|  |
| --- |
| **DAYANANDA SAGAR UNIVERSITY**  Devarakaggalahalli, Harohalli  Kanakapura Road, Ramanagara - 562112, Karnataka, India  A blue and black text  Description automatically generated |

**Bachelor of Technology**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

**DJANGO**

**Mini Project Report**

**(COLLEGE ERP SYSTEM)**

By

**PAVANKUMAR P S - [ENG22CS0117]**

**SHARANU MITTA - [ENG22CS0160]**

**SRUJAN S - [ENG22CS0187]**

**VIKRAM G RATHOD - [ENG22CS0207]**

**Under the supervision of**

**DR. TANVIR HABIB SARDAR**

**DAYANANDA SAGAR UNIVERSITY**

A blue and black text

Description automatically generated

Department of Computer Science & Engineering

Devarakaggalahalli, Harohalli

Kanakapura Road, Ramanagara - 562112, Karnataka, India

CERTIFICATE

This is to certify that the Digital image processing Mini Project work titled **“COLLEGE ERP SYSTEM”** is carried out by **PAVANKUMAR P S (ENG22CS0117), SHARANU MITTA (ENG22CS0160), SRUJAN S (ENG22CS0187) , VIKRAM G RATHOD (ENG22CS0160),** bonafide students of fifth semester of Bachelor of Technology in Computer Science and Engineering at the School of Engineering, Dayananda Sagar University, Bangalore in partial fulfilment for the award of degree in Bachelor of Technology in Computer Science and Engineering, during the year **2024-2025**.

|  |  |
| --- | --- |
| **DR. TANVIR HABIB SARDAR** | **DR. GIRISHA G S** |
| Assistant Professor  Dept. of CS&E,  School of Engineering  Dayananda Sagar University  Date:-24-12-2024 | Chairman CSE  School of Engineering  Dayananda Sagar University  Date:24-12-2024 |

DECLARATION

We, **PAVANKUMAR P S (ENG22CS0117), SHARANU MITTA (ENG22CS0160), SRUJAN S (ENG22CS0187) VIKRAM G RATHOD (ENG22CS0207)** are students of fifth semester B. Tech in **Computer Science and Engineering**, at School of Engineering, **Dayananda Sagar University**, hereby declare that the Mini Project titled **“COLLEGE ERP SYSTEM”** has been carried out by us and submitted in partial fulfilment for the award of degree in **Bachelor of Technology in Computer Science and Engineering** during the academic year **2024‑2025.**

|  |  |
| --- | --- |
| Students |  |
| Name1: PAVANKUMAR P S  USN: ENG22CS0117 |  |
| Name2: SHARANU MITTA  USN: ENG22CS0160  Name3: SRUJAN S  USN: ENG22CS0187 |  |
| Name3: VIKRAM G RATHOD  USN: ENG22CS0207 |  |
|  |  |
| Place: BENGALURU  Date :24-12-2024 |  |

**ACKNOWLEDGEMENT**

*It is a great pleasure for us to acknowledge the assistance and support of many individuals who have been responsible for the successful completion of Digital Image Processing mini project work.*

*First, we take this opportunity to express our sincere gratitude to School of Engineering & Technology, Dayananda Sagar University for providing us with a great opportunity to pursue our Bachelor’s degree in this institution.*

*We would like to thank* ***Dr. Udaya Kumar Reddy K R, Dean, School of Engineering & Technology, Dayananda Sagar University*** *for his constant encouragement and expert advice.*

*It is a matter of immense pleasure to express our sincere thanks to* ***Dr. Girisha G S,******Department Chairman****,* ***Computer Science and Engineering, Dayananda Sagar University,*** *for providing right academic guidance that made our task possible.*

*We would like to thank our guide* ***Dr. Tanvir Habib Sardar****,* ***Assistant Professor****,* ***Dept. of Computer Science and Engineering****,* ***Dayananda Sagar University****, for sparing his/her valuable time to extend help in every step of our project work, which paved the way for smooth progress and fruitful culmination of the project.*

*We are also grateful to our family and friends who provided us with every requirement throughout the course.*

*We would like to thank one and all who directly or indirectly helped us in the mini project work.*

**TABLE OF CONTENTS**

Page

CHAPTER 1 INTRODUCTION……….................................................................... 06

CHAPTER 2 ABSTRACT…………….…….......................................................... 07

CHAPTER 3 REQUIREMENTS............................................................................. 08

CHAPTER 4 DESIGN FLOW………....................................................................... 10

CHAPTER 5 METHODOLOGY……………............................................................. 11

CHAPTER 6 APPLICATIONS……………………………………………………… 13

CHAPTER 7 CONCLUSION……………………………………………………… 15

CHAPTER 8 REFERENCES……………………………………………………… 16

CHAPTER1

INTRODUCTION

The objective of College Information Management System is to allow the administrator of any organization the ability to edit and find out the personal details of a student and allows the student to keep up to date his profile. It’ll also facilitate keeping all the records of students, such as their id, name, mailing address, phone number, DOB etc. So all the information about a student will be available in a few seconds. Overall, it’ll make Student Information an easier job for the administrator and the student of any organization.

The main purpose of this project is to illustrate the requirements of the project College Information Management System and is intended to help any organization to maintain and manage personal data. It is a comprehensive project developed from the ground up to fulfill the needs of colleges as they guide their students. This integrated information management system connects daily operations in the college environment ranging from Attendance management to communicational means among students and teachers. This reduces data error and ensures that information is always up-to-date throughout the college. It provides a single source of data repository for streamlining your processes and for all reporting purposes. It has a simple user interface and is intuitive. This insures that the users spend less time in learning the system and hence, increase their productivity. Efficient security features provide data privacy and hence, increase their productivity.

## Introduction to problem domain

As we know that, a college consists of different departments, such as course departments, fees management, library, event management etc. Nowadays applications and uses of information technologies is increased as compared to before, each of these individual departments has its own computer system to do their own functionalities. By having one main system they can interact with each other from their respected system by having valid user id and password.

## Aim of the problem

The objective of College Information Management System is to allow the ad- ministrator of any organization the ability to edit and find out the personal details of a student and allows the student to keep up to date his profile. It’ll also facilitate keeping all the records of students, such as their id, name, mailing address, phone number, DOB etc. So all the information about a student will be available in a few seconds. Overall, it’ll make Student Information an easier job for the administrator and the student of any organization.

The main purpose of this project is to illustrate the requirements of the project College Information Management System and is intended to help any organization to maintain and manage personal data. It is a comprehensive project developed from the ground up to fulfill the needs of colleges as they guide their students. This integrated information management system connects daily operations in the college environment ranging from Attendance management to communicational means among students and teachers. This reduces data error and ensures that information is always up-to-date throughout the college. It provides a single source of data repository for streamlining your processes and for all reporting purposes. It has a simple user interface and is intuitive. This insures that the users spend less time in learning the system and hence, increase their productivity. Efficient security features provide data privacy and hence, increase their productivity.

## Time schedule for completion of the project work

The Project schedule activities will consist of following:

1. Forming The Team
2. Selecting The Project Title
3. System Requirement Collection
4. System Design
5. Acquiring the required resources
6. Coding
7. Testing of the Application
8. Deployment

CHAPTER 2

ABSTRACT

Managing the diverse and complex operations of an educational institution requires an efficient and integrated system. The College ERP (Enterprise Resource Planning) System is a comprehensive solution designed to streamline administrative, academic, and management processes within a college environment.

This project, developed using Django, a high-level Python web framework, offers a centralized platform for handling key functions such as student enrolment , attendance tracking, faculty management, course scheduling, and examination results. The system eliminates the need for disparate tools by integrating all essential modules into a single application, thereby improving operational efficiency and reducing redundancies.

The project employs Django's Model-View-Template (MVT) architecture to ensure a structured, scalable, and secure application. Key features include:

* **Student Module**: Registration, academic records, and attendance tracking.
* **Faculty Module**: Assignment management, timetable scheduling, and performance reviews.
* **Administrative Tools**: Fee management, announcements, and event planning.
* **Examination Management**: Online assessments and result generation.

By automating routine tasks and providing real-time data access, the system enhances decision-making and minimizes manual errors. The project also emphasizes user-friendliness with an intuitive interface and role-based access control to ensure security and confidentiality.

This College ERP system serves as a robust and scalable solution, adaptable to the evolving needs of modern educational institutions, making it an invaluable tool for effective college management.

CHAPTER 3

REQUIREMENTS

#### **1. Functional Requirements (What the system should do)**

* **User Roles**:
* The system should have three main user roles:
  + **Admin**: Manages the system, creates courses, assigns faculty, and manages students.
  + **Faculty**: Manages student attendance and grades for the courses they teach.
  + **Students**: Can view their own attendance, grades, and enrolled courses.

**Explanation**:

Defining user roles ensures that each user only accesses features relevant to them. For example, students cannot change grades, and faculty cannot manage user accounts.

* **Student Features**:
  + Register and log in to their accounts.
  + View their assigned courses, attendance records, and grades.

**Explanation**:

These features allow students to monitor their academic progress in real-time.

* **Faculty Features**:
  + Mark student attendance for their assigned courses.
  + Add or update grades for students.

**Explanation**:

Faculty can focus on their academic responsibilities without handling unrelated administrative tasks.

* **Admin Features**:
  + Manage users (students and faculty) by creating, updating, or deleting accounts.
  + Create and assign courses to faculty members.

**Explanation**:

Admin users oversee the system and ensure it functions smoothly by managing critical data like user accounts and course assignments.

#### **2. Technical Requirements (Tools and technologies used)**

* **Frontend**:
  + Basic HTML, CSS, and Bootstrap for building a simple and responsive interface.

**Explanation**:

HTML and CSS handle structure and styling, while Bootstrap ensures the system looks good on devices like phones, tablets, and desktops.

* **Backend**:
  + Django framework (Python) for handling server-side logic.

**Explanation**:

Django simplifies development by providing built-in tools for authentication, database management, and routing.

* **Database**:
  + SQLite for storing data like user accounts, attendance records, and grades.

**Explanation**:

SQLite is easy to set up and ideal for small to medium-sized projects.

#### **3. Hardware Requirements (Minimum setup needed)**

* **Development Machine**:
  + A computer with Python 3 installed.
  + At least 2 GB of RAM and 5 GB of free disk space.

**Explanation**:

These basic requirements are sufficient for developing and running the project locally.

#### **4. Non-Functional Requirements (How the system should behave)**

* **Usability**:
  + The interface should be simple and easy to navigate for all users (admin, faculty, and students).

**Explanation**:

A user-friendly system ensures that users can perform their tasks without technical expertise.

* **Security**:
  + All users must log in using a secure username and password.

**Explanation**:

Authentication prevents unauthorized access to sensitive data like grades and attendance records.

* **Portability**:
  + The system should run on any web browser, making it accessible across devices.

**Explanation**:

Web-based accessibility allows users to use the system without installing additional software.

CHAPTER 4

DESIGN FLOW

# System Design

Various Design concepts and processes were applied to this project. Following concepts like separation of concerns, the software is divided into individual modules that are functionally independent and incorporates information hiding. The software is divided into 3 modules which are students, teachers and administrators. We shall look at each module in detail.

## Student

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.

### • Student information

Each student can view only their own personal information. This includes their personal details like name, phone no, address etc. Also, they can view the courses they are enrolled in and the attendance, marks of each of those.

### • Attendance information

Attendance for each course will be displayed. This includes the number of attended classes and the attendance percentage. If the attendance percentage if below a specified threshold, say 75%, It will be marked in red otherwise it be in green. There will also be a day wise attendance view for each course which shows the date and status. This will be presented in a calender format.

### • Marks information

There will be 5 events and 1 semester end examination for each course. The marks for each of these will be provided in the ERP system.

### • Notifications and events

This section is common to all students. Notification are messages from the admin such as declaration of holidays, test time-table etc. The events and their details are specified here.

## Teacher

Each teacher belongs to a department and are assigned to classes with a course. Teachers will also have a username and password to login. The different views for teachers are described below.

### • Information

The teachers will have access to information regarding the courses and classes they are assigned to. Details of the courses include the credits, the syllabus plan. Details of the class include the department, semester, section and the list of students in each class. The teacher will also have access to information of students who belong to the same class as as the teacher.

### • 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. There will be two radio buttons next to each student name, one for present and the other for absent. There will also be an option for extra classes. Teachers can edit the attendance of each student either for each student individually or for the whole class.

### • Marks

The teacher can enter the marks for the 5 events and 1 SEE for each course they are assigned. They also have the ability to edit the marks in case of any changes. Reports such as the report card including all the marks and CGPA of a student can be generated.

## Administrator

The administrator will have access to all the information in the different tables in the database. They will access to all the tables in a list form. They will be able to add a entry in any table and also edit them. The design of the view for the admin will provide a modular interface so that querying the tables will be optimized. They will be provided with search and filter features so that they can access data efficiently.

## Class Diagram

The class diagram states the different classes involved in the software. For each class, a set of attributes and method are included. The relationship between the classes are also specified. For example, the teacher class has the attributes Id, name, phone no, address and methods such as marking attendance, declaring marks and preparing report cards. Each instance of the teacher class belongs to a department. This is specified by the relationship between Teacher and Department classes.

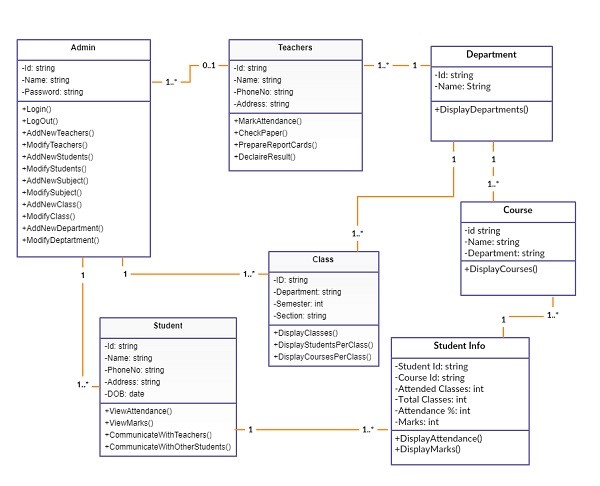


Figure 3.1: Class diagram of college ERP

## Entity Relationship Diagram

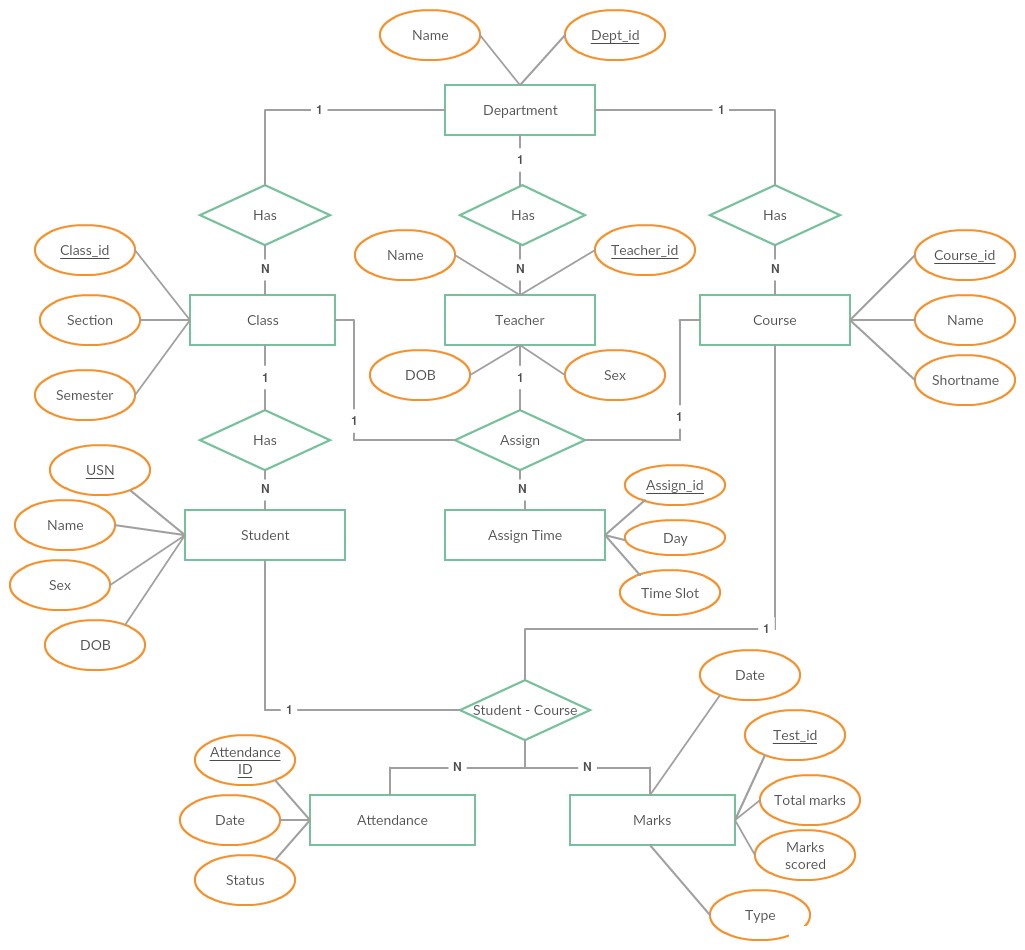


Figure 3.2: Entity Relationship diagram of college ERP

## Architectural design

The ERP software requires the architectural design to represent the design of the software. Here we define a collection of hardware and software components and their interfaces to establish the framework for the development of this software.

There exists number of components of the system which are integrated to form a system. The set of connectors will help in coordination, communication, and cooperation between the components. The ERP software is built for computer-based system. It exhibits the data centric style of architecture.

### Architectural style

In the college ERP software, the database stores the data of all the students and faculties and the stored data is updated, added, deleted or modified. So it exhibits the **data centric architectural style.**

In this architecture different components communicate with the shared data repository. The components access a shared data structure and are relatively independent.

The components are:

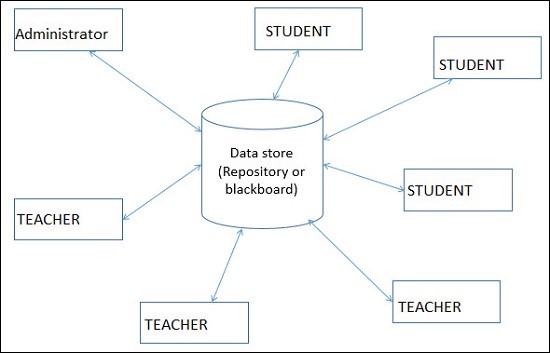


Figure 3.3: Data Centric architectural style

#### • Central data

Also known as data store or data repository, which is responsible for providing permanent data storage. It represents the current state. It stores the information of students, attendance of students and faculties of each day, salary of all the faculties etc...

#### • Data accessors

Data accessors one of the components, they are also called as clients. A data accessor operates on the central data store, perform computations, and might put back the results. Which includes students, faculties and administrator. Students requests to access the data from the repository and gets the request serviced. Faculty members modify the data in the repository. Administrator can add or delete the clients.

#### • interface

Interface is the connecting component between data repository and clients’ client interact with the data through the web server.

The operation of one client does not depend on the others. They are independent of each other. This data centered architecture will promote integrability. This means that the existing components can be changed and new client components can be added to the architecture without the permission or concern of other clients. Addition of removal of students and faculties can be done without the permission of other students and faculties.

CHAPTER 5

METHODOLOGY

The college ERP system has three main user classes. These include the students, teachers and administrator. This section will explain in detail all the features and the working of those for each user class.

## Student

### Login

Each student in the college is assigned a unique username and password by the administrator. The username is the same as their USN and so is the password. They may change it later according to their wish.

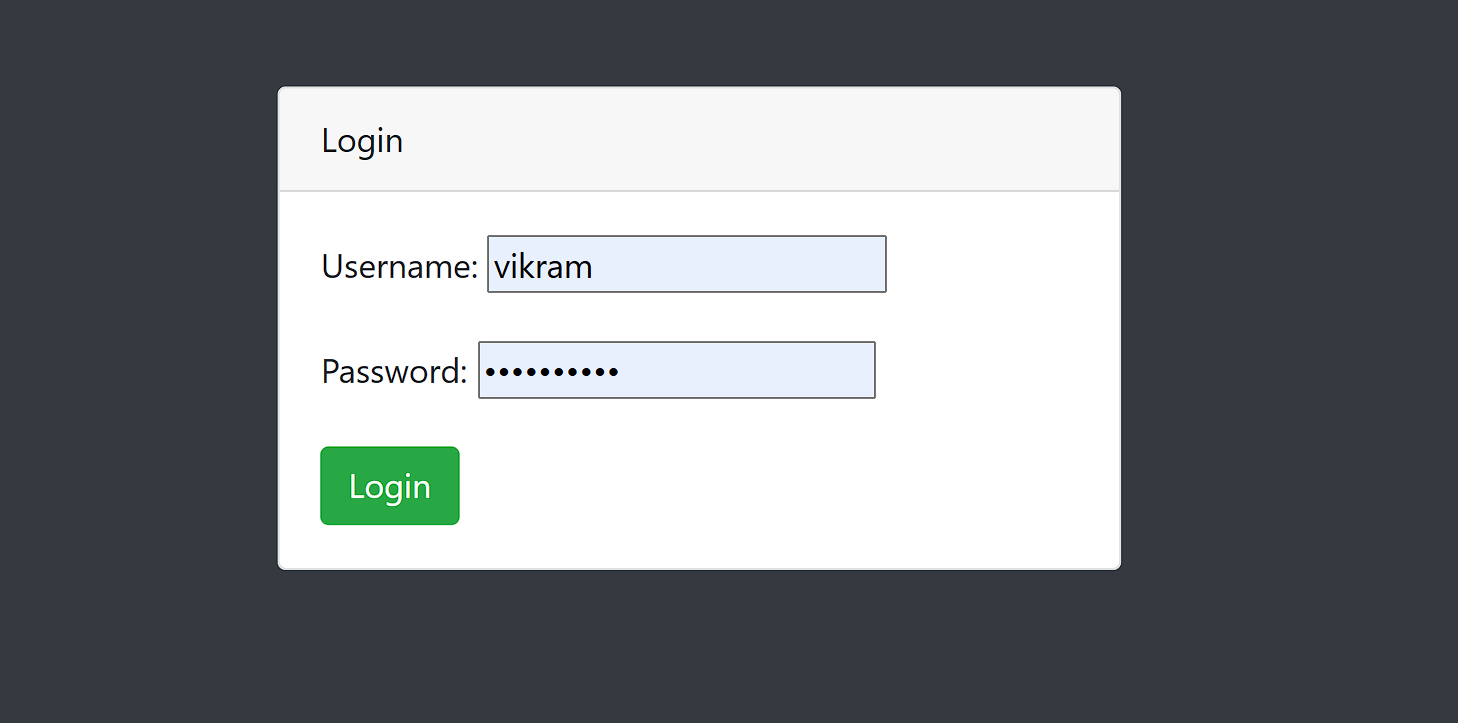


Figure 4.1: Student Login Page

### Homepage

After successful login, the student is presented a homepage with their main sections, attendance, marks and timetable. In the attendance section the student can view their attendance status which includes the total classes, attended classes and the attendance percentage for each of their courses.

In the marks section, the student can view the marks for each of their courses out of 20 for 3 internal assessments, 2 events. Also, the semester end examination for 100 marks. Lastly, the timetable provides the classes assigned to that student and day and time of each in a tabular form.

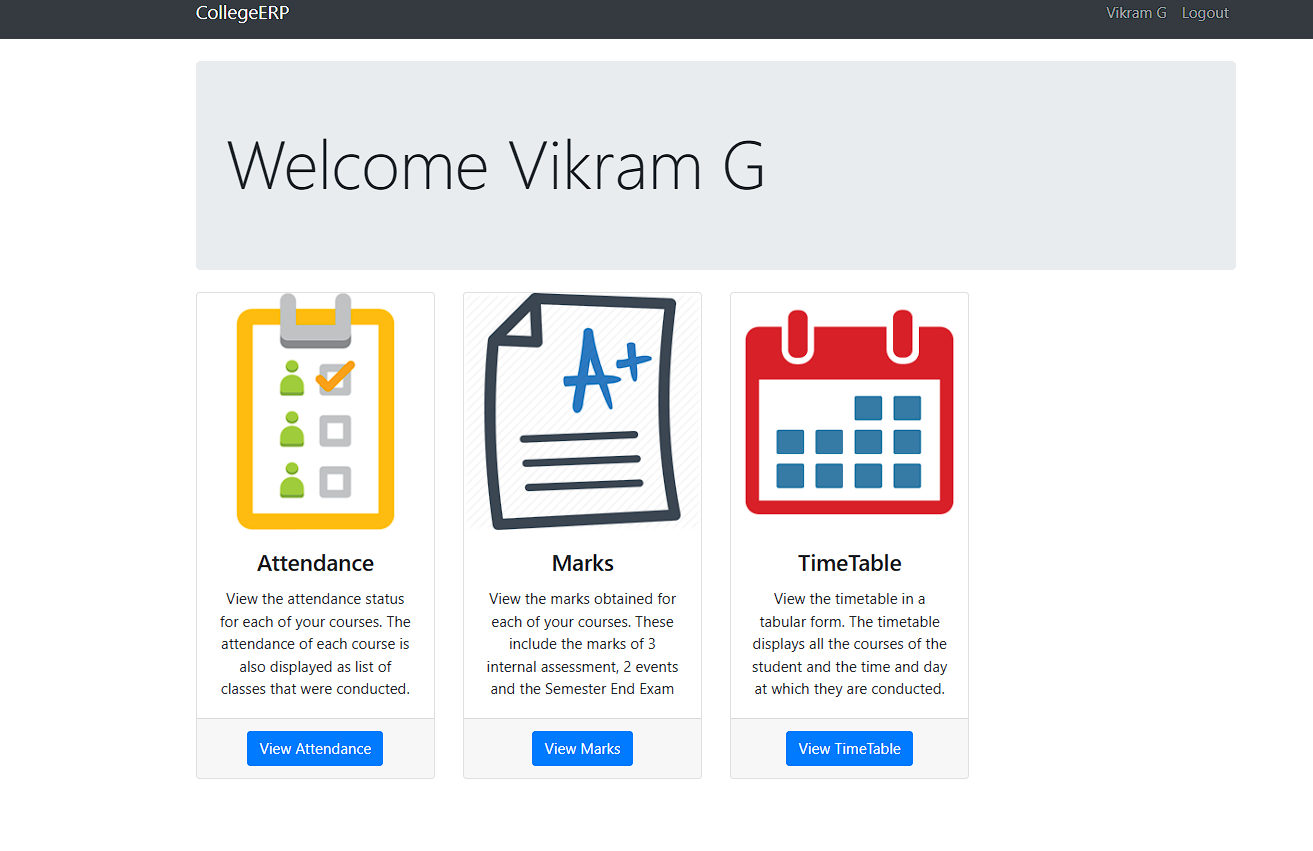


Figure 4.2: Student Home Page

### Attendance

On the attendance page, there is a list of courses that is dependent on each student. For each course, the course id and name are display along with the attended classes, total classes and the attendance percentage for that particular course. If the attendance percentage is below 75 for any course, it is displayed in red denoting shortage of attendance, otherwise it is green. If there is any shortage, it specifies the number of classes to attend to make up for it. If you click on each course, it takes you to the attendance detail page.

#### Attendance Detail

This page displays more details for the attendance in each course. For each the course, there is a list of classes conducted and each is marked with the date, day and whether the student was present or absent on that particular date.

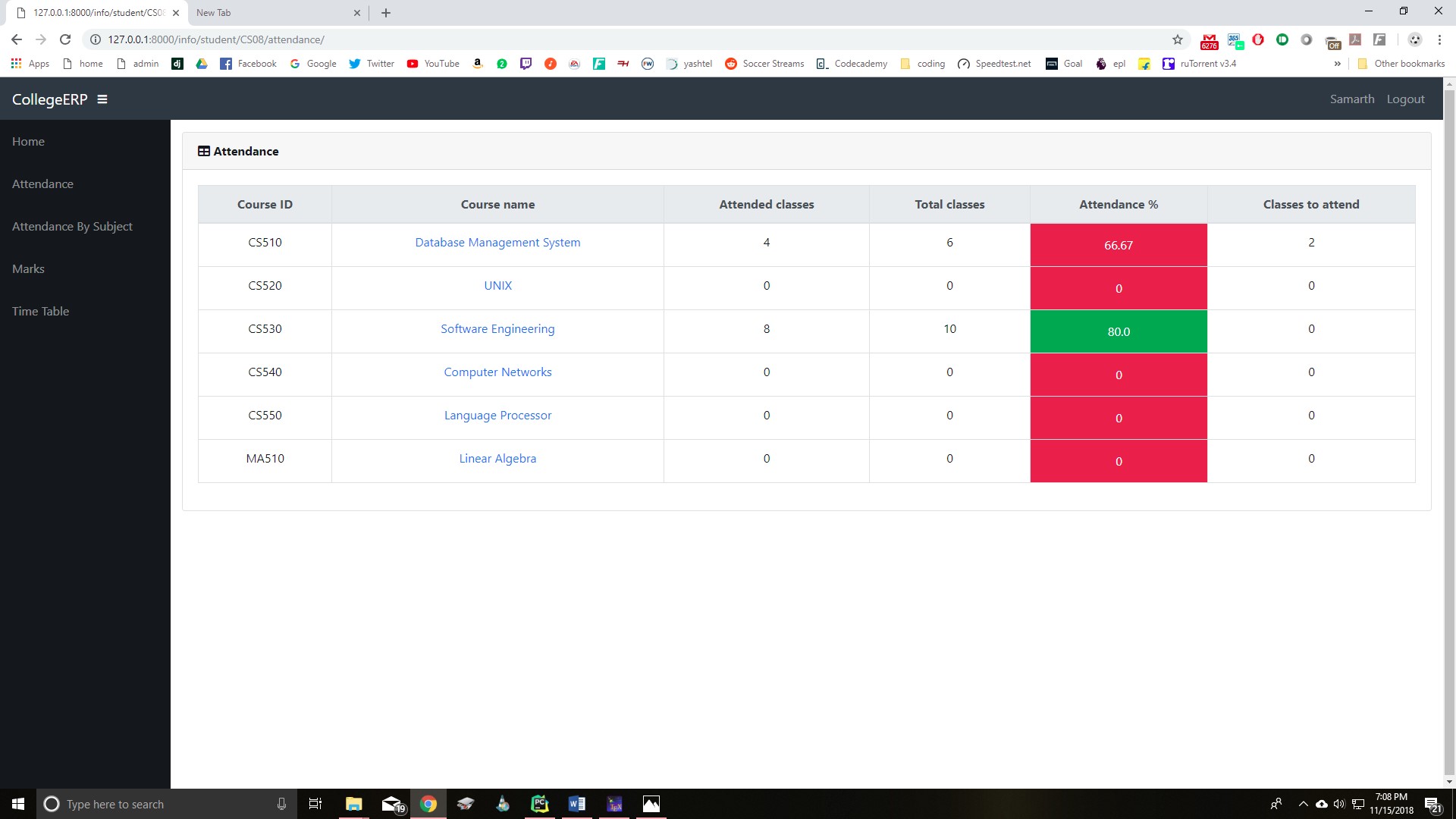


Figure 4.3: Student Attendance Page

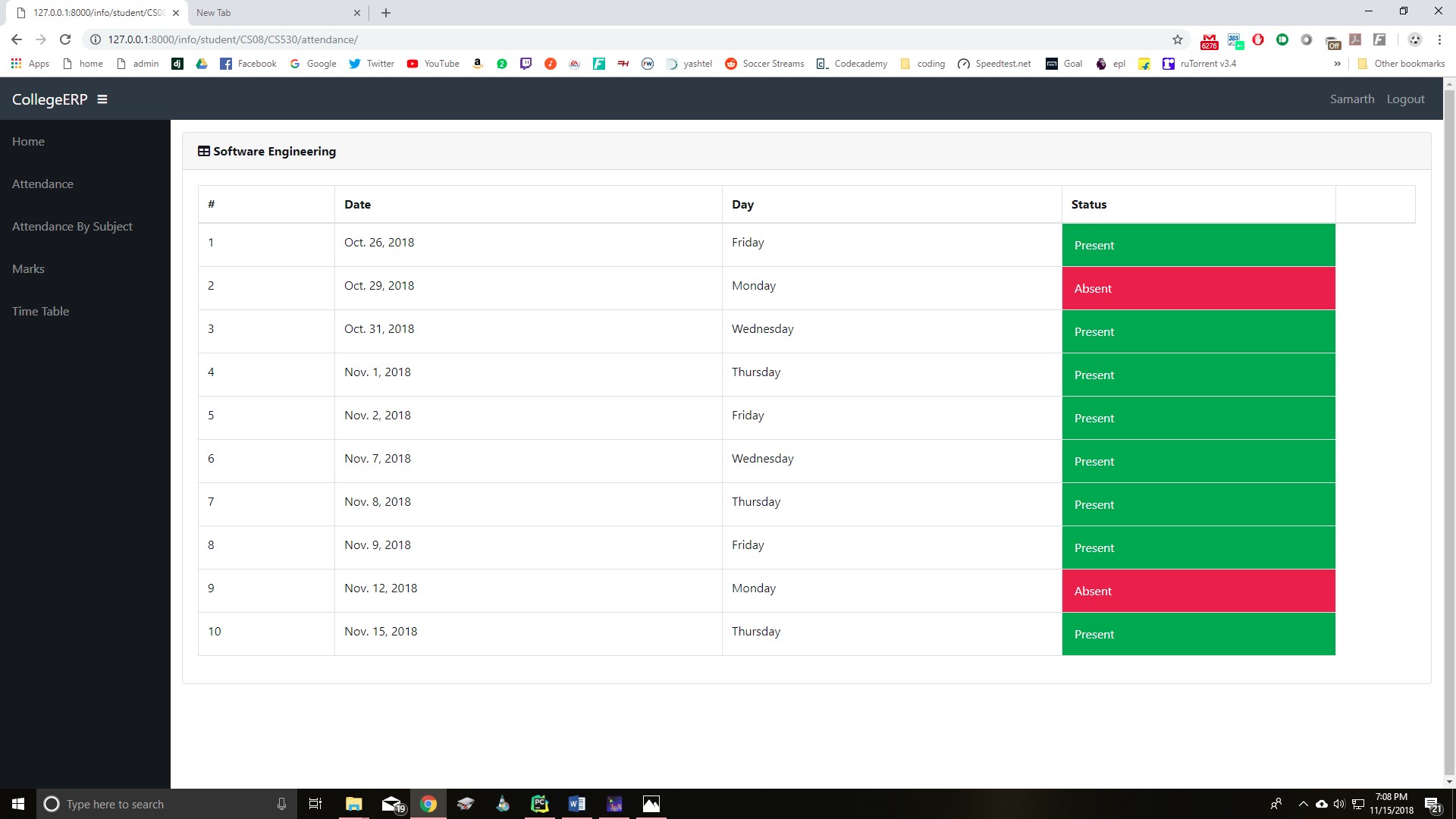


Figure 4.4: Student Attendance Detail Page

### Marks

The Marks page is a table with an entry for each of their courses. The course id and name are specified along the marks obtained in each of the tests and exams. The tests include 3 internal assessments with marks obtained out of a total of 20, 2 events such as project, assignment, quiz etc., with marks out of

20. Lastly, one semester end exam with marks out of 100.

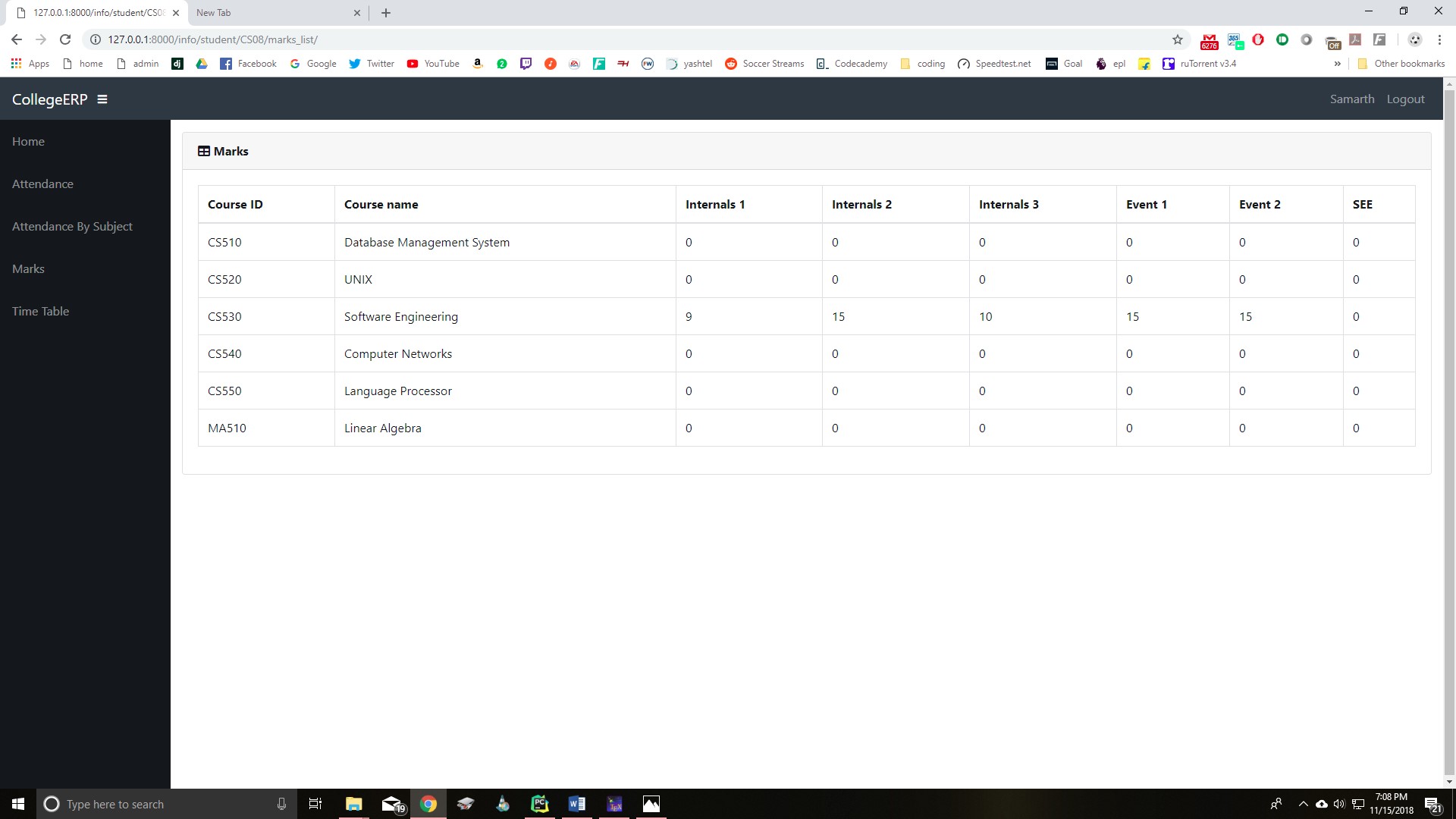


Figure 4.5: Student Marks Page

### Timetable

This page is a table which lists the day and timings of each of the classes assigned to the student. The row headers are the days of the week and the column headers are the time slots. So, for each day, it specifies the classes in the time slots. The timetable is generated automatically from the assign table, which is a table containing the information of all the teachers assigned to a class with a course and the timings the classes.



Figure 4.6: Student Timetable

## Teacher

### Login

Each teacher in the college is assigned a unique username and password by the administrator. The username is their teacher ID and the same for password. The teacher may change the password later.

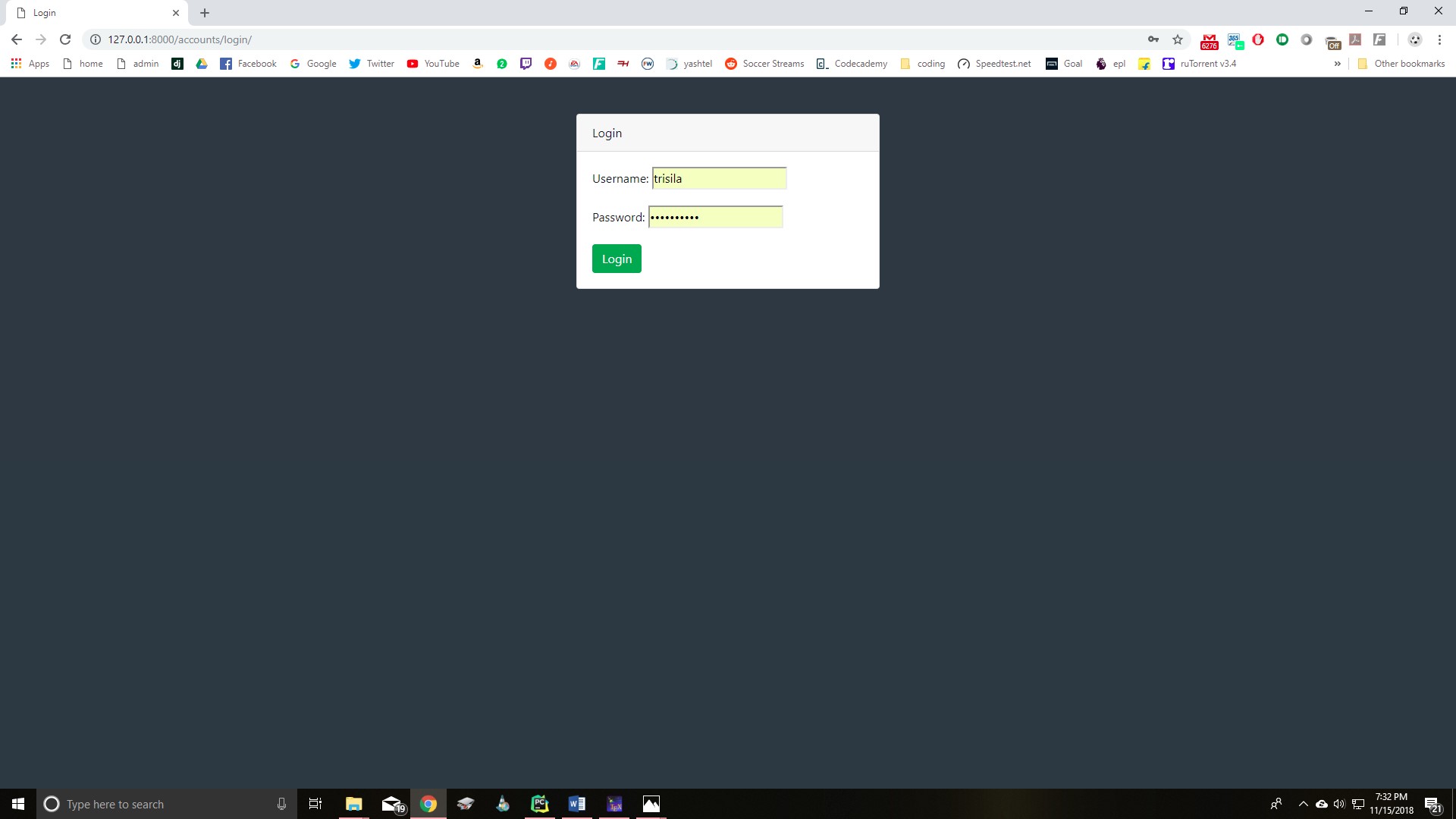


Figure 4.7: Teacher Login

### Homepage

After successful login, the student is presented a homepage with their main sections, attendance, marks, timetable and reports. In the attendance section, the teacher can enter the attendance of their respective students for the days on which classes were conducted. There is a provision to enter extra classes and view/edit the attendance of each individual student. In the marks section, the teacher may enter the marks for 3 internals, 2 events and 1 SEE for each student. They can also edit each of the entered marks. The timetable provides the classes assigned to the teacher with the day and timings in a tabular form. Lastly, the teacher can generate reports for each of their assigned class.

### Attendance

There is a list of all the class assigned to teacher. So, for each class there are 3 actions available. They

are,

#### Enter Attendance

On this page, the classes scheduled or conducted is listed in the form of a list. Initially, all the scheduled classes will be listed from the start of the semester to the current date. Thus, if there is class scheduled for today, it will automatically appear on top of the list. If the attendance of any day is not marked it will be red, otherwise green if marked. Classes can also be cancelled which will make that date as yellow. While entering the attendance, the list of students in that class is listed and there are two options next

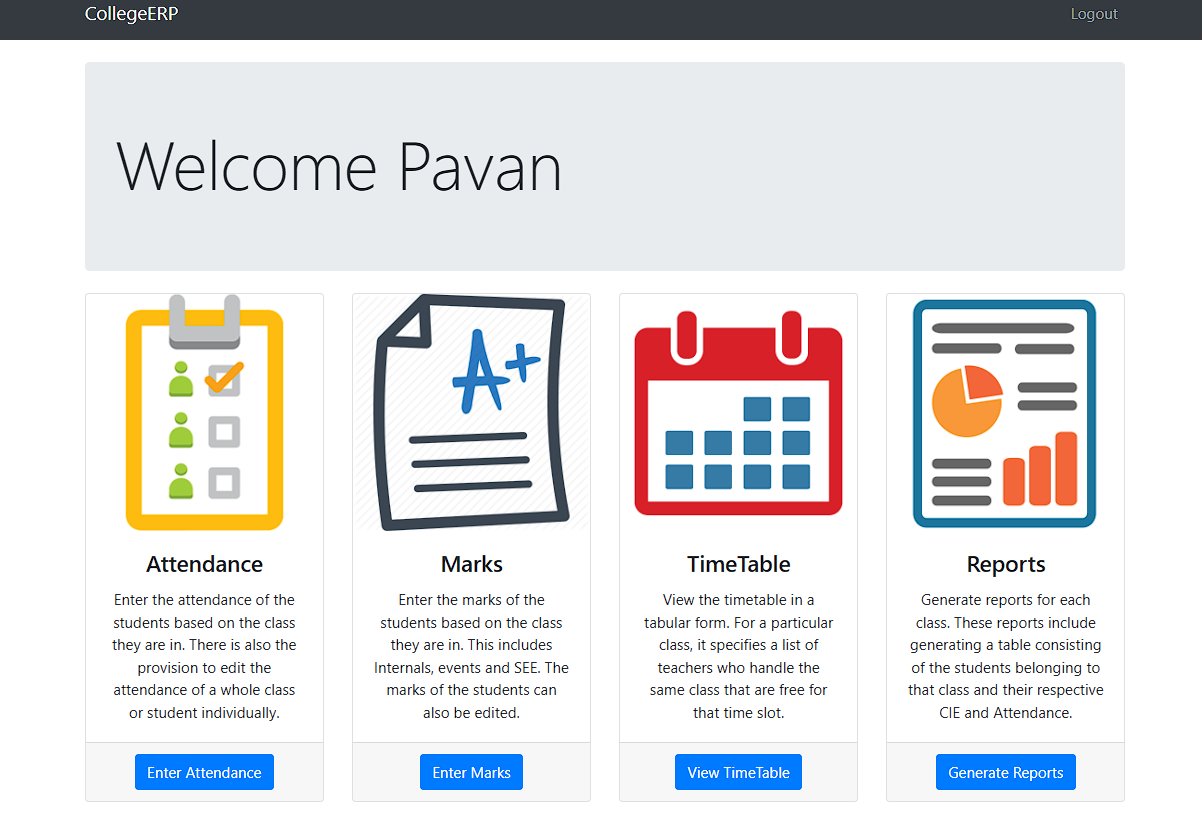


Figure 4.8: Teacher homepage

to each. These options are in the form of a radio button for present and absent. All the buttons are initially marked as present and the teacher just needs to change for the absent students.

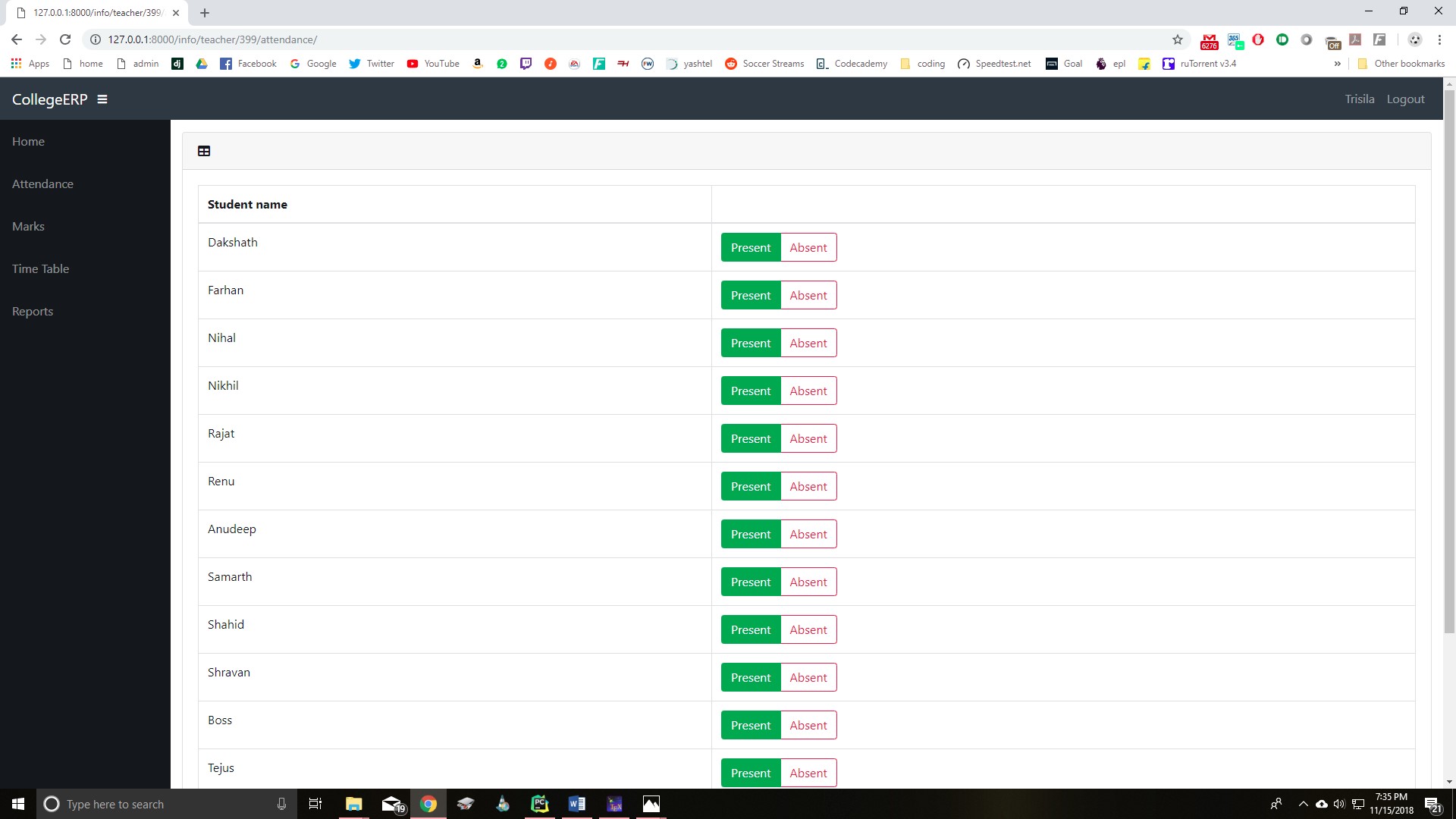


Figure 4.9: Entering attendance

#### Edit Attendance

After entering attendance, the teacher can also edit it. It is similar to screen for entering attendance, only the entered attendance is saved and display. The teacher can change the appropriate attendance and save it.

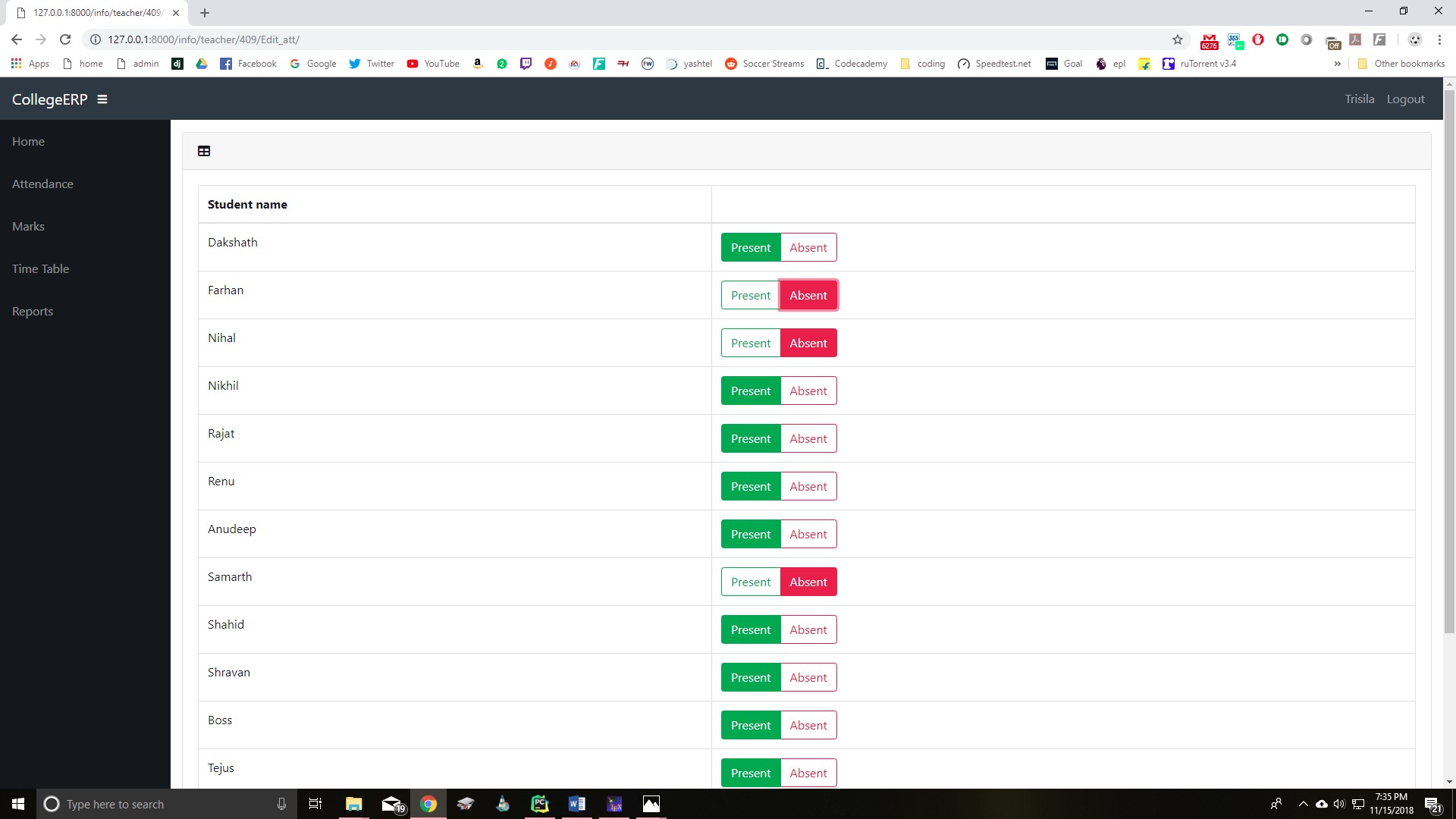


Figure 4.10: Editing attendance

with CIE below 25 are marked in red and are not eligible to write the semester end exam. Also, the attendance percentage is displayed with students below 75% marked in red.

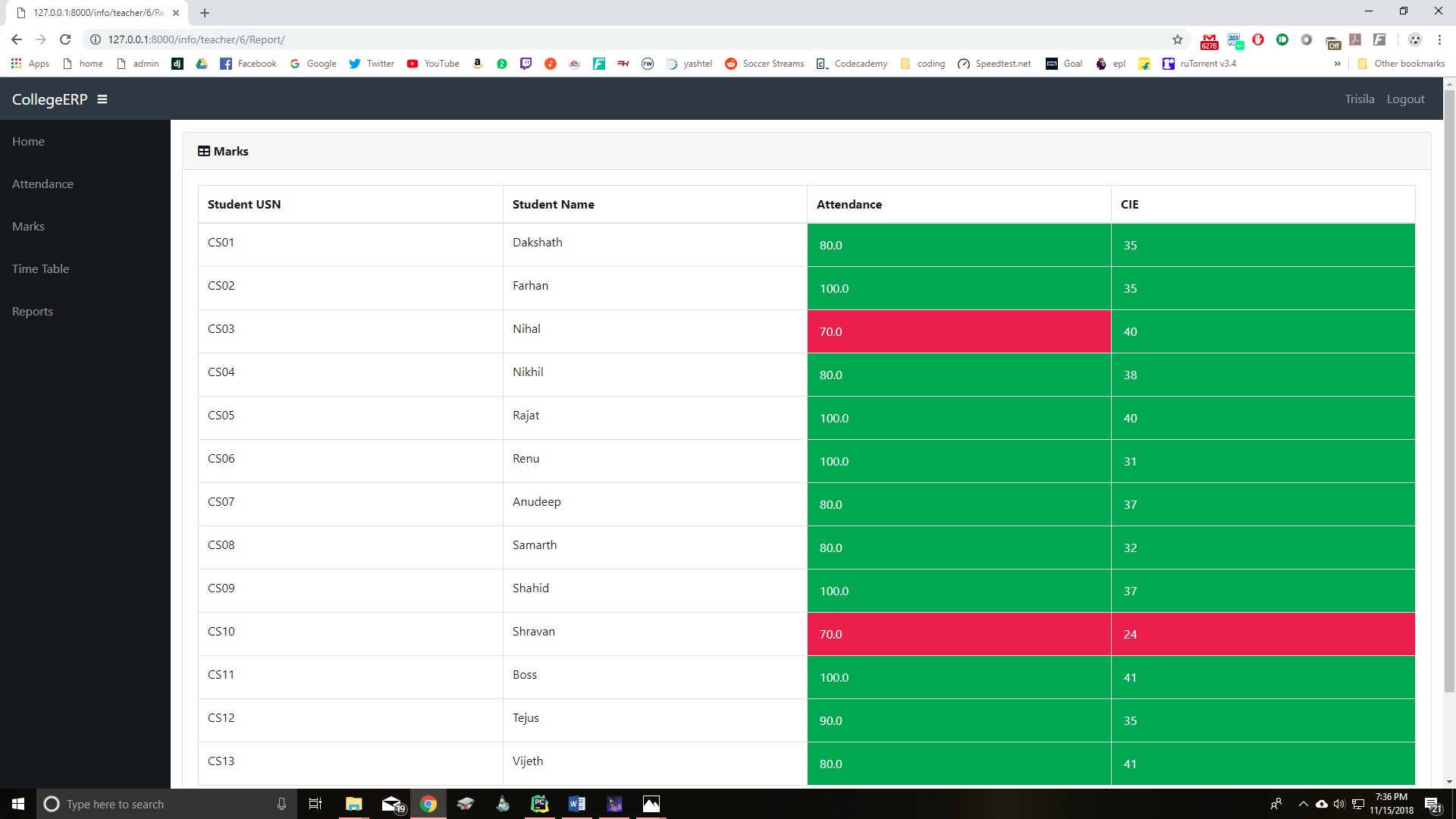


Figure 4.18: CIE and attendance for a class of students

## Administrator

The administrator is responsible for adding and maintaining all the departments, students, teachers, classes and courses. All this data is stored in the database in their respective tables. The admin is also responsible for adding and maintaining the list of teachers assigned to class with a course and the timings. This information is stored in the Assign table. The admin also has access to the marks and attendance of each student and can modify them.

There are several features in place to ensure that querying the database is quick and efficient for the administrator. As the database has the potential to become huge, there is a search feature for every table including student, teacher etc. The search has get a specific record based on name or id. Also, it can filter the record based on department, class etc.

Figure 3.19 shows the homepage for the admin, it lists all the different tables in the database. Figure 3.20 shows the details of the class table. Each class consists of a list of students as shown.

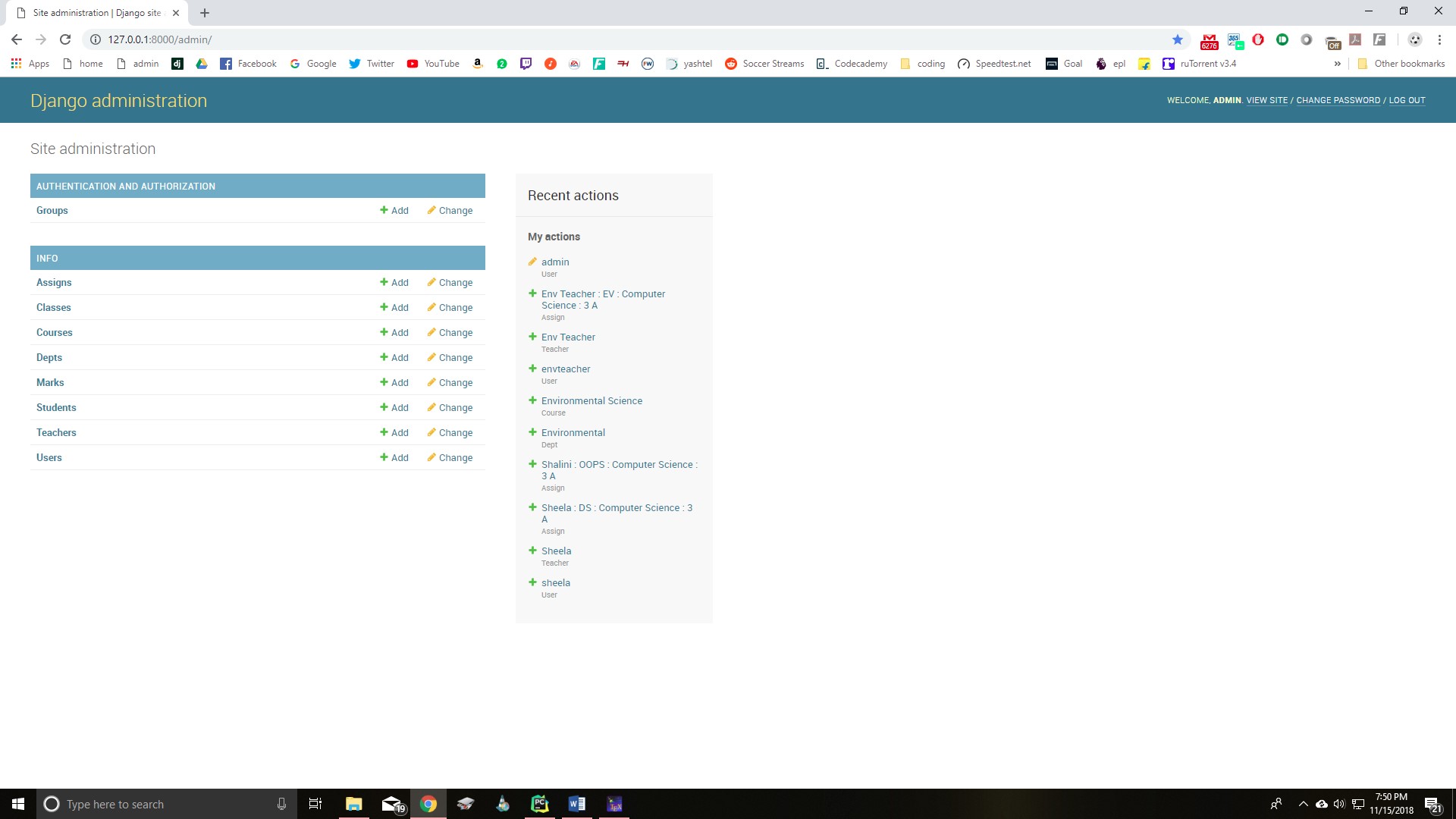


Figure 4.19: Admin homepage

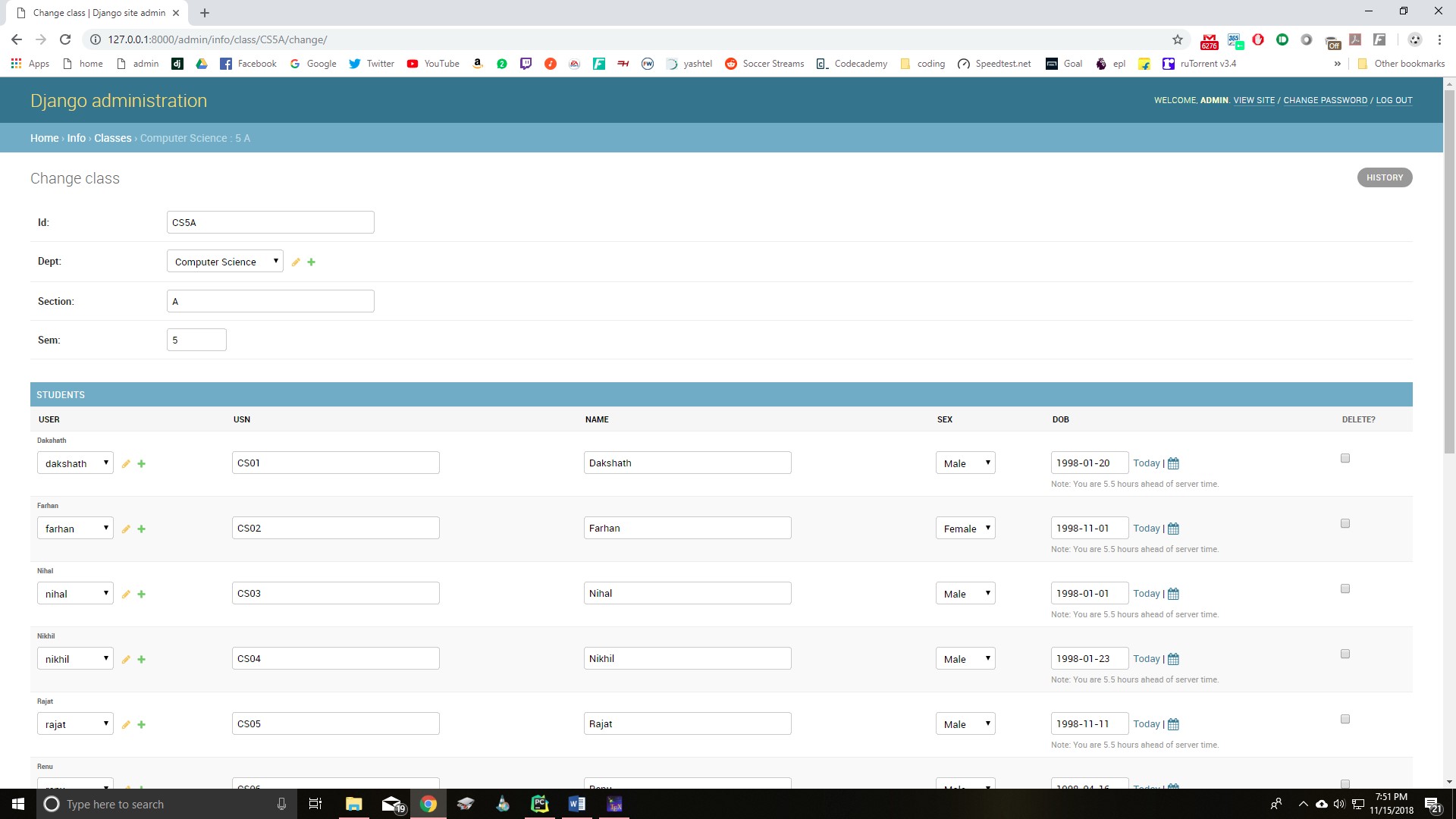


Figure 4.20: Admin students table page

CHAPTER 6

APPLICATIONS

#### **Student Management**

* **Features**:
  + Manage student profiles, admission details, and academic records.
  + Track attendance and academic performance.
* **Applications**:
  + Streamlines student enrollment and data storage.
  + Allows students to access their academic details online.

#### **2. Faculty Management**

* **Features**:
  + Manage faculty profiles and teaching schedules.
  + Track faculty workload and performance.
* **Applications**:
  + Simplifies timetable creation and course allocation.
  + Helps administrators assess faculty contributions and workloads.

#### **3. Course and Curriculum Management**

* **Features**:
  + Manage course details, syllabi, and schedules.
  + Assign students and faculty to specific courses.
* **Applications**:
  + Ensures efficient course scheduling.
  + Provides students with up-to-date course materials and schedules.

#### **4. Attendance and Examination Management**

* **Features**:
  + Record student attendance and generate reports.
  + Manage examination schedules, question papers, and grading.
* **Applications**:
  + Reduces manual effort in tracking attendance.
  + Automates exam management and result generation.

#### **5. Fee and Financial Management**

* **Features**:
  + Manage fee collection and payment tracking.
  + Generate financial reports for institutional analysis.
* **Applications**:
  + Simplifies fee payment for students.
  + Helps institutions track financial inflow and reduce errors in bookkeeping.

#### **6. Administrative Operations**

* **Features**:
  + Handle announcements, event management, and resource allocation.
  + Manage library, laboratory, and other institutional resources.
* **Applications**:
  + Improves communication between administration, faculty, and students.
  + Enhances resource utilization and tracking.

#### **7. Reporting and Analytics**

* **Features**:
  + Generate detailed reports on student performance, faculty efficiency, and operational data.
  + Visualize key metrics for better decision-making.
* **Applications**:
  + Provides actionable insights for academic and administrative improvement.
  + Helps in long-term planning and development.

#### **8. Communication and Notifications**

* **Features**:
  + Send notifications about events, deadlines, and announcements.
  + Facilitate communication between students, faculty, and administration.
* **Applications**:
  + Ensures timely delivery of critical information.
  + Reduces communication gaps within the institution.

#### **9. Alumni Management**

* **Features**:
  + Maintain a database of alumni and their career progress.
  + Facilitate alumni interactions and contributions to the institution.
* **Applications**:
  + Strengthens alumni relationships.
  + Encourages alumni involvement in college events and mentorship.

CHAPTER 7

CONCLUSION

The College ERP System developed using Django provides a comprehensive and integrated platform to manage the diverse operations of an educational institution. By automating administrative, academic, and operational tasks, the system enhances efficiency, reduces errors, and saves time for all stakeholders.

With its user-friendly interface and role-based access control, the system ensures seamless interaction for administrators, faculty, and students. It fosters better communication, real-time data access, and transparency within the institution. The modular and scalable design allows for future enhancements, making it adaptable to the growing needs of modern educational institutions.

In conclusion, the College ERP System not only simplifies day-to-day management but also empowers institutions to focus on their primary goal: delivering quality education. Its successful implementation can significantly contribute to the streamlined functioning and growth of the institution.

CHAPTER 8

REFERENCES

* **Django Documentation**:
  + Comprehensive official documentation for the Django framework, which guided the backend development.
  + URL: [https://docs.djangoproject.com](https://docs.djangoproject.com/)
* **Bootstrap Framework**:
  + Official website for Bootstrap, used to design responsive user interfaces.
  + URL: [https://getbootstrap.com](https://getbootstrap.com/)
* **SQLite Documentation**:
  + Reference for managing the lightweight database during development.
  + URL: <https://sqlite.org/docs.html>
* **Django REST Framework**:
  + For creating APIs to ensure scalability.
  + URL: [https://www.django-rest-framework.org](https://www.django-rest-framework.org/)