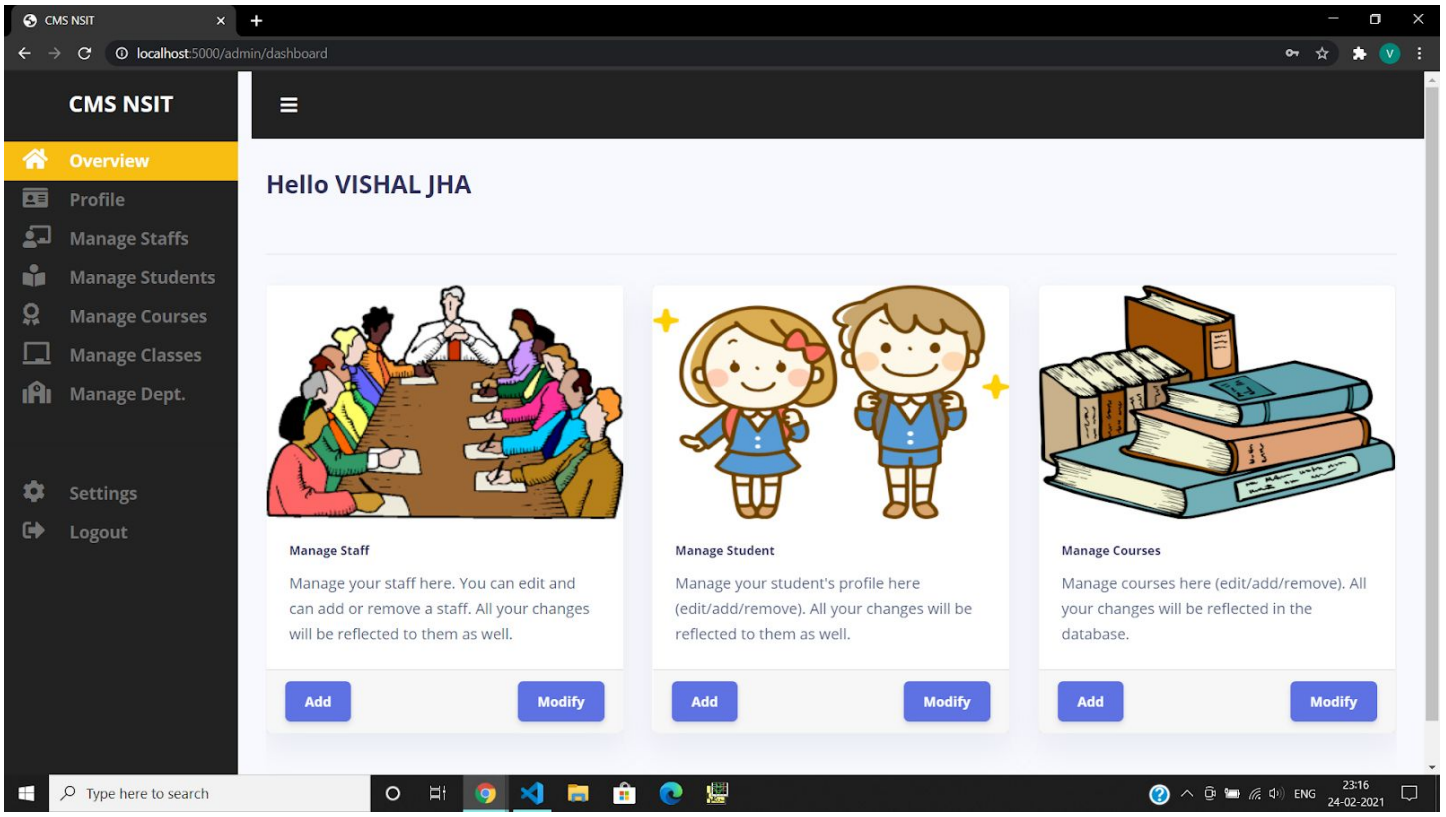
3. ADMIN LOGIN



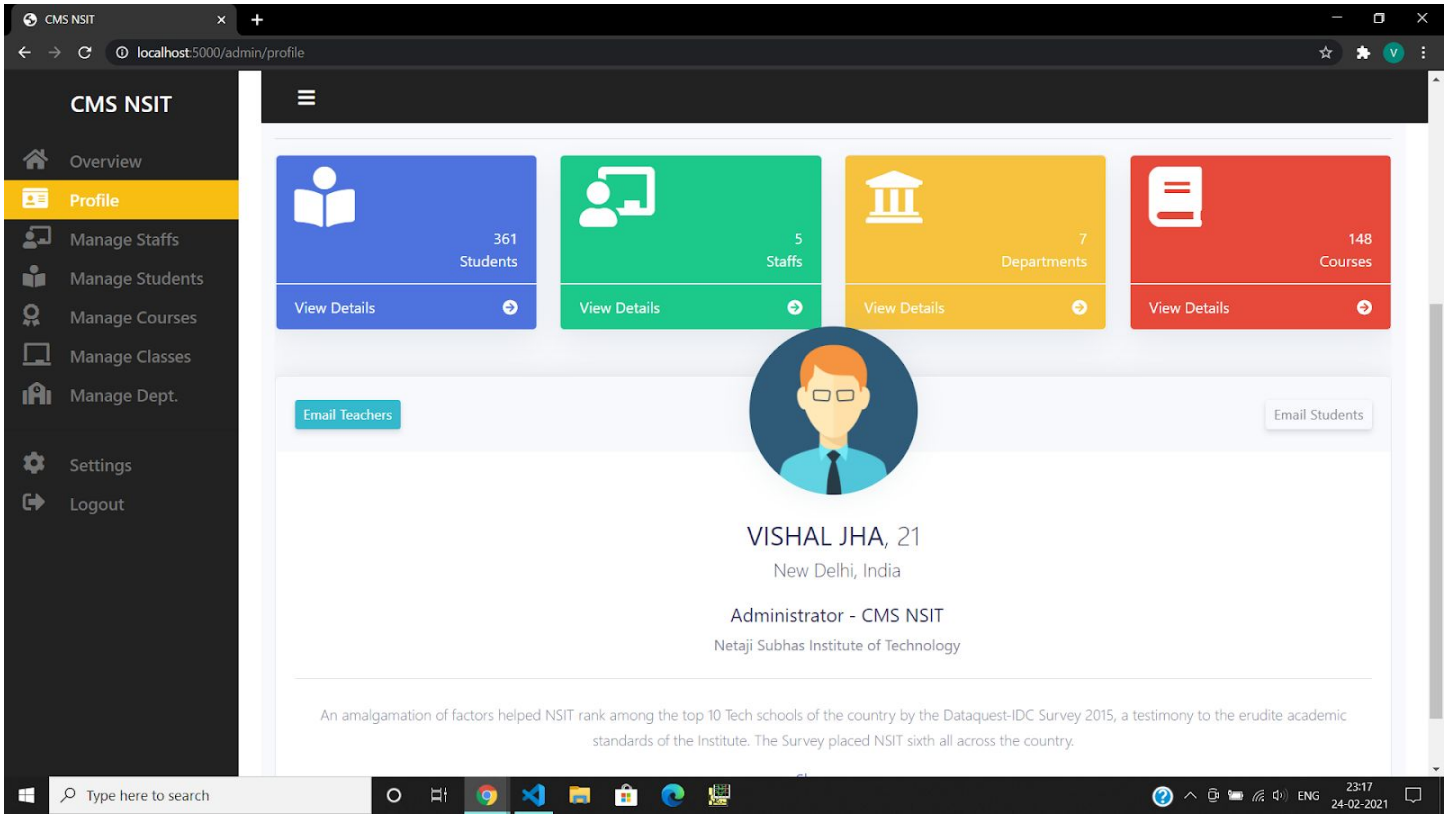
4. FORGOT PASSWORD - ADMIN



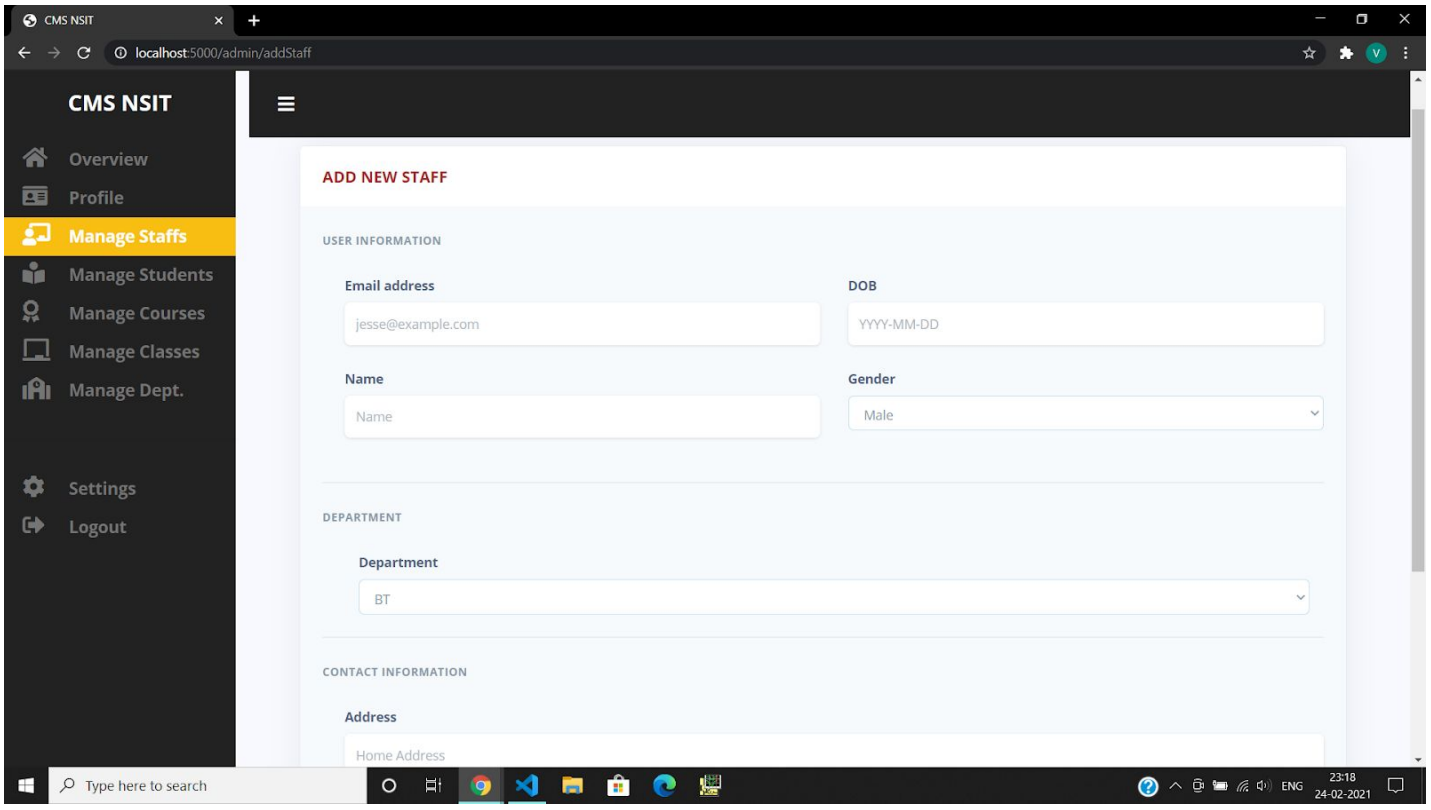
5. ADMIN DASHBOARD

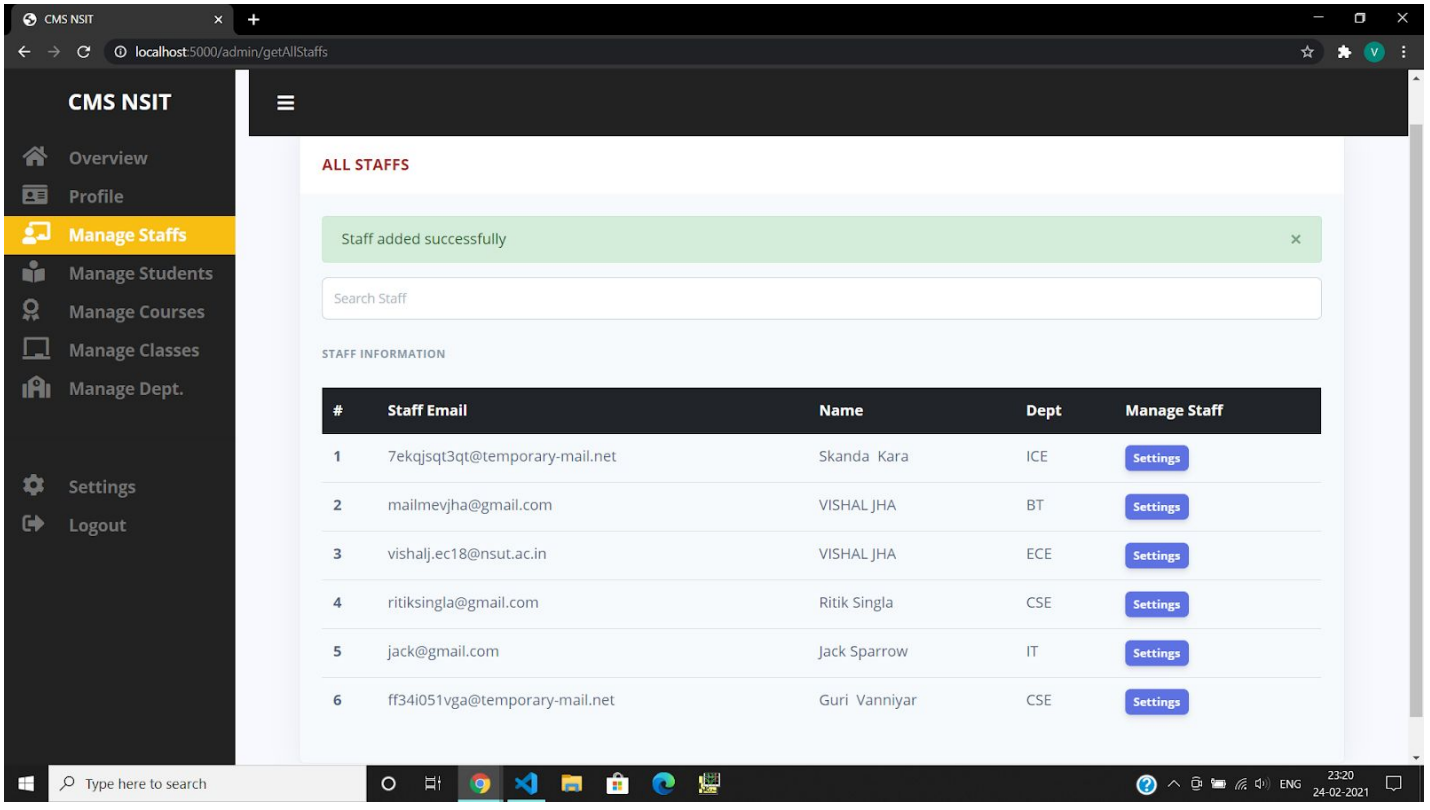


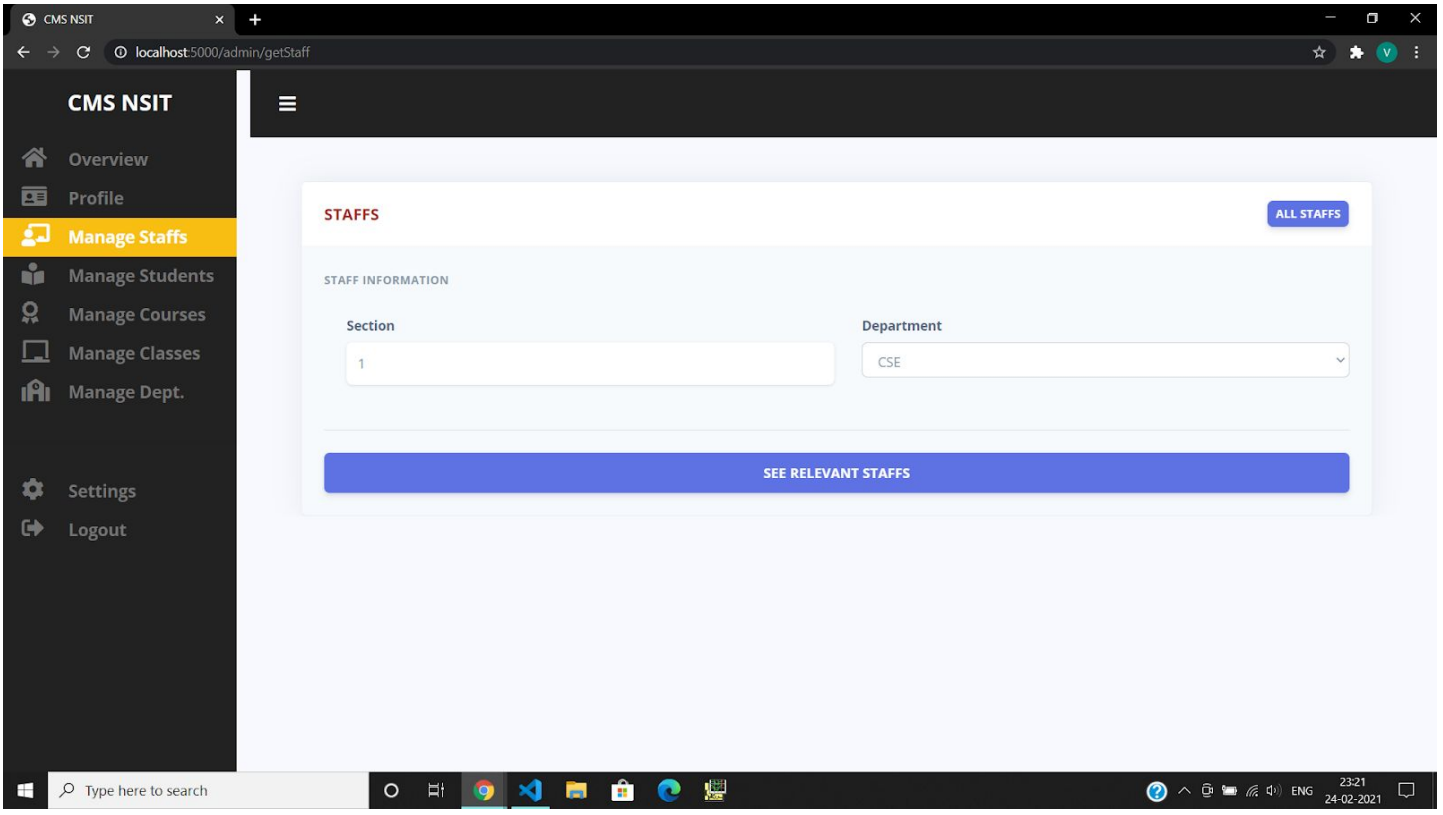
6. ADMIN PROFILE



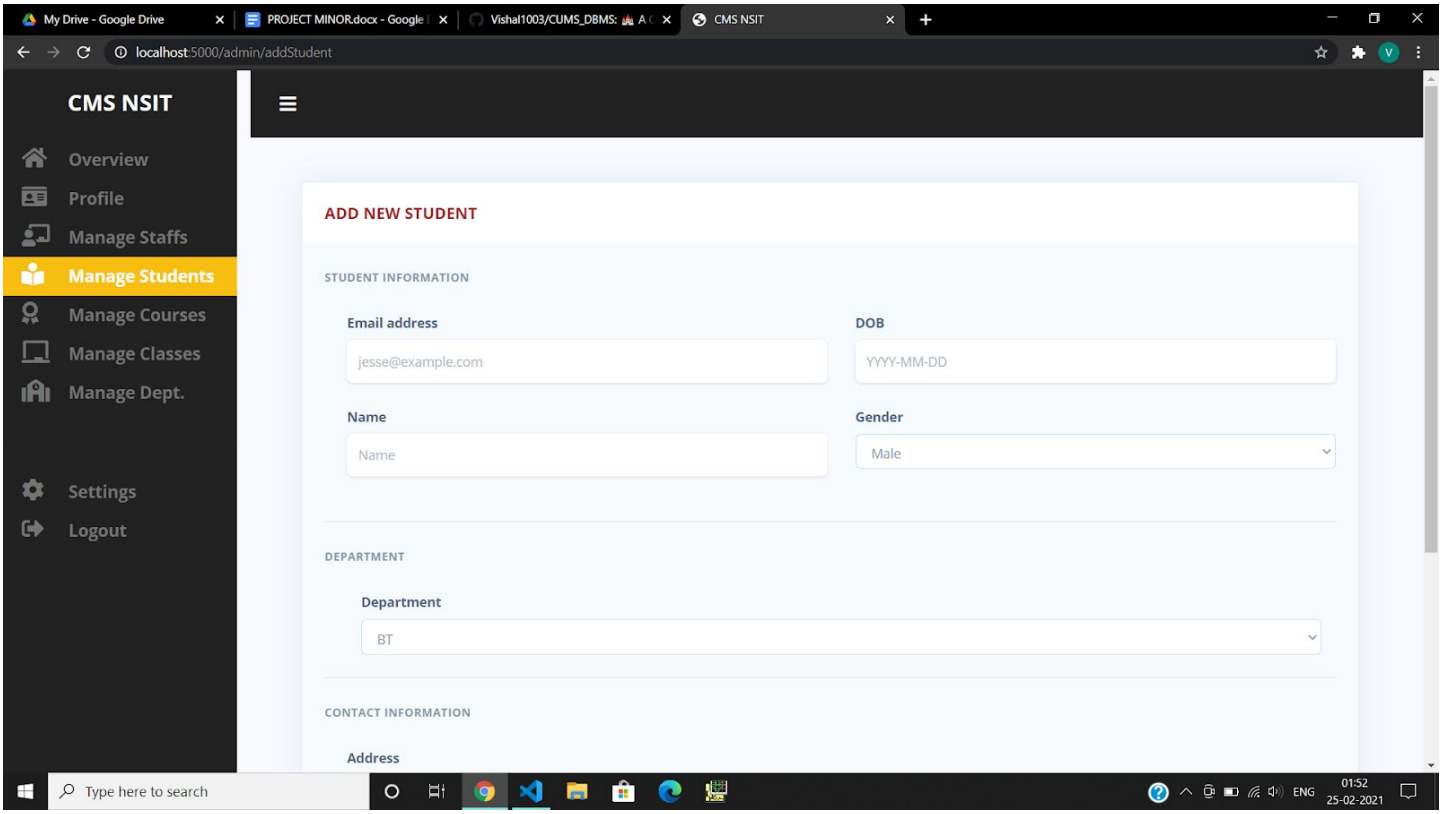
7. ADMIN (Manage Staff)

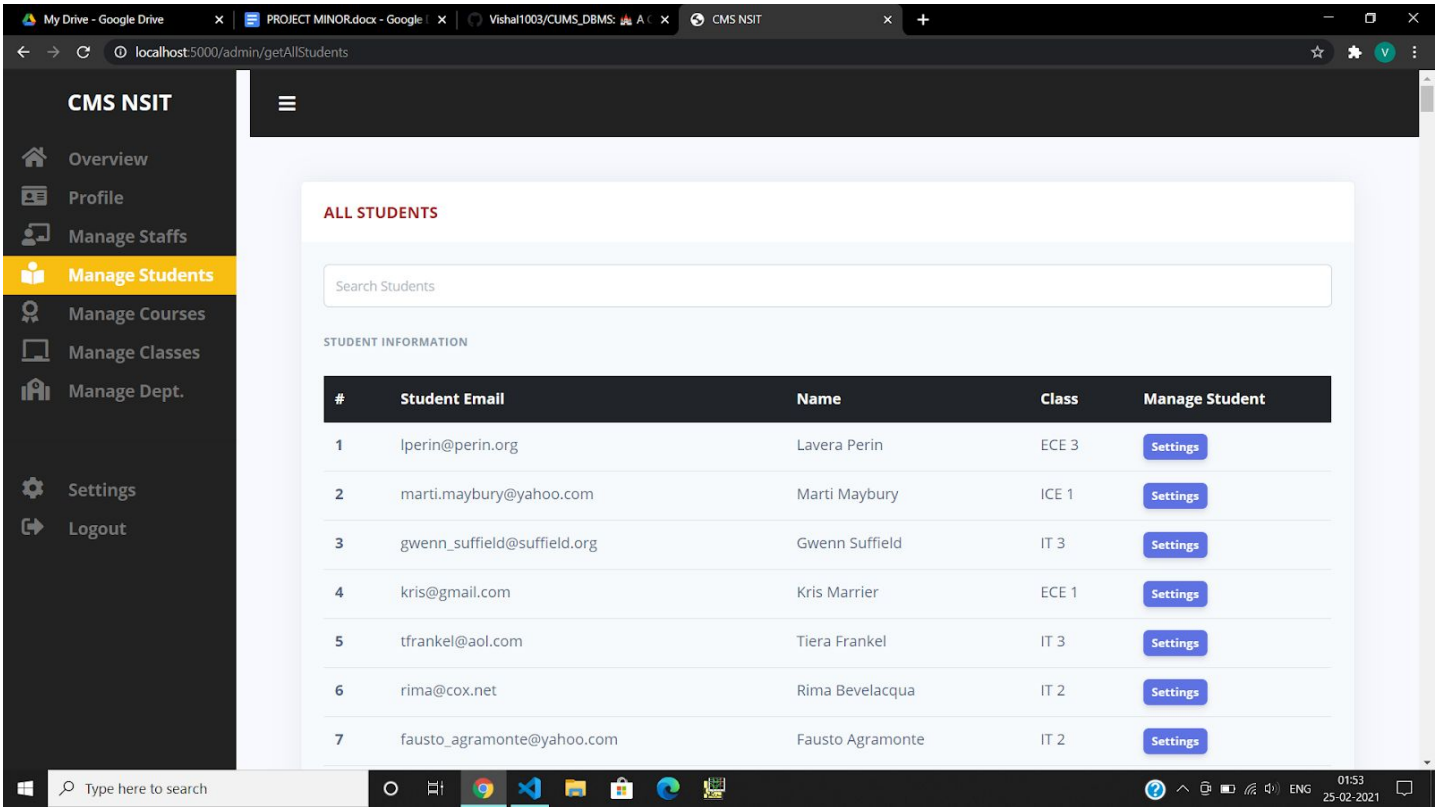




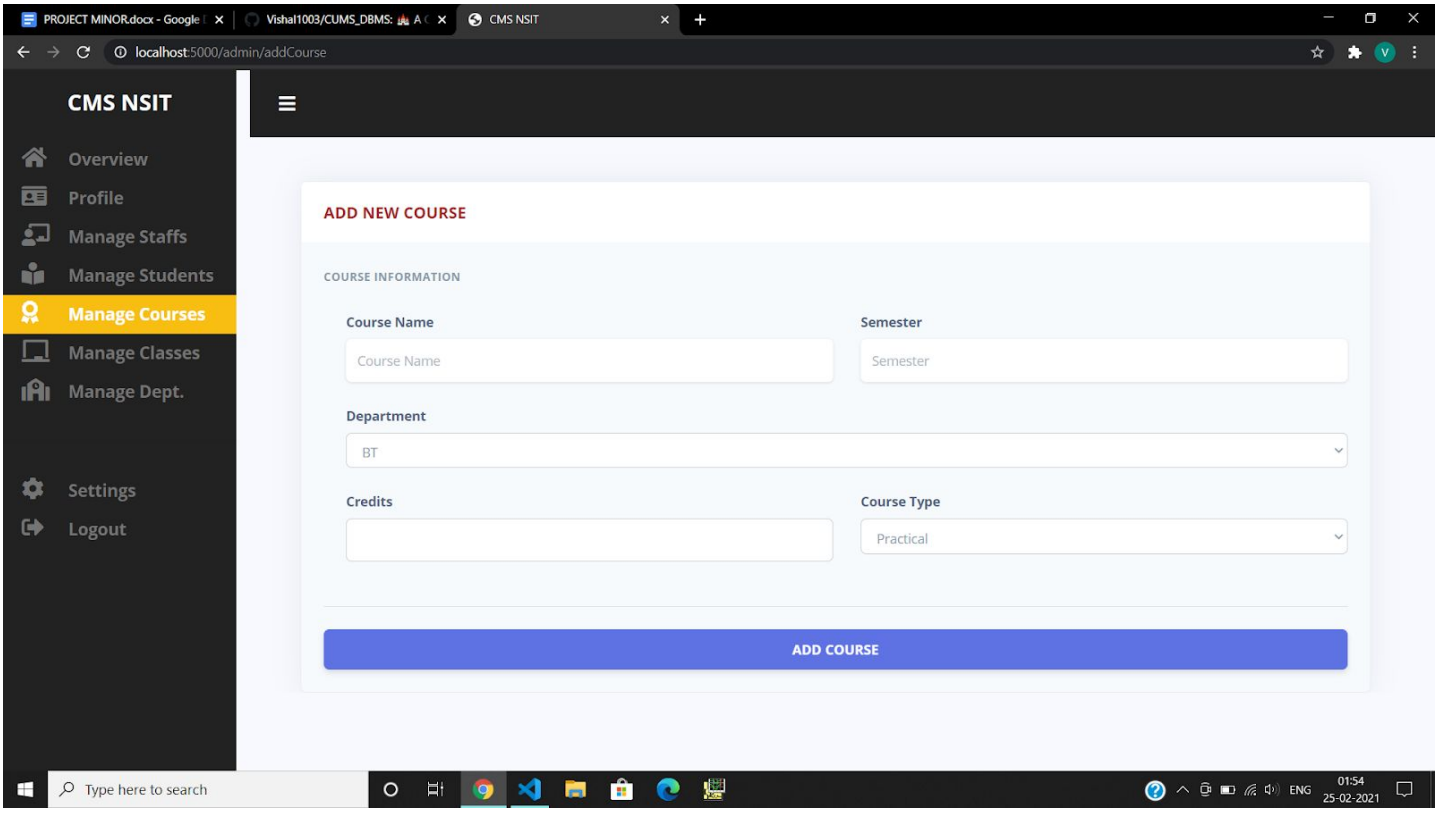


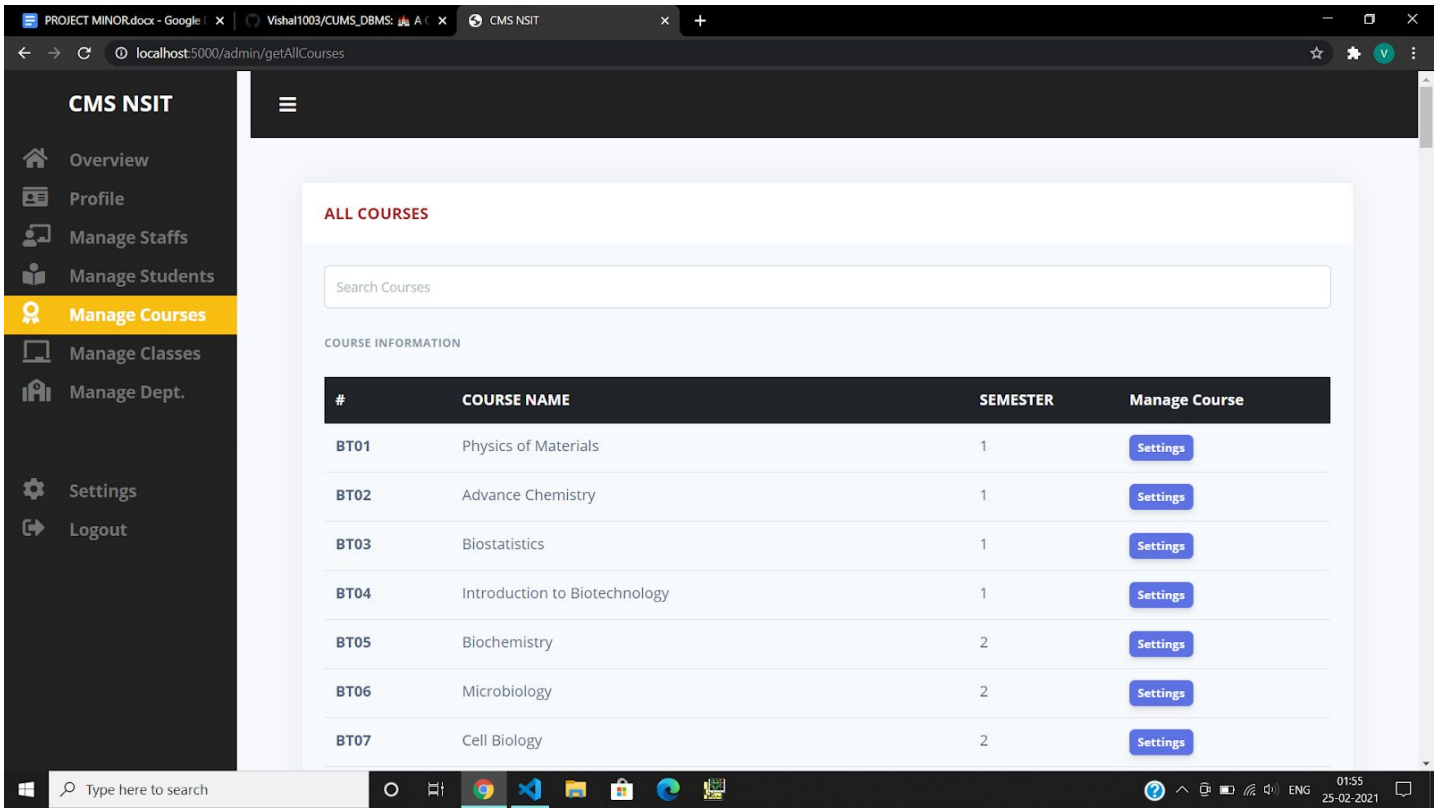
8. ADMIN (Manage Student)



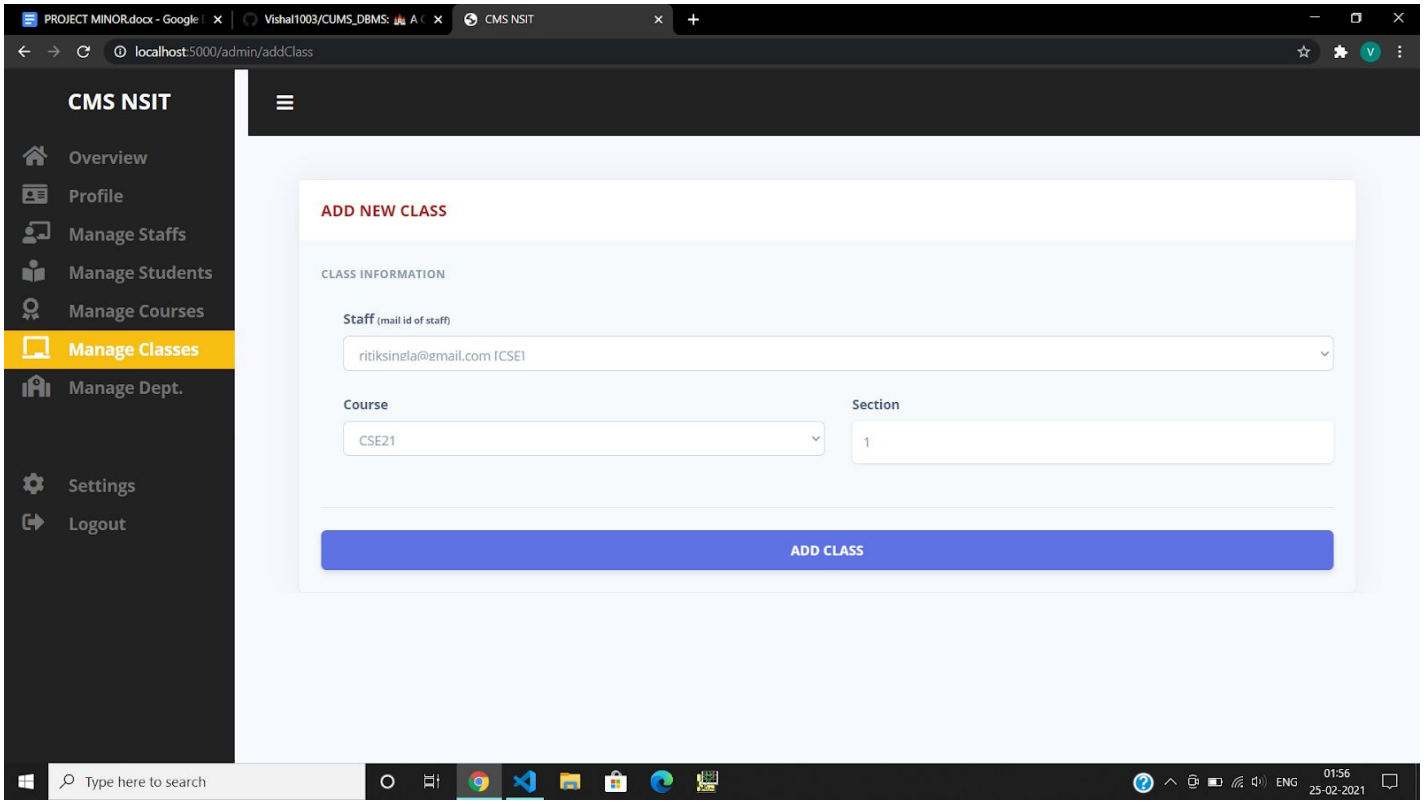


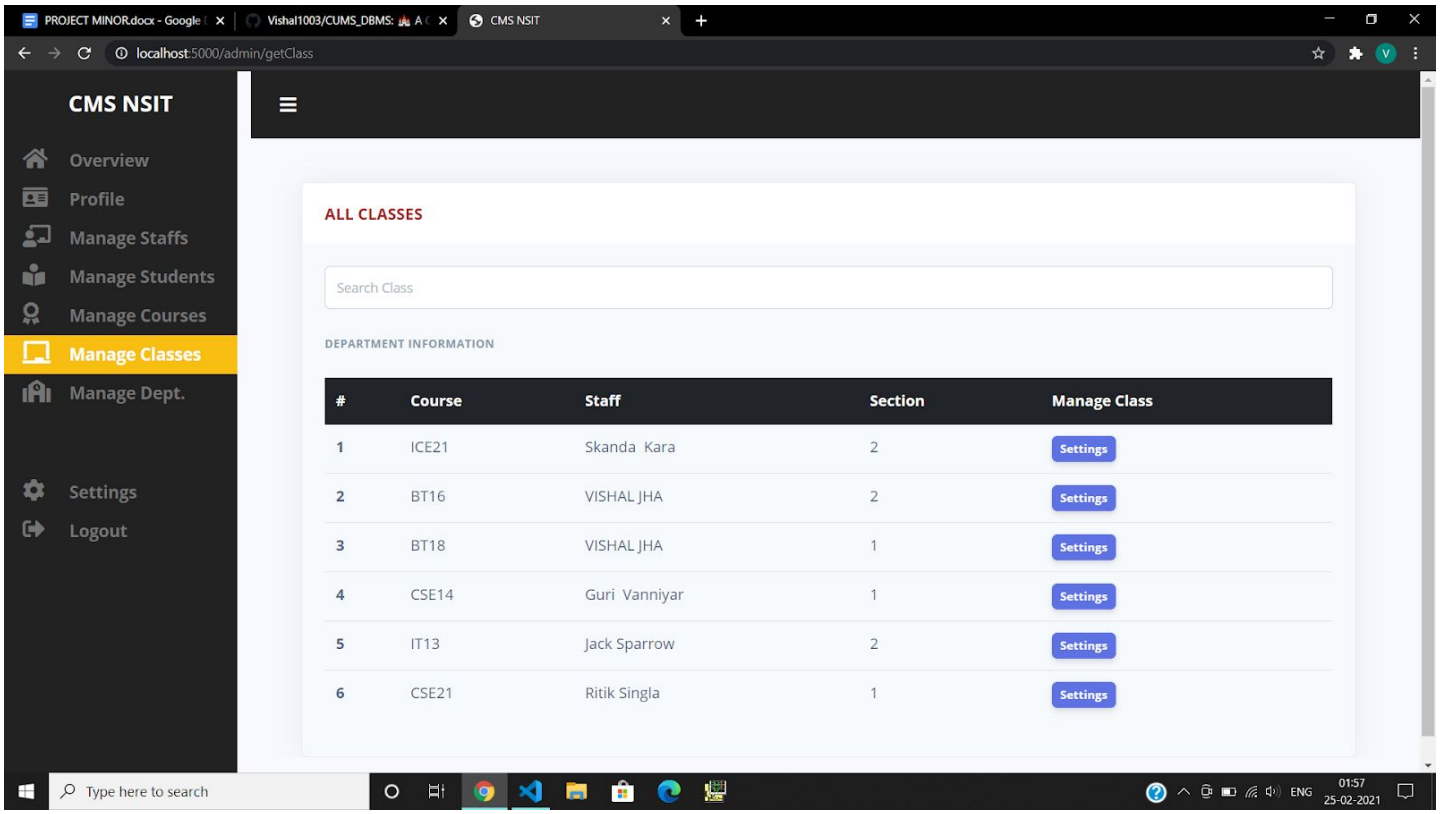
9. ADMIN (Manage Courses)



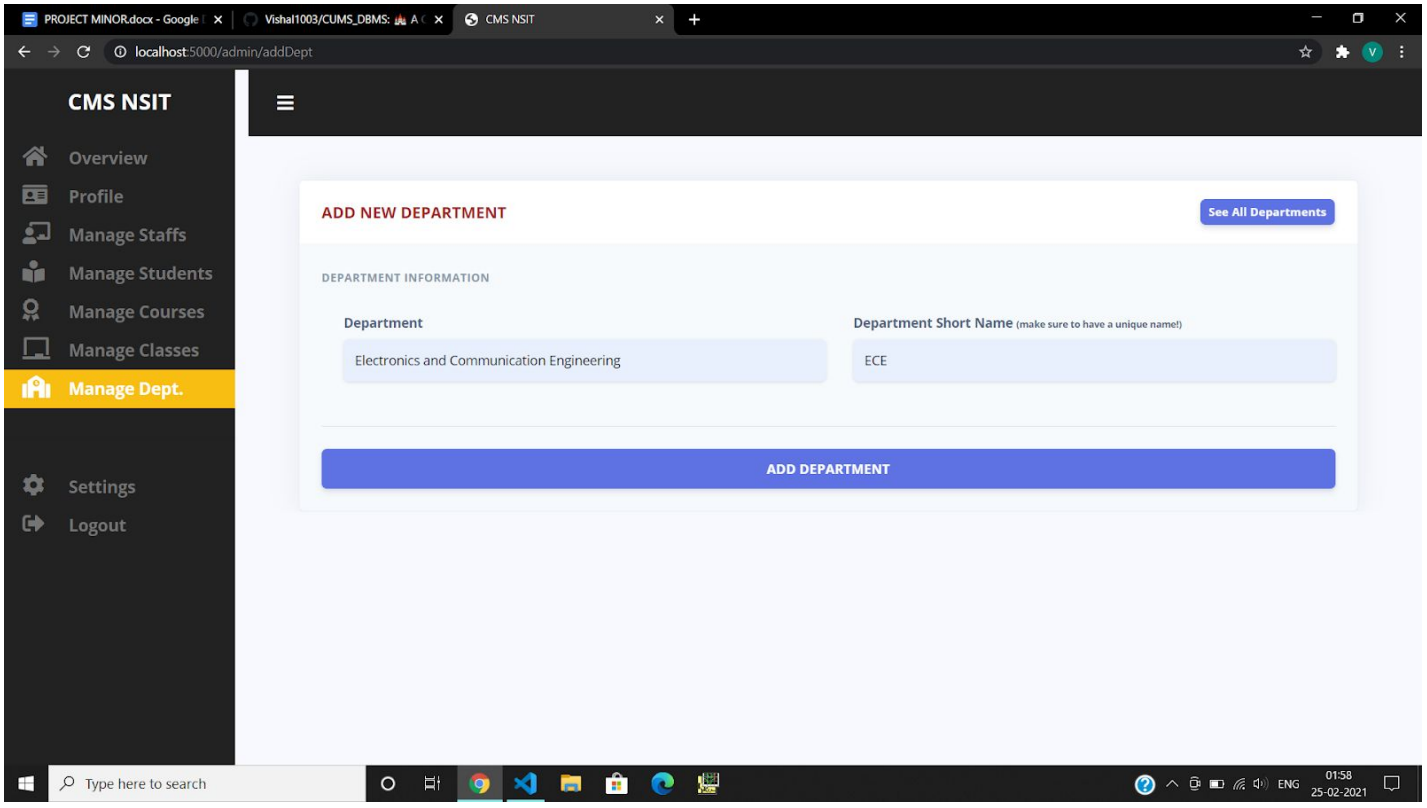


10. ADMIN (Manage Classes)

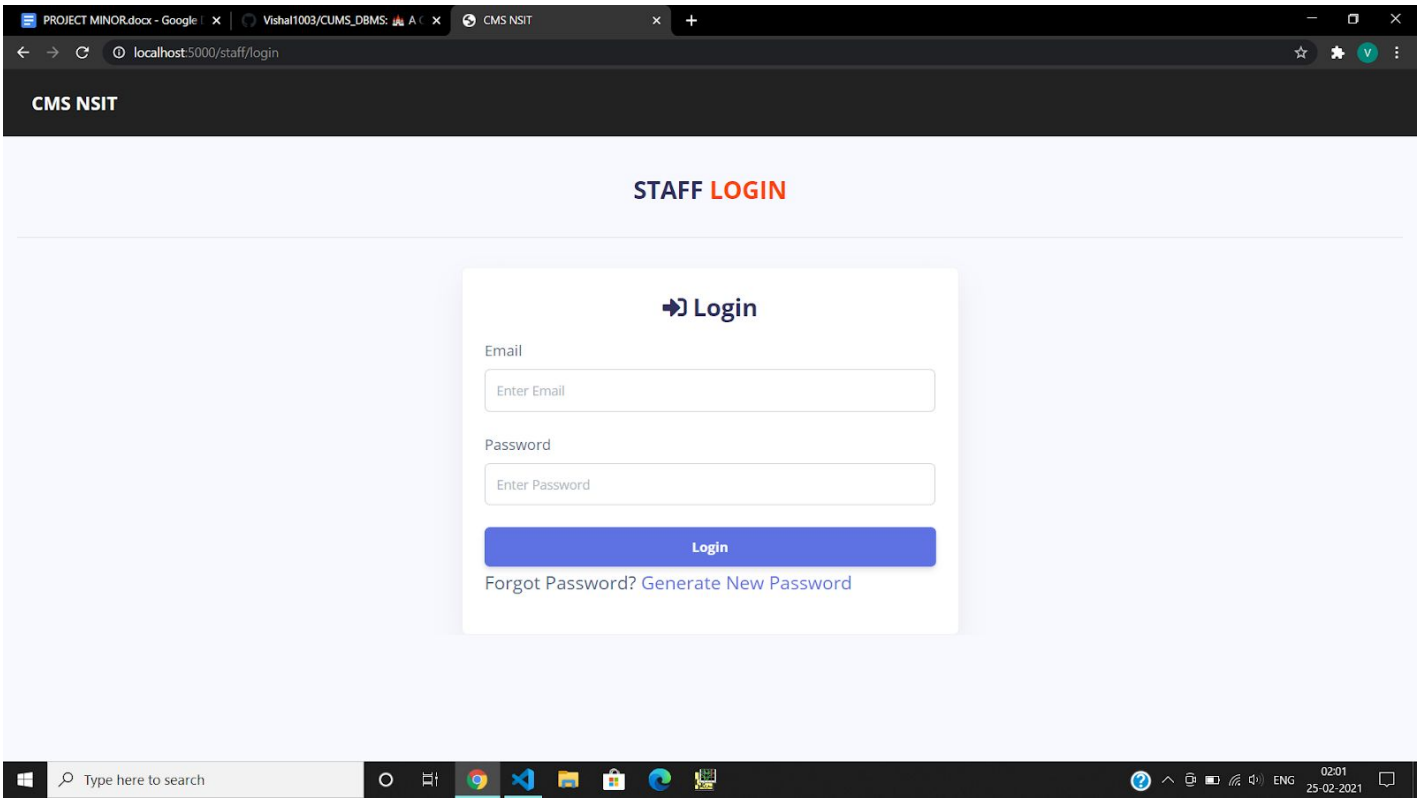




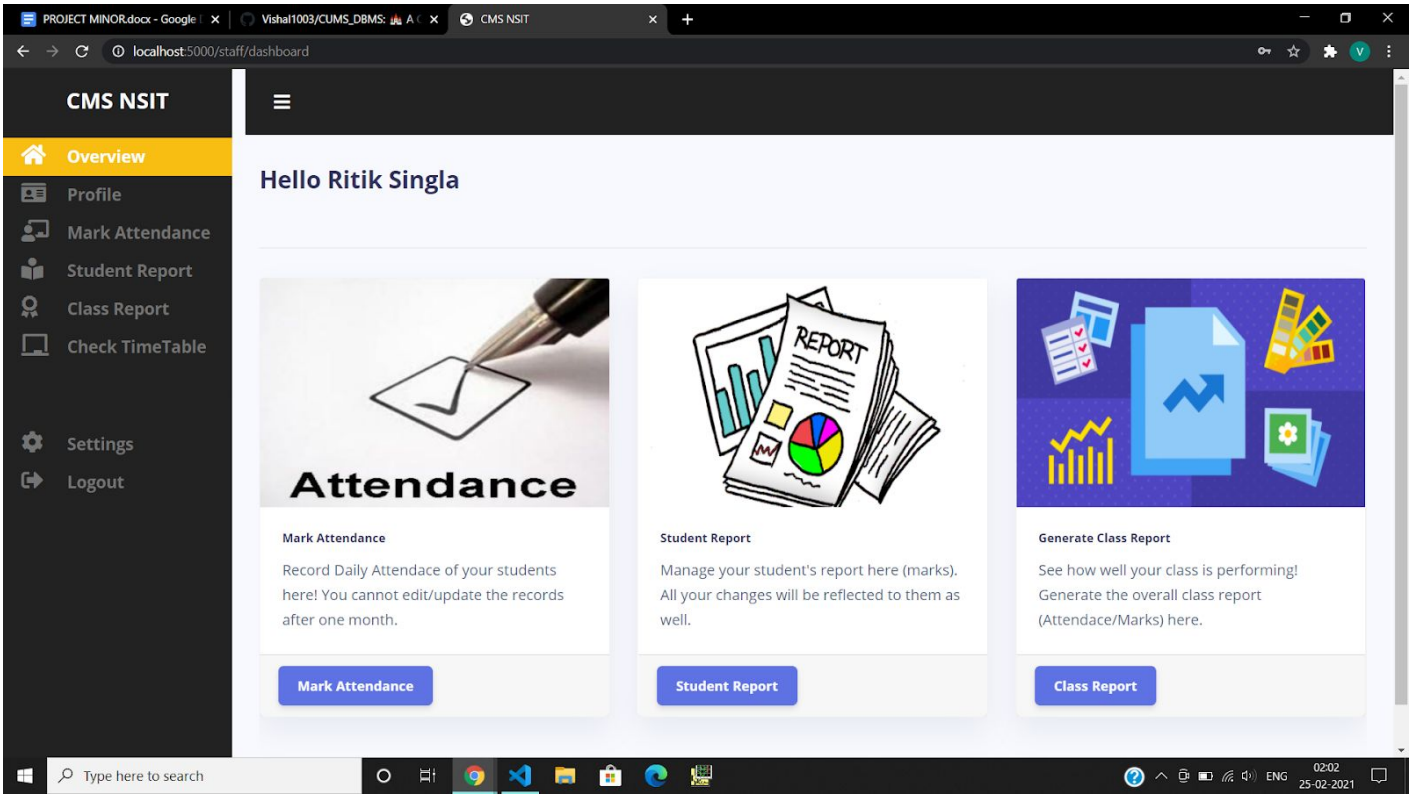
11. ADMIN (Manage Departments)



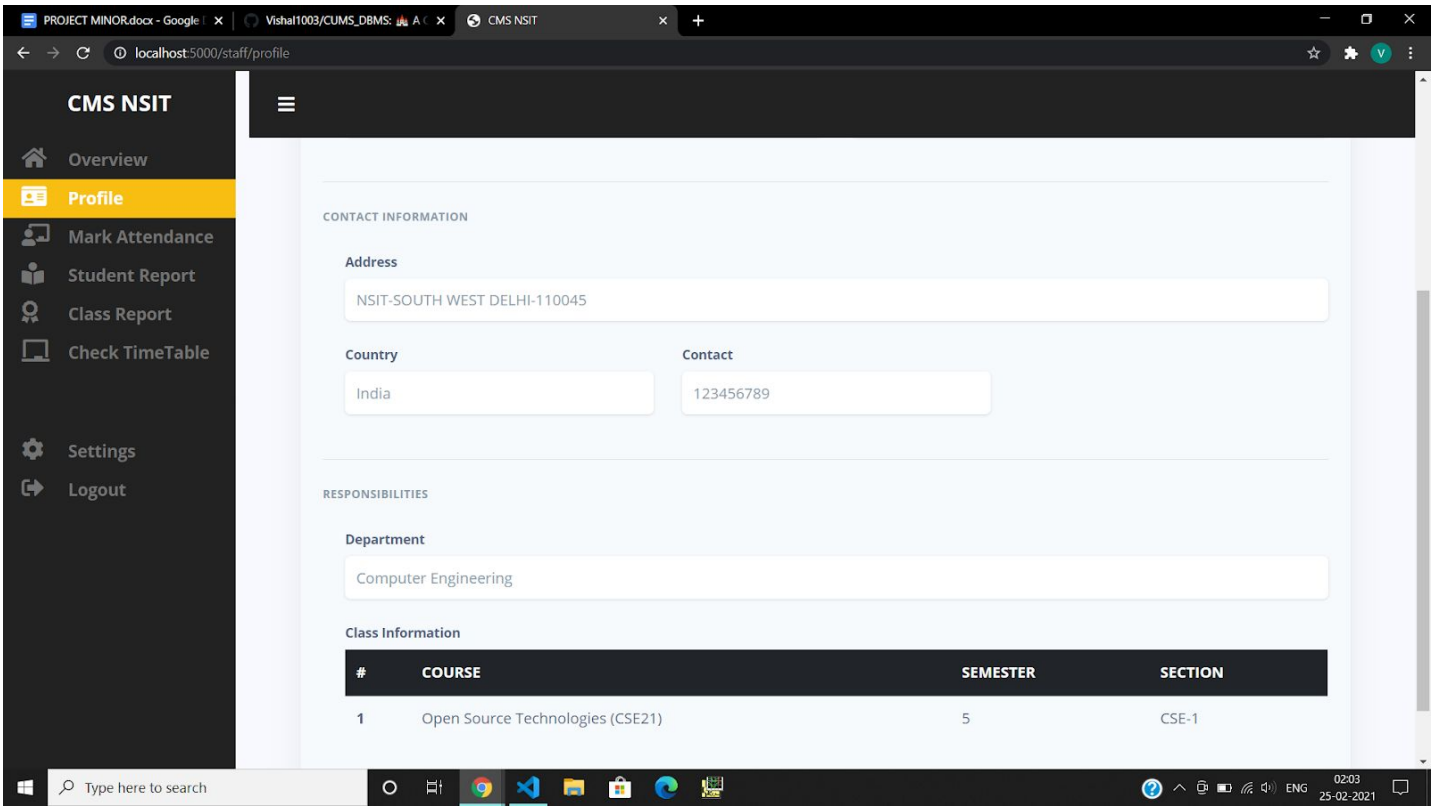
12. STAFF Login



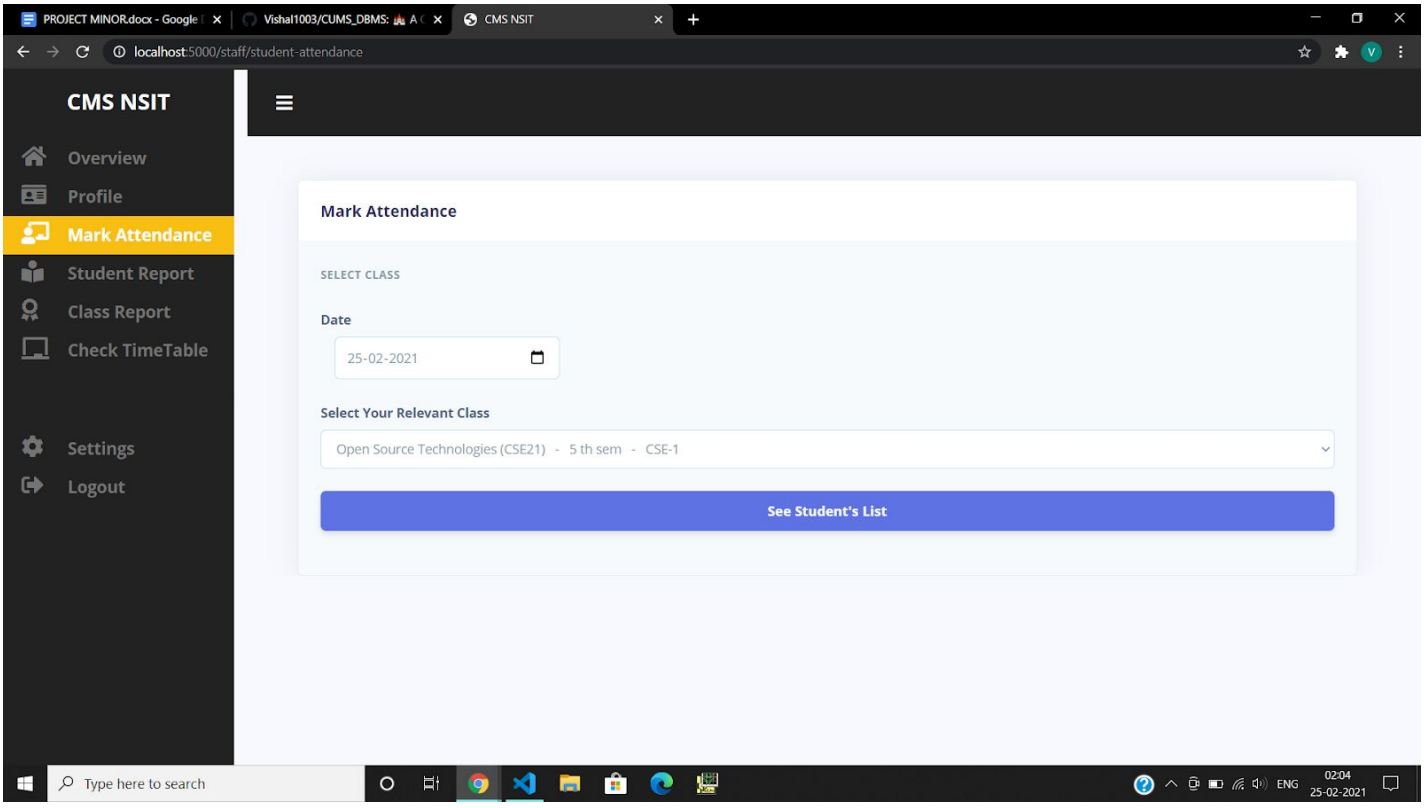
13. STAFF Dashboard

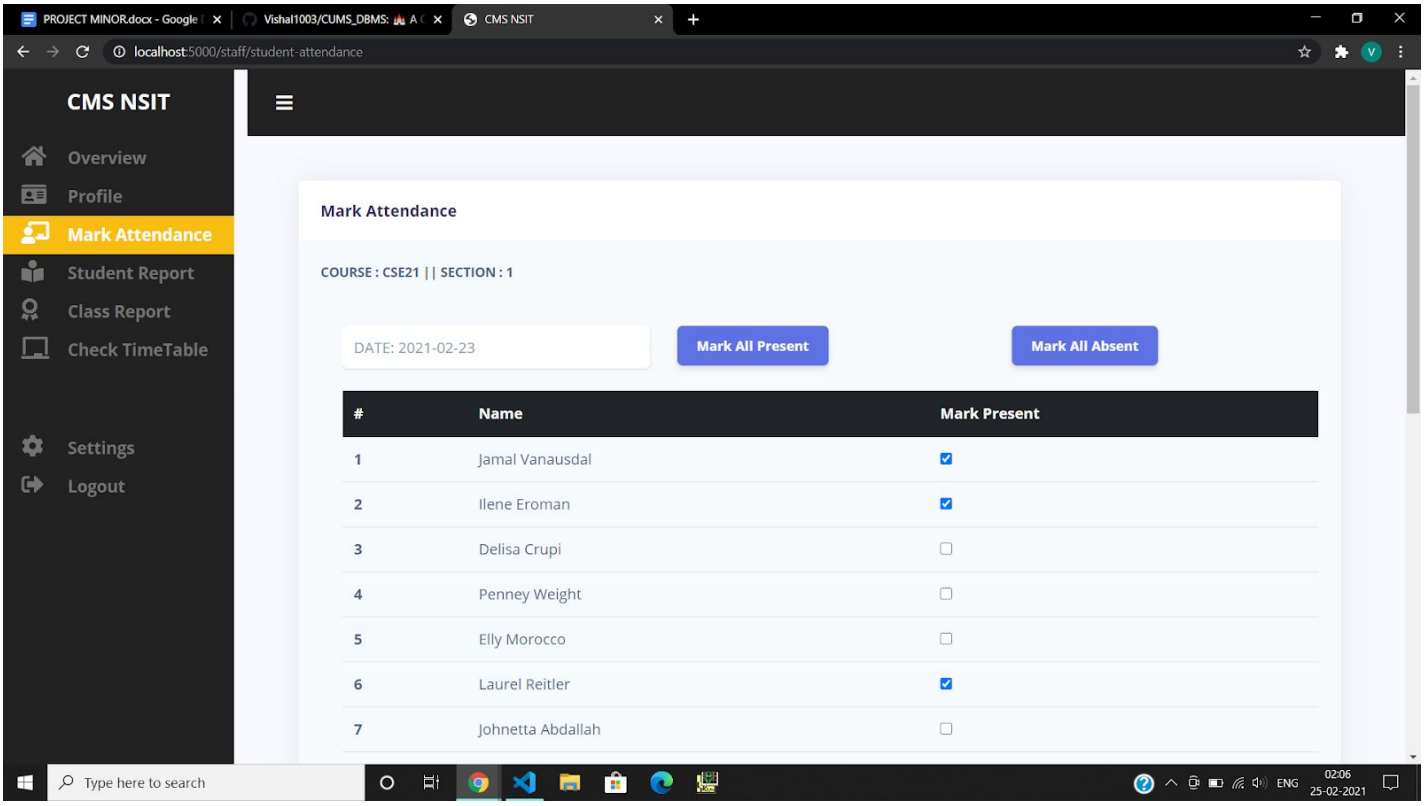


14. STAFF Profile

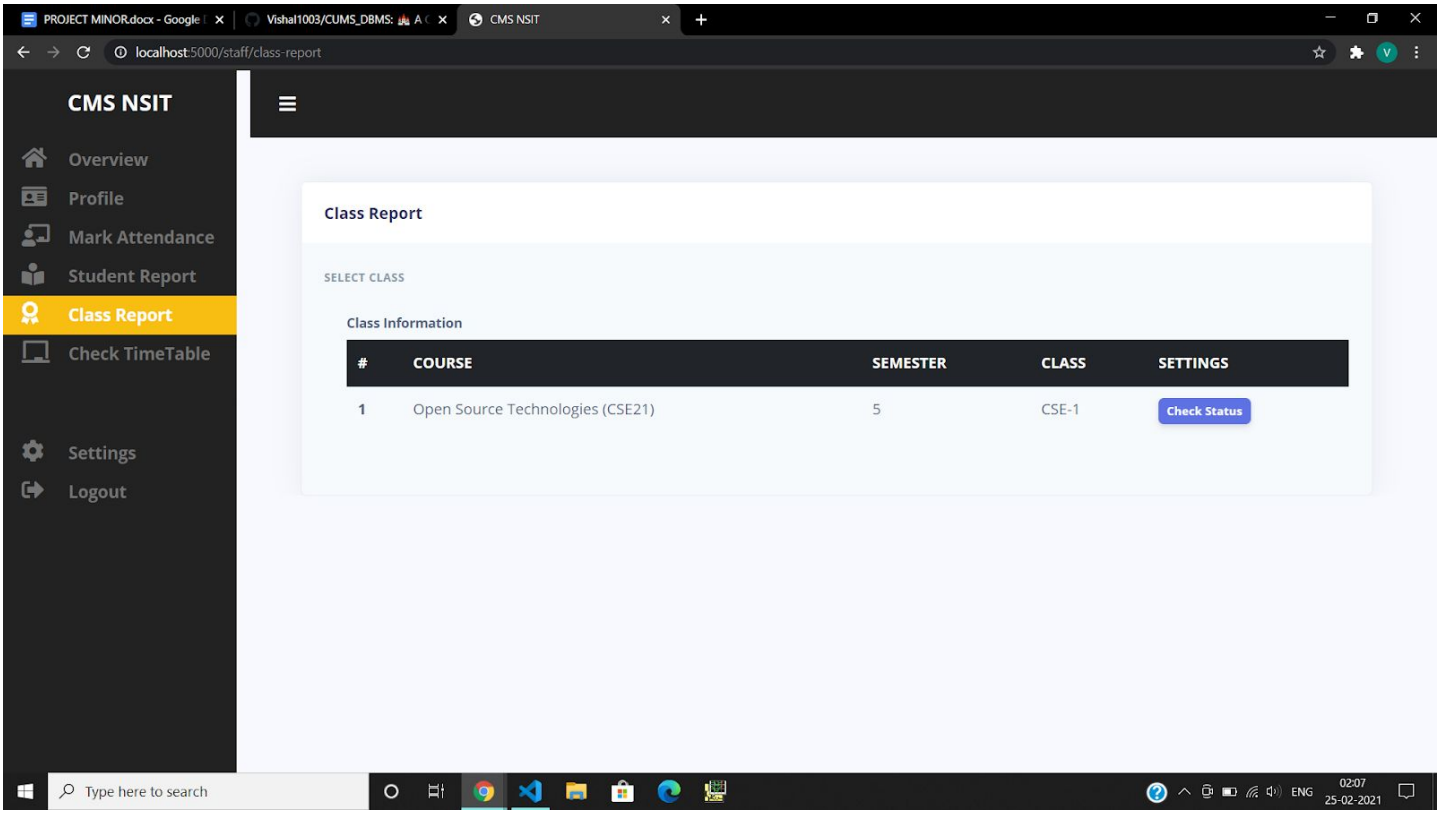


15. STAFF (Mark Attendance)

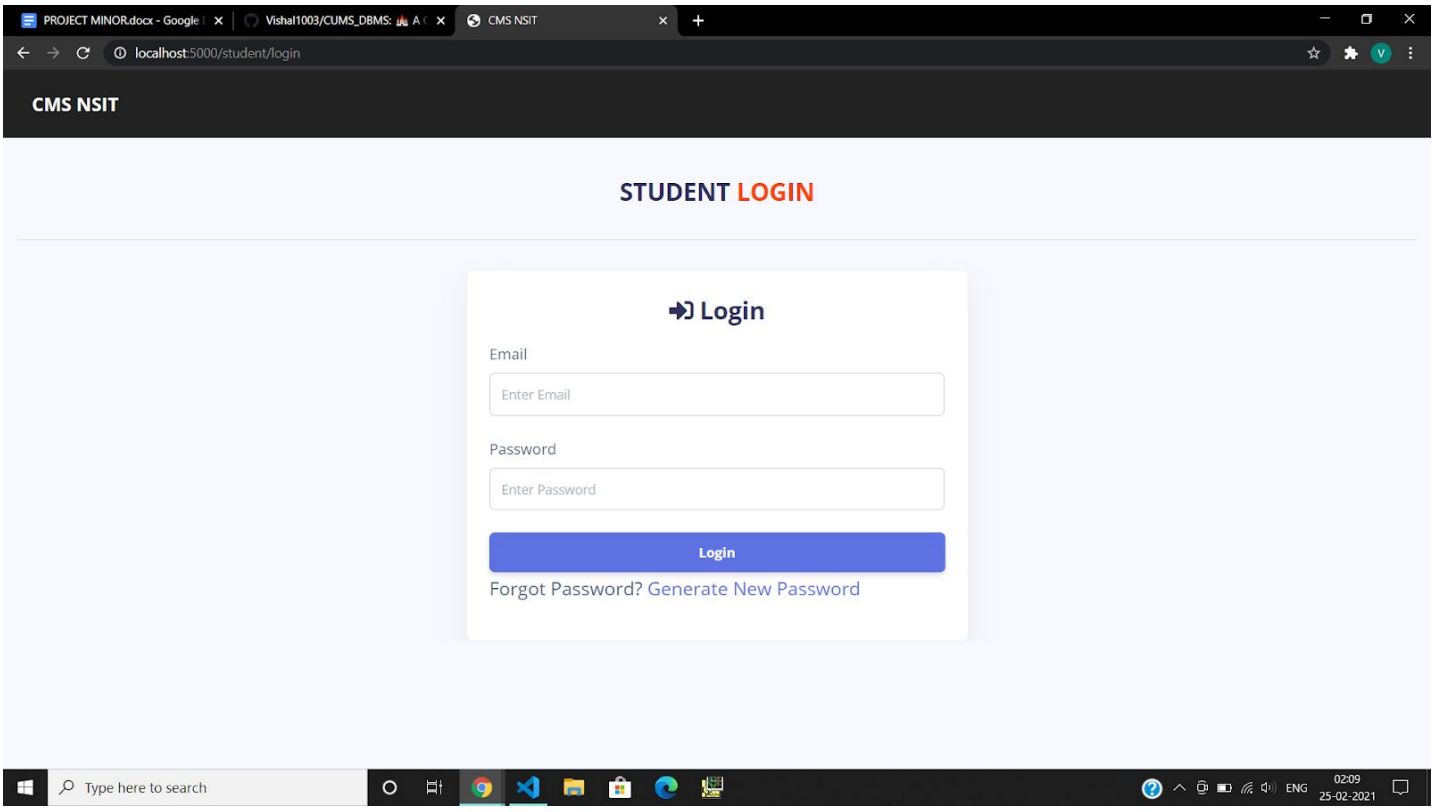




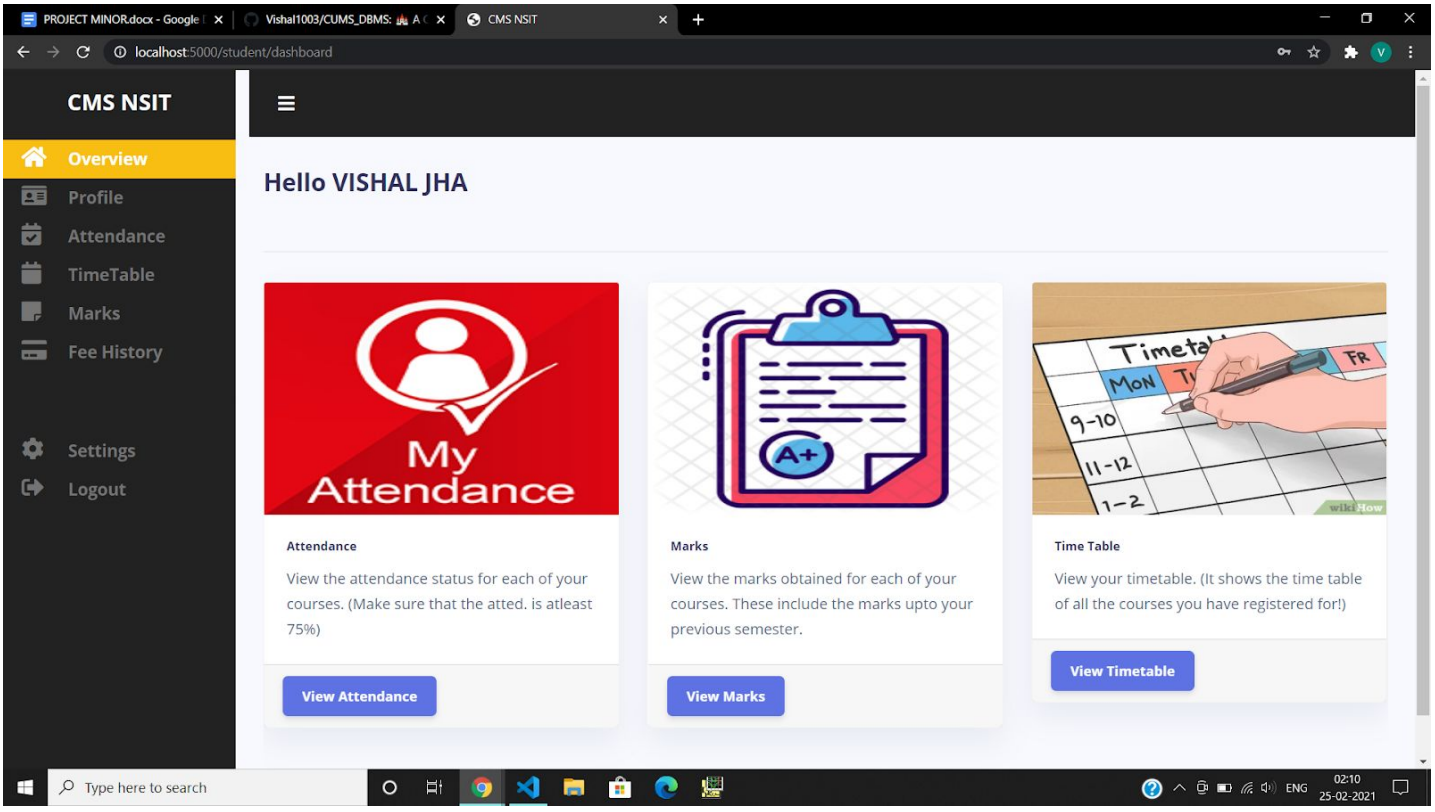
16. STAFF (Reports)



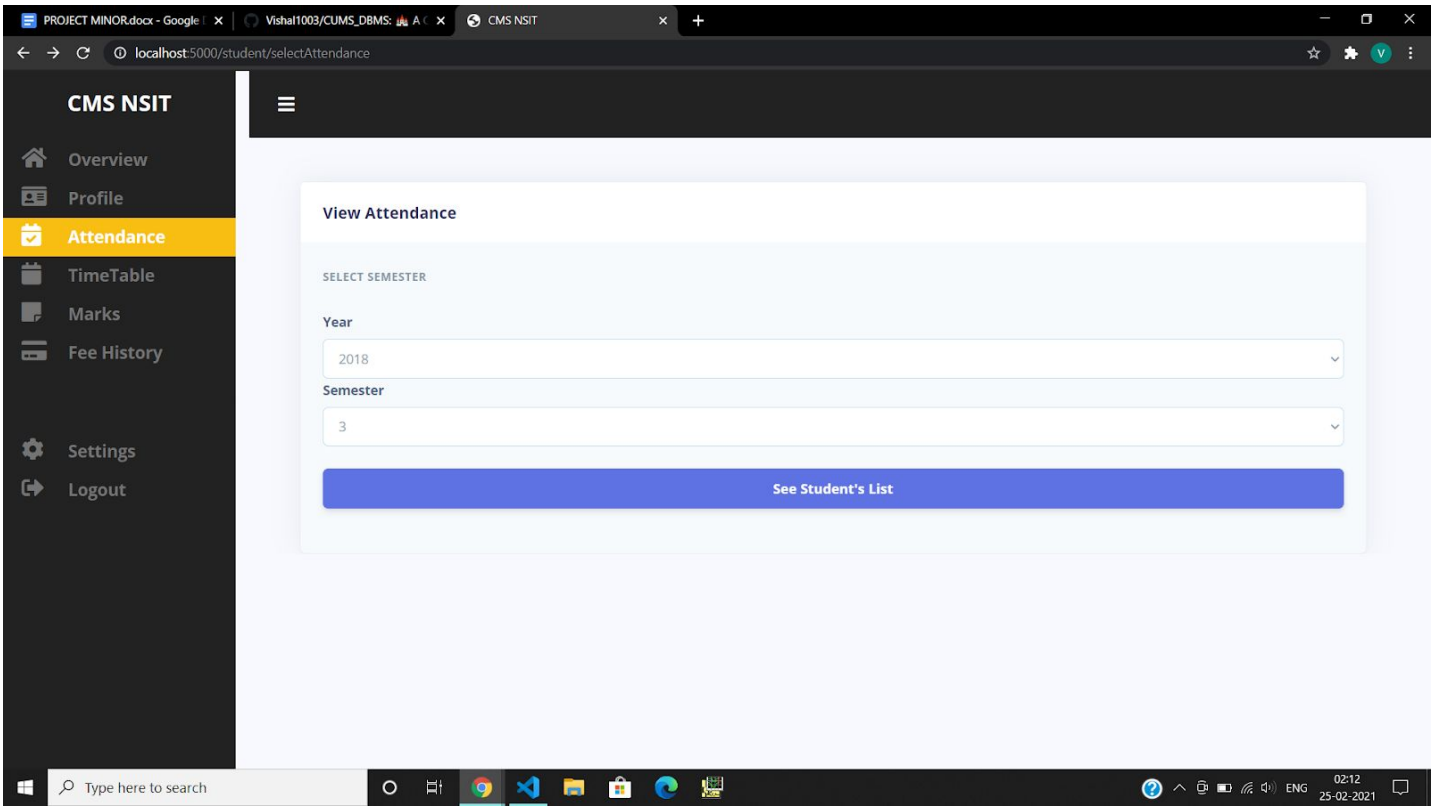
17. Student LOGIN

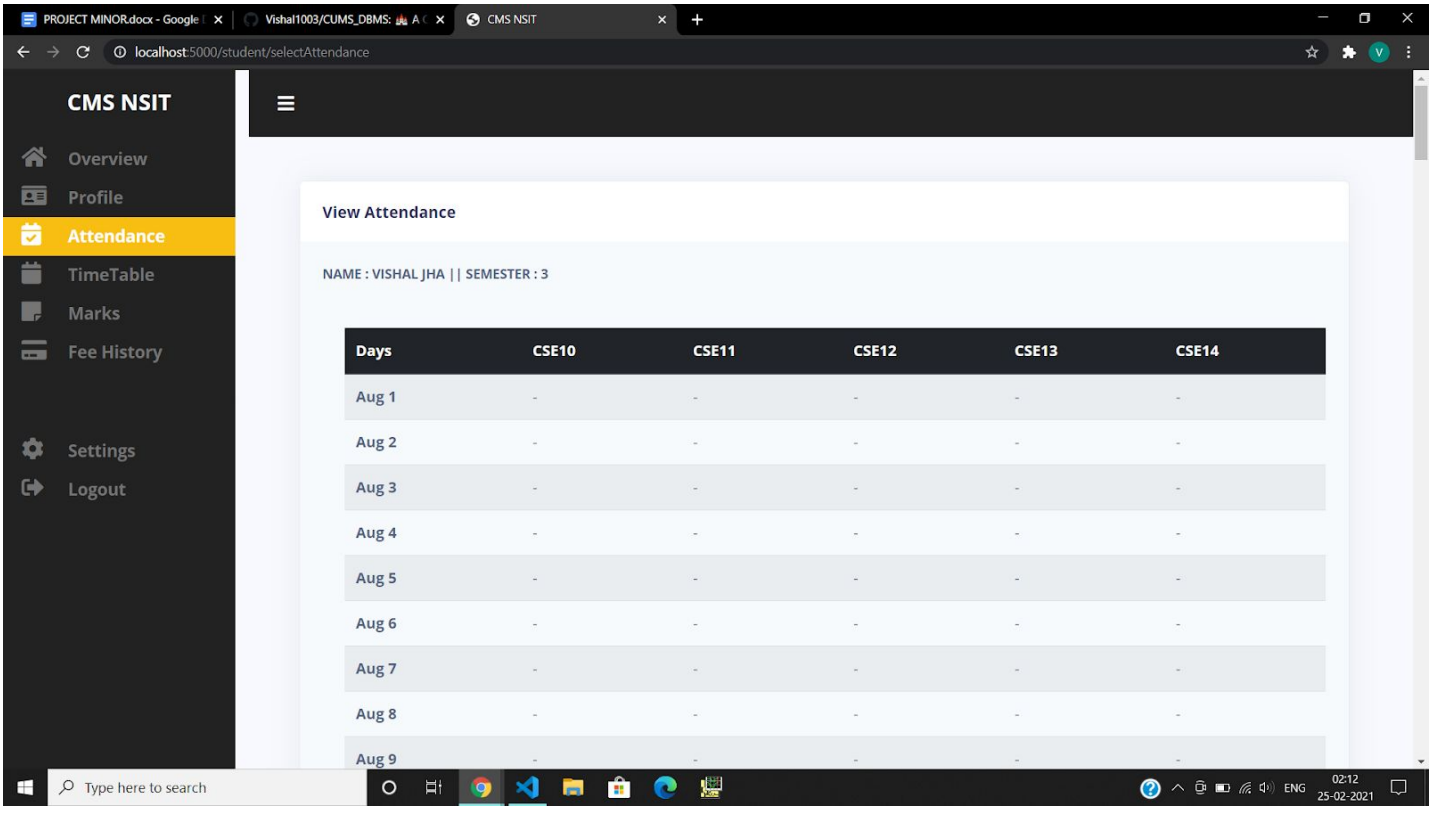


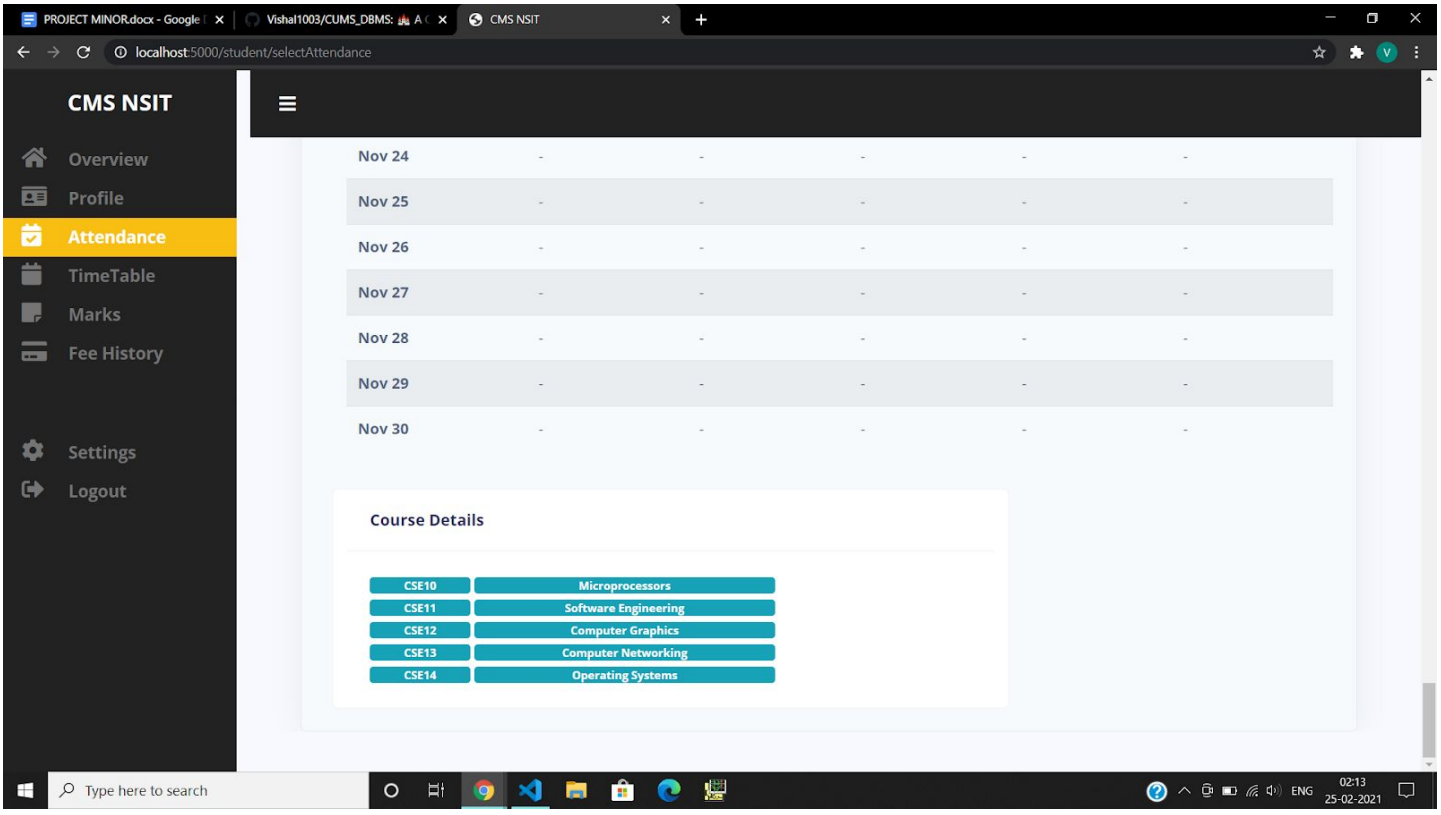
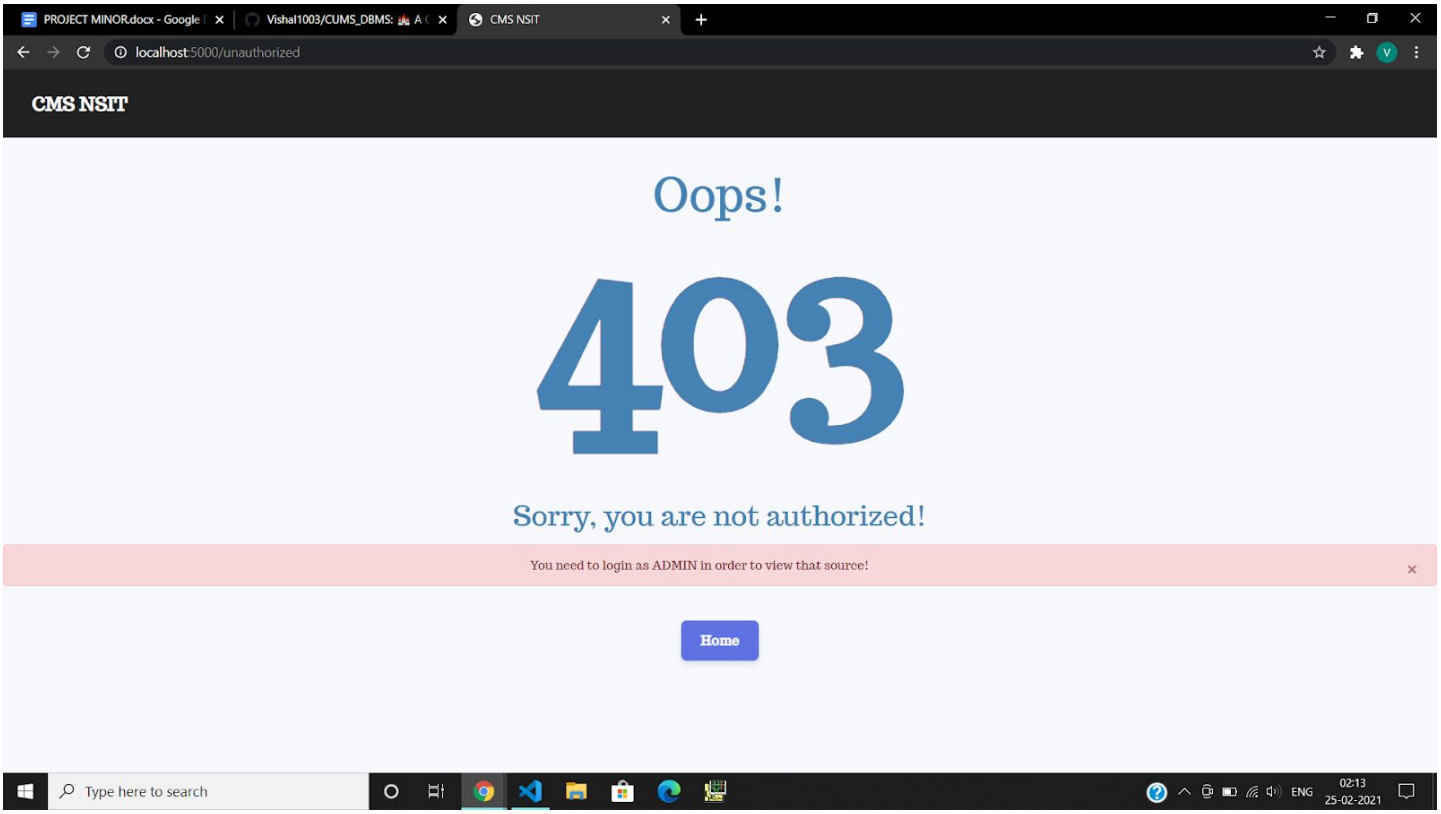
18. Student Dashboard



19. Student (View Attendance)







20. Unauthorized route