

**A Project/Dissertation Report**

on

**University Management System ERP**

*Submitted in partial fulfillment of the  
requirement for the award of the degree of*

**B. Tech (CSE)**



(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

**Under The Supervision of**

**Mr. Padmanbhan P.**

**Assistant Professor**

Submitted By

Aparna Shukla & Omkar Aditya

18SCSE1140078 & 18SCSE1140080

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING  
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**GALGOTIAS UNIVERSITY, GREATER NOIDA**

**INDIA**

**DECEMBER, 2021**

## Acknowledgment

First, we would like to thank Galgotias University for implementing SCSE Project as a subject for our course. We would also like to thank our Professor **Mr. Padmanbhan P (Assistant Professor)** for giving us an opportunity to do this project about College ERP. We would also like to thank our system and faculties for giving us more information for the requirements for our project. We would also like to thank our friends who were our backbone and helped us in collecting the requirements for our project. At last, we would like to thank all the other people who were involved directly or indirectly in collecting the requirements.

Aparna Shukla(18SCSE1140078)

Omkar Aditya (18SCSE1140080)



**SCHOOL OF COMPUTING SCIENCE AND  
ENGINEERING**  
**GALGOTIAS UNIVERSITY, GREATER NOIDA**

**CANDIDATE'S DECLARATION**

We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled “**University Management System ERP**” in partial fulfillment of the requirements for the award of the B.Tech(CSE) submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of July, 2021 to December 2021, under the supervision of **Mr. Padmanbhan P**, Assistant Professor, Department of Computer Science and Engineering, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by us for the award of any other degree of this or any other places.

Aparna Shukla 18SCSE1140078

Omkar Aditya 18SCSE1140080

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor Name

Designation

**CERTIFICATE**

The Final Thesis/Project/Dissertation Viva-Voce examination of Aparna Shukla:  
18SCSE1140078 and Omkar Aditya: 18SCSE1140080 has been held on \_\_\_\_\_  
and his/her work is recommended for the award of B.Tech (CSE)

**Signature of Examiner(s)**

**Signature of Supervisor(s)**

**Signature of Project Coordinator**

**Signature of Dean**

Date:

Place: Greater Noida

## Abstract

Enterprise Resource Planning system, popularly known as ERP system has become very popular as an enterprise management software tool. By using Existing System for accessing information from files is a difficult task and there is no quick and easy way to keep the records of students and staff. Lack of automation is also there in the Existing System.

The aim of Our System is to reduce the workload and to save significant staff time. Title of the project as University Management ERP is the system that deals with the issues related to a particular institution. It is the very useful to the student as well as the faculties to easy access to finding the details. The University Management ERP provides appropriate information to users based on their profiles and role in the system. This project is designed keeping in view the day-to-day problems faced by a college system.

The fundamental problem in maintaining and managing the work by the administrator is hence overcome. This System provide the automate admissions no manual processing is required. This is a paperless work. It can be monitored and controlled remotely. It reduces the man power required. It provides accurate information always. All years together gathered information can be saved and can be accessed at any time. The data which is stored in the repository helps in taking intelligent decisions by the management providing the accurate results. The storage facility will ease the job of the operator.

Thus, the system developed will be helpful to the administrator by easing his/her task providing the accurate results. The storage facility will ease the job of the operator. This project is successfully implemented with all the features and modules of the college management system as per requirements.

**Keywords:** - Web Development, Enterprise Resource Planning, Web Application, Python, University Management, Django.

## Contents

Title	Page No.
<b>Acknowledgement</b>	<b>I</b>
<b>Candidate Declaration</b>	<b>II</b>
<b>Certificate</b>	<b>III</b>
<b>Abstract</b>	<b>IV</b>
<b>List of Figures</b>	<b>VII</b>
<b>Chapter 1</b>	<b>1.1 Introduction</b>
	1.1.1 Introduction to Problem Domain
	1.1.2 Aim of the Problem
	1.1.3 Time Schedule
	<b>1.2 Formulation of Problem</b>
	1.2.1 What is the purpose of this Project?
	1.2.2 Why do we need ERP?
	1.2.3 Identification of Stakeholders
	1.2.4 Problems With Current ERP System
	1.2.5 Tool and Technology Used
<b>Chapter 2</b>	<b>Project Design</b>
	2.1 Literature Survey
	2.2 Architecture
	2.3 Class Diagram
	2.4 ER Diagram
	2.5 Architectural Design
<b>Chapter 3</b>	<b>Technologies Used</b>
	<b>3.1 Django</b>
	<b>3.2 Sqlite Database</b>
	<b>3.3 Web Application</b>
	3.3.1 How does a Web Application Works
	3.3.2 Benefits of a Web Application
<b>Chapter 4</b>	<b>Working of Project</b>
	<b>4.1 Student</b>
	4.1.1 Login
	4.1.2 Homepage
	4.1.3 Attendance
	4.1.4 Marks
	4.1.5 Timetable

	<b>4.2 Teacher</b>	<b>32</b>
	4.1.1 Login	32
	4.1.2 Attendance	33
	4.1.3 Marks	35
	4.1.4 Timetable	36
	<b>4.3 Administrator</b>	<b>38</b>
	4.3.1 Login	38
	4.3.2 Homepage	39
	4.3.3 Users	40
	<b>4.4 Instructions</b>	<b>44</b>
<b>Chapter 5</b>	<b>Results and Discussion</b>	<b>43</b>
	<b>5.1 Testing Methods</b>	<b>43</b>
	5.1.1 White Box Testing	43
	5.1.2 Black Box Testing	44
	<b>5.2 Results of testing</b>	<b>45</b>
<b>Chapter 6</b>	<b>6.1 Conclusion</b>	<b>47</b>
	<b>6.2 Future Work</b>	<b>48</b>
	<b>References</b>	<b>50</b>

## List of Figure

<b><u>Figure No</u></b>	<b><u>Figure Name</u></b>	<b><u>Page No</u></b>
<b>1</b>	Class Diagram	<b>13</b>
<b>2</b>	ER Diagram	<b>14</b>
<b>3</b>	Architectural Design	<b>16</b>
<b>4</b>	Student Login	<b>27</b>
<b>5</b>	Student Homepage	<b>28</b>
<b>6</b>	Student Attendance	<b>29</b>
<b>7</b>	Student Marks	<b>30</b>
<b>8</b>	Student Timetable	<b>31</b>
<b>9</b>	Teacher Login	<b>32</b>
<b>10</b>	Teacher Attendance	<b>33</b>
<b>11</b>	Attendance Edit	<b>34</b>
<b>12</b>	Teacher Marks	<b>35</b>
<b>13</b>	Teacher Timetable	<b>36</b>
<b>14</b>	Free Teachers	<b>37</b>
<b>15</b>	Administrator Login	<b>38</b>
<b>16</b>	Administrator Homepage	<b>39</b>
<b>17</b>	Administrator Add User	<b>40</b>
<b>18</b>	Administrator Edit User	<b>41</b>
<b>19</b>	Starting Server in Command Terminal	<b>42</b>



## **CHAPTER-1**

### **Introduction**

#### **1.1 Introduction:**

The objective of University Management ERP 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 University Management ERP 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 ensures 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.

##### **1.1.1 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.

### **1.1.2 Aim of the problem**

The objective of University Management ERP 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 University Management ERP 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 ensures 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.

### **1.1.3 Time schedule for completion of the project work**

The Project schedule activities will consist of following:

- Forming The Team and selecting project title
- System Requirement Collection
- System Design
- Coding
- Testing of the Application and Deployment

## **1.2 Problem Formulation:**

### **1.2.1 What is the purpose of this project?**

There is currently an ERP system in our college. But not everyone is happy with the system. While it is a step towards automating the college activities, it comes with its own set of problems. This project is designed to implement a college ERP system to eradicate some of these problems and add some features of our own that would add value to system.

### **1.2.2 Why do we need ERP?**

Nowadays, in schools and colleges, it is very difficult to manage each and everything manually. Super- vising and maintaining the whole database of a school or college can be time-consuming and challenging especially if it's done on a regular basis. So, we need to handle and manage everything smartly. ERP software makes it easy to track the progress of every department of school and automate different functions. With ERP every-thing can be seen on a single dashboard. The administrator can manage the college from anywhere. The possibilities of maintaining the whole database of a college with ERP software are endless.

Some of the prominent roles of ERP are:

- Manages the office and automates different functions.
- Helps in long-term management and planning of all departments of college.
- Eliminates the need for having multiple management software for each department.
- Daily activities like attendance can be digitalized and automated.
- Leave module for teachers can be automated.

### **1.2.3 Identification of stakeholders**

Enterprise Resource planning implementation is a difficult and complex decision where it involves people issues more than technological issues. Identification of stakeholders is a key step during the process of ERP implementation

## **Teachers**

Teachers are the key stakeholders of the college ERP. Because they are the one who manage, edit, update the contents of the database of students such as attendance, internal marks, CGPA etc...

It also helps them to assign their class to other teachers when they are on leave. This makes it easier to identify who among them are free to take the class at that time. So, this software helps them reduce their overhead and make their tasks easier and simple.

## **Students**

Students are end users of ERP software. The attendance, internal marks uploaded by the teachers are viewed by students. It helps them track their attendance status. It also helps them to communicate with teachers and their classmates. So, students make up another set of stakeholders of this software.

## **Administrator**

College administrator is responsible for maintaining the database of the college. They will have the privilege to modify the database i.e., to add/remove students/teachers/staff, update information regarding each of these. It is their responsibility to maintain the database of students who pass out from the college and who freshly get admission to the college. So, the Administrator play a major role in the ERP.

### **1.2.4 Problems with the current ERP system**

The problem with the ERP software is if the faculty applies for leave and wants to allocate the class to any other faculty, then the request goes to all the faculties of all the departments. This should not happen because other department faculty cannot handle the class for any other department i.e., if the faculty of Computer Science department applies for the leave and if the request is sent, it must be sent to the faculties of the Computer Science department only and not for any other department like Civil, Mechanical, E&E, and so on.

When the faculty is inserting the attendance into the system, there must be a separate space for the faculty to fill what topics they have covered in the class. It will be time consuming for the faculty to enter the topic every time. So, for this purpose the software must be designed in such a way that it inserts the topic automatically. Firstly, all the topics and the duration for the faculty in

which the faculty must cover must be mentioned. And then the faculty must investigate it and cover the syllabus according to the plan. This can also keep a track of the lecturer what they are teaching.

If the doubts are raised by the students, then that would lead to shortage of time to cover the syllabus. classes when the students are free. For taking the extra class, the faculty must block in the time table and it must be visible to all the faculties of that class so that there would be no collision in handling the extra class.

### **1.2.5 Tools and Technology Used:**

#### **User Interfaces**

The User interface is made using Bootstrap. Firstly, there will be a simple login page separate for students and teachers. Each student and teacher will have a unique interface. The teachers will be able to view their respective students and update their attendance and marks using an effortless interface.

#### **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 webapp.

#### **Software Interfaces**

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

- Operating System: Windows operating system for its best support and user-friendliness.
- Django: back-end of the website as Django is a simple python framework and is suitable for beginners.
- Database: SQLite database, which comes as default with Django

#### **Communications Interfaces**

This project is to be deployed an online website. All the users can connect to the database server from anywhere and have access to their information.

## **Safety requirements**

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed-up log, up to the time of failure.

## **Security requirements**

The database contains sensitive information of all the students and staff. Therefore, optimal security measures must be taken to ensure data is safe from unauthorized users.

## **Software Quality Attributes**

**Availability:** The users must always be able to view their information so that they can keep track regularly.

**Correctness:** The information about attendance and marks must be correct to not feed wrong information to the users.

**Portability:** The users access the ERP from various platforms such as desktops and mobile phones. The webapp must be portable to all platforms and the user experience must be optimal.

## **CHAPTER-2**

### **Literature Survey**

#### **2.1 Literature Survey**

After analyzing the requirements of the task to be performed, the next step is to analyze the problem and understand its context. The first activity in the phase is studying the existing system and other is to understand the requirements and domain of the new system. Both the activities are equally important, but the first activity serves as a basis of giving the functional specifications and then successful design of the proposed system. Understanding the properties and requirements of a new system is more difficult and requires creative thinking and understanding of existing running system is also difficult, improper understanding of present system can lead diversion from solution.

#### **Analysis Model**

The model that is basically being followed is the WATER FALL MODEL, which states that the phases are organized in a linear order. First of all, the feasibility study is done. Once that part is over the requirement analysis and project planning begins. The design starts after the requirement analysis is complete and the coding begins after the design is complete. Once the programming is completed, the testing is done. In this model the sequence of activities performed in a software development project are:

- Requirement Analysis
- Project Planning
- System design
- Detail design
- Coding
- Unit testing
- System integration & testing

Here the linear ordering of these activities is critical. End of the phase and the output of one phase is the input of another phase. The output of each phase is to be consistent with the overall requirement of the system. Some of the qualities of spiral model are also incorporated like after the people concerned with the project review completion of each of the phase the work done.

WATER FALL MODEL was being chosen because all requirements were known beforehand and the objective of our software development is the computerization/automation of an already existing manual working system.

## **Feasibility Study**

Preliminary investigation examine project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

### **Technical Feasibility**

Technical Feasibility centers on the existing computer system hardware, software, etc. and to some extent how it can support the proposed addition. This involves financial considerations to accommodate technical enhancements. Technical support is also a reason for the success of the project. The techniques needed for the system should be available and it must be reasonable to use. Technical Feasibility is mainly concerned with the study of function, performance, and constraints that may affect the ability to achieve the system. By conducting an efficient technical feasibility, we need to ensure that the project works to solve the existing problem area.

Since the project is designed with HTML with CSS and JS as Front end and Django as Back end, it is easy to install in all the systems wherever needed. It is more efficient, easy and user-friendly to understand by almost everyone. Huge amount of data can be handled efficiently using Django as back end. Hence this project has good technical feasibility.

### **Operational Feasibility**

People are inherently instant to change and computers have been known to facilitate change. An estimate should be made to how strong a reaction the user staff is likely to have towards the development of the computerized system. The staff is accustomed to computerized systems. These kinds of systems are becoming more common day by day for evaluation of the software engineers. Hence, this system is



operationally feasible. As this system is technically, economically and operationally feasible, this system is judged feasible.

### **Economic Feasibility**

The role of interface design is to reconcile the differences that prevail among the software engineer's design model, the designed system meet the end user requirement with economical way at minimal cost within the affordable price by encouraging more of proposed system. Economic feasibility is concerned with comparing the development cost with the income/benefit derived from the developed system. In this we need to derive how this project will help the management to take effective decisions.

Economic Feasibility is mainly concerned with the cost incurred in the implementation of the software. Since this project is developed using HTML, CSS, JS and Django which is more commonly available and even the cost involved in the installation process is not high. Even if we want to train the persons in this area the cost involved in training is also very less. Hence this project has good economic feasibility. The system once developed must be used efficiently. Otherwise, there is no meaning for developing the system. For this a careful study of the existing system and its drawbacks are needed. The user should be able to distinguish the existing one and proposed one, so that one must be able to appreciate the characteristics of the proposed system, the manual one is not highly reliable and also is considerably fast. The proposed system is efficient, reliable and also quickly responding.

## 2.2 Architecture

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.

### 2.2.1 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 is 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 calendar 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.

### 2.2.2 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 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.

### **2.2.3 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 an 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.

## 2.3 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 is 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.

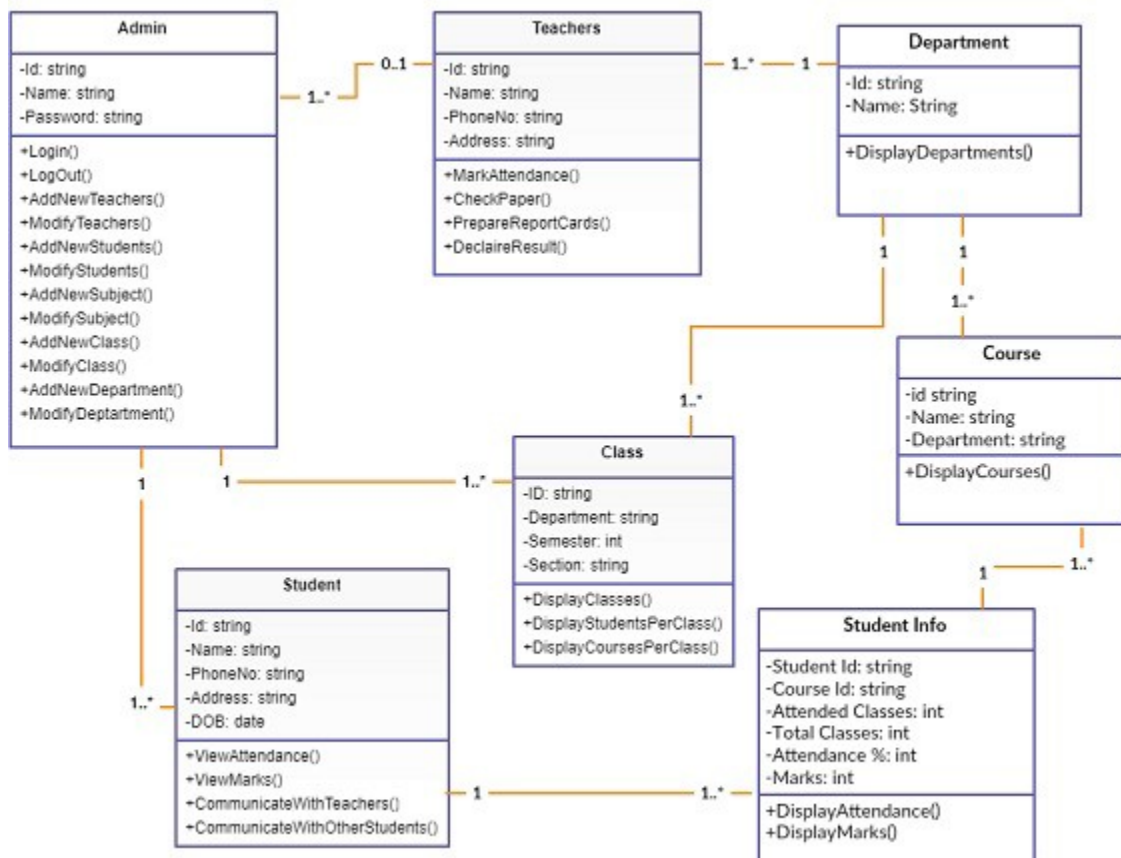


Figure 2.1: Class diagram of college ERP

## 2.4 Entity Relationship Diagram

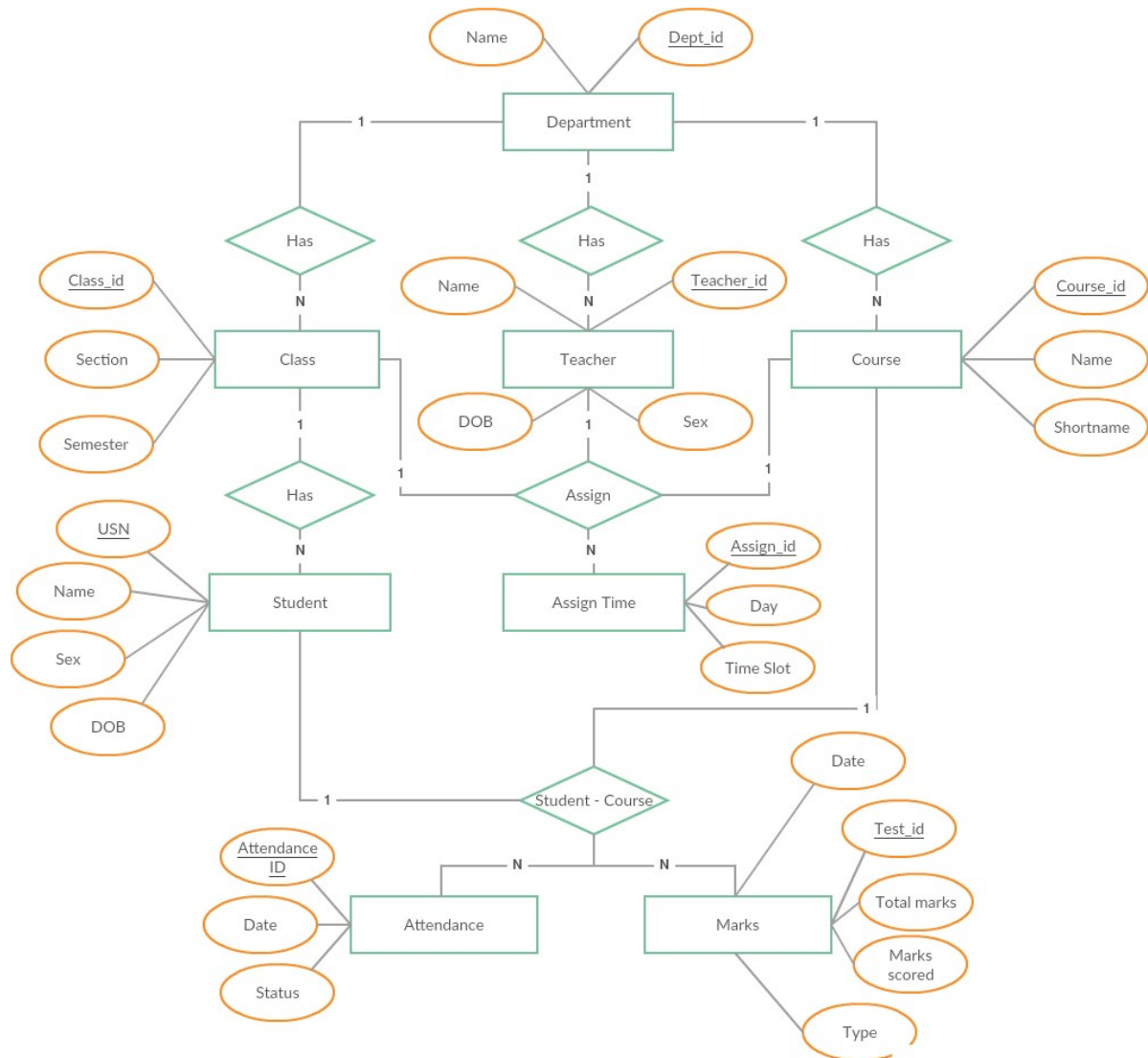


Figure 2.2: Entity Relationship diagram of college ERP

## **2.5 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.

### **2.5.1 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:

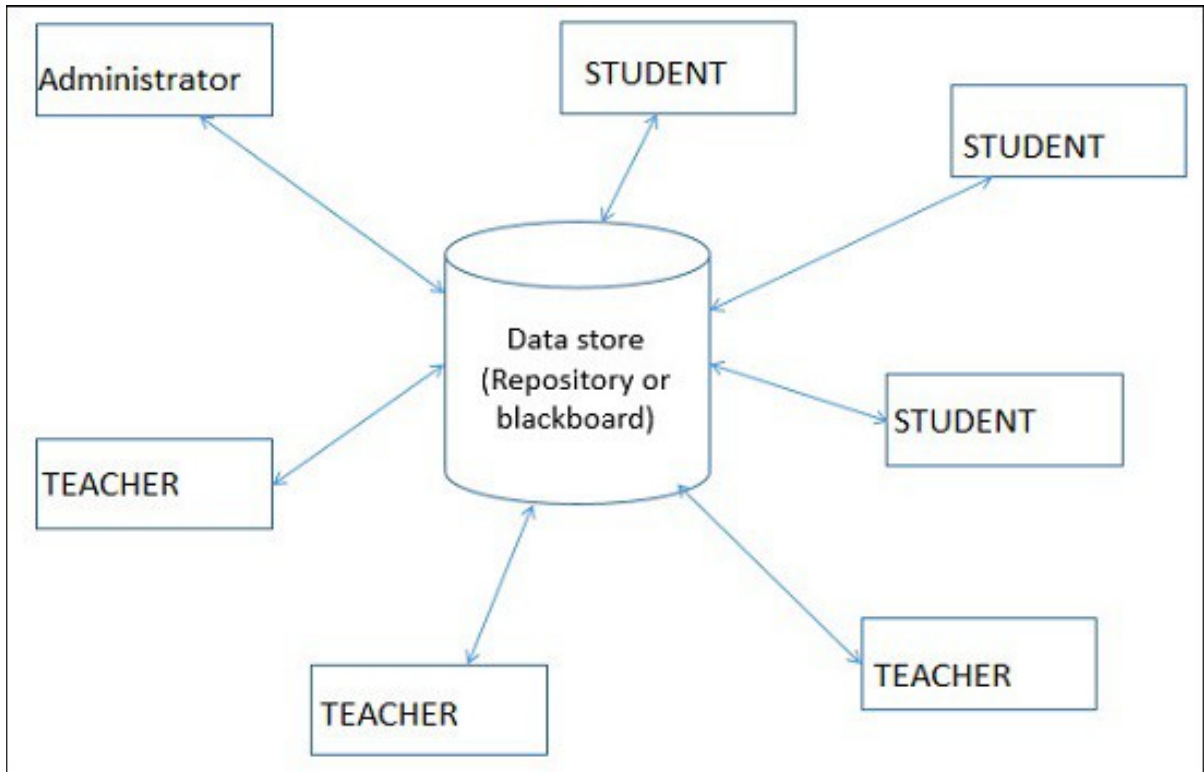


Figure 2.3: Data Centric architectural style



- **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-3**

### **TECHNOLOGY USED**

#### **3.1 DJANGO**

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support.

Django helps you write software that is:

#### **Complete**

Django follows the "Batteries included" philosophy and provides almost everything developers might want to do "out of the box". Because everything you need is part of the one "product", it all works seamlessly together, follows consistent design principles, and has extensive and [up-to-date documentation](#).

#### **Versatile**

Django can be (and has been) used to build almost any type of website — from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, XML, etc.). The site you are currently reading is built with Django!

Internally, while it provides choices for almost any functionality you might want (e.g. several popular databases, templating engines, etc.), it can also be extended to use other components if needed.

## Secure

Django helps developers avoid many common security mistakes by providing a framework that has been engineered to "do the right things" to protect the website automatically. For example, Django provides a secure way to manage user accounts and passwords, avoiding common mistakes like putting session information in cookies where it is vulnerable (instead cookies just contain a key, and the actual data is stored in the database) or directly storing passwords rather than a password hash.

## Scalable

Django uses a component-based "[shared-nothing](#)" architecture (each part of the architecture is independent of the others, and can hence be replaced or changed if needed). Having a clear separation between the different parts means that it can scale for increased traffic by adding hardware at any level: caching servers, database servers, or application servers. Some of the busiest sites have successfully scaled Django to meet their demands (e.g. Instagram and Disqus, to name just two).

## Maintainable

Django code is written using design principles and patterns that encourage the creation of maintainable and reusable code. In particular, it makes use of the Don't Repeat Yourself (DRY) principle so there is no unnecessary duplication, reducing the amount of code. Django also promotes the grouping of related functionality into reusable "applications" and, at a lower level, groups related code into modules (along the lines of the [Model View Controller \(MVC\)](#) pattern).

## Portable

Django is written in Python, which runs on many platforms. That means that you are not tied to any particular server platform, and can run your applications on many flavors of Linux, Windows, and Mac OS X. Furthermore, Django is well-supported by many web hosting providers, who often provide specific infrastructure and documentation for hosting Django sites.

In a traditional data-driven website, a web application waits for HTTP requests from the web browser (or other client). When a request is received the application works out what is needed based on the URL and possibly information in POST data or GET data. Depending on what is required it may then read or write information from a database or perform other tasks required to satisfy the request. The application will then return a response to the web browser, often dynamically creating an HTML page for the browser to display by inserting the retrieved data into placeholders in an HTML template.

Django web applications typically group the code that handles each of these steps into separate files:

- **URLs:** While it is possible to process requests from every single URL via a single function, it is much more maintainable to write a separate view function to handle each resource. A URL mapper is used to redirect HTTP requests to the appropriate view based on the request URL. The URL mapper can also match particular patterns of strings or digits that appear in a URL and pass these to a view function as data.
- **View:** A view is a request handler function, which receives HTTP requests and returns HTTP responses. Views access the data needed to satisfy requests via *models*, and delegate the formatting of the response to *templates*.
- **Models:** Models are Python objects that define the structure of an application's data, and provide mechanisms to manage (add, modify, delete) and query records in the database.
- **Templates:** A template is a text file defining the structure or layout of a file (such as an HTML page), with placeholders used to represent actual content. A *view* can dynamically create an HTML page using an HTML template, populating it with data from a *model*. A template can be used to define the structure of any type of file; it doesn't have to be HTML!

Django web applications manage and query data through Python objects referred to as models. Models define the structure of stored data, including the field *types* and possibly also their maximum size, default values, selection list

options, help text for documentation, label text for forms, etc. The definition of the model is independent of the underlying database — you can choose one of several as part of your project settings. Once you've chosen what database you want to use, you don't need to talk to it directly at all — you just write your model structure and other code, and Django handles all the "dirty work" of communicating with the database for you.

The Django model provides a simple query API for searching the associated database. This can match against a number of fields at a time using different criteria (e.g. exact, case-insensitive, greater than, etc.), and can support complex statements (for example, you can specify a search on U11 teams that have a team name that starts with "Fr" or ends with "al").

Template systems allow you to specify the structure of an output document, using placeholders for data that will be filled in when a page is generated. Templates are often used to create HTML, but can also create other types of document. Django supports both its native templating system and another popular Python library called Jinja2 out of the box (it can also be made to support other systems if needed).

The preceding sections show the main features that you'll use in almost every web application: URL mapping, views, models and templates. Just a few of the other things provided by Django include:

- **Forms:** HTML Forms are used to collect user data for processing on the server. Django simplifies form creation, validation, and processing.
- **User authentication and permissions:** Django includes a robust user authentication and permission system that has been built with security in mind.
- **Caching:** Creating content dynamically is much more computationally intensive (and slow) than serving static content. Django provides flexible caching so that you can store all or part of a rendered page so that it doesn't get re-rendered except when necessary.
- **Administration site:** The Django administration site is included by default when you create an app using the basic skeleton. It makes it trivially easy to

provide an admin page for site administrators to create, edit, and view any data models in your site.

- **Serialising data:** Django makes it easy to serialise and serve your data as XML or JSON. This can be useful when creating a web service (a website that purely serves data to be consumed by other applications or sites, and doesn't display anything itself), or when creating a website in which the client-side code handles all the rendering of data.

## **3.2 SQLITE DATABASE**

This **SQLite tutorial** teaches you everything you need to know to start using SQLite effectively. In this tutorial, you will learn SQLite step by step through extensive hands-on practices.

This SQLite tutorial is designed for developers who want to use SQLite as the back-end database or to use SQLite to manage structured data in applications including desktop, web, and mobile apps.

SQLite is an open-source, zero-configuration, self-contained, stand-alone, transaction relational database engine designed to be embedded into an application.

### **Data definition**

In this section, you'll learn how to create database objects such as tables, views, indexes using SQL data definition language.

- SQLite Data Types – introduce you to the SQLite dynamic type system and its important concepts: storage classes, manifest typing, and type affinity.
- Create Table – show you how to create a new table in the database.
- Alter Table – show you how to use modify the structure of an existing table.
- Rename column – learn step by step how to rename a column of a table.
- Drop Table – guide you on how to remove a table from the database.
- VACUUM – show you how to optimize database files.

## Changing data

This section guides you on how to update data in the table using insert, update, delete, and replace statements.

- Insert – insert rows into a table
- Update – update existing rows in a table.
- Delete – delete rows from a table.
- Replace – insert a new row or replace the existing row in a table.

## Joining tables

- SQLite join – learn the overview of joins including inner join, left join, and cross join.
- Inner Join – query data from multiple tables using the inner join clause.
- Left Join – combine data from multiple tables using the left join clause.
- Cross Join – show you how to use the cross join clause to produce a cartesian product of result sets of the tables involved in the join.
- Self Join – join a table to itself to create a result set that joins rows with other rows within the same table.
- Full Outer Join – show you how to emulate the full outer join in the SQLite using left join and union clauses.

## Constraints

- Primary Key – show you how to define the primary key for a table.
- NOT NULL constraint – learn how to enforce values in a column are not NULL.
- UNIQUE constraint – ensure values in a column or a group of columns are unique.
- CHECK constraint – ensure the values in a column meet a specified condition defined by an expression.
- AUTOINCREMENT – explain how the AUTOINCREMENT column attribute works and why you should avoid using it.

## SQLite tools

- SQLite Commands – show you the most commonly used command in the sqlite3 program.
- SQLite Show Tables – list all tables in a database.
- SQLite Describe Table – show the structure of a table.
- SQLite Dump – how to use dump command to backup and restore a database.
- SQLite Import CSV – import CSV files into a table.
- SQLite Export CSV – export an SQLite database to CSV files.

### **3.3 WEB APPLICATION**

A web application is a computer program that utilizes web browsers and web technology to perform tasks over the Internet.

Millions of businesses use the Internet as a cost-effective communications channel. It lets them exchange information with their target market and make fast, secure transactions. However, effective engagement is only possible when the business is able to capture and store all the necessary data, and have a means of processing this information and presenting the results to the user.

Web applications use a combination of server-side scripts (PHP and ASP) to handle the storage and retrieval of the information, and client-side scripts (JavaScript and HTML) to present information to users. This allows users to interact with the company using online forms, content management systems, shopping carts and more. In addition, the applications allow employees to create documents, share information, collaborate on projects, and work on common documents regardless of location or device.

#### **3.3.1 How a web application works**

Web applications are usually coded in browser-supported language such as JavaScript and HTML as these languages rely on the browser to render the



program executable. Some of the applications are dynamic, requiring server-side processing. Others are completely static with no processing required at the server.

The web application requires a web server to manage requests from the client, an application server to perform the tasks requested, and, sometimes, a database to store the information. Application server technology ranges from ASP.NET, ASP and ColdFusion, to PHP and JSP.

Here's what a typical web application flow looks like:

1. **User** triggers a request to the **web server** over the **Internet**, either through a web browser or the application's user interface
2. **Web server** forwards this request to the appropriate **web application server**
3. **Web application server** performs the requested task – such as querying the **database** or processing the data – then generates the results of the requested data
4. **Web application server** sends results to the **web server** with the requested information or processed data
5. **Web server** responds back to the client with the requested information that then appears on the user's display.

### **3.3.2 Benefits of a web application**

- Web applications run on multiple platforms regardless of OS or device as long as the browser is compatible
- All users access the same version, eliminating any compatibility issues
- They are not installed on the hard drive, thus eliminating space limitations
- They reduce software piracy in subscription-based web applications (i.e. SaaS)
- They reduce costs for both the business and end user as there is less support and maintenance required by the business and lower requirements for the end user's computer.

Increased Internet usage among companies and individuals has influenced the way businesses are run. This has led to the widespread adoption of web applications as companies shift from traditional models to cloud-based and grid models. Web applications give businesses the ability to streamline their operations, increase efficiency, and reduce costs.

These online apps such as email clients, word processors, spreadsheets, and other programs provide the same functionality as the desktop versions. However, they have an added advantage of working across multiple platforms, having a broader reach, and being easily accessible from anywhere.

## CHAPTER-4

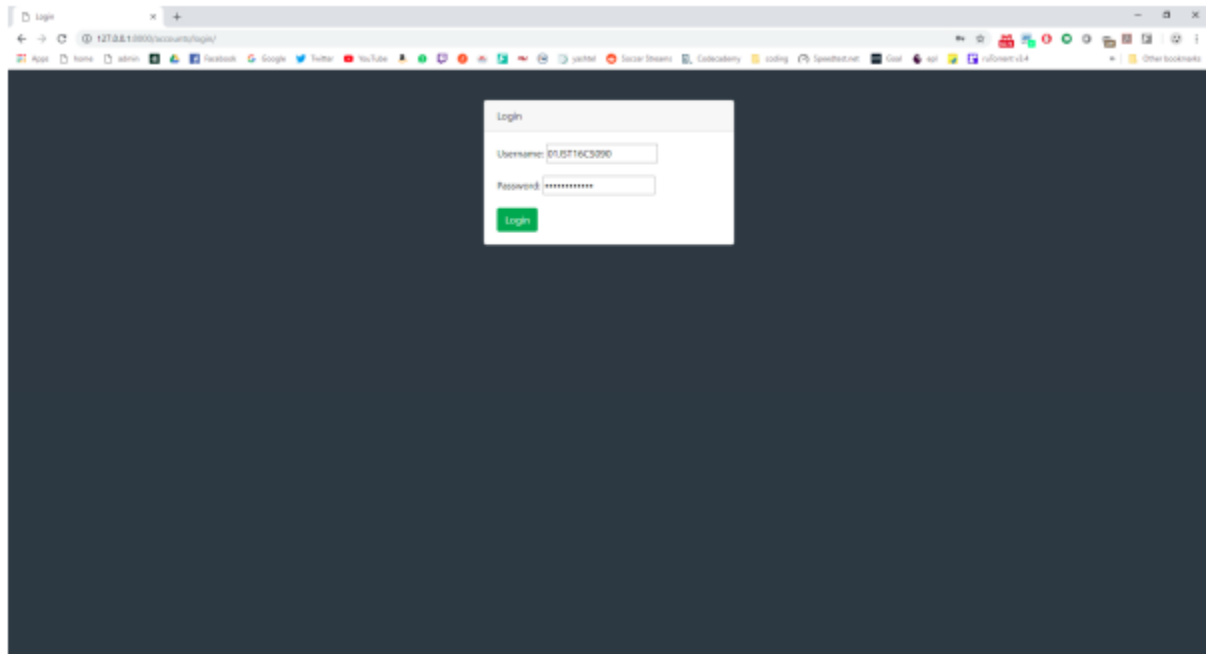
### WORKING OF PROJECT

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

#### **4.1 Student**

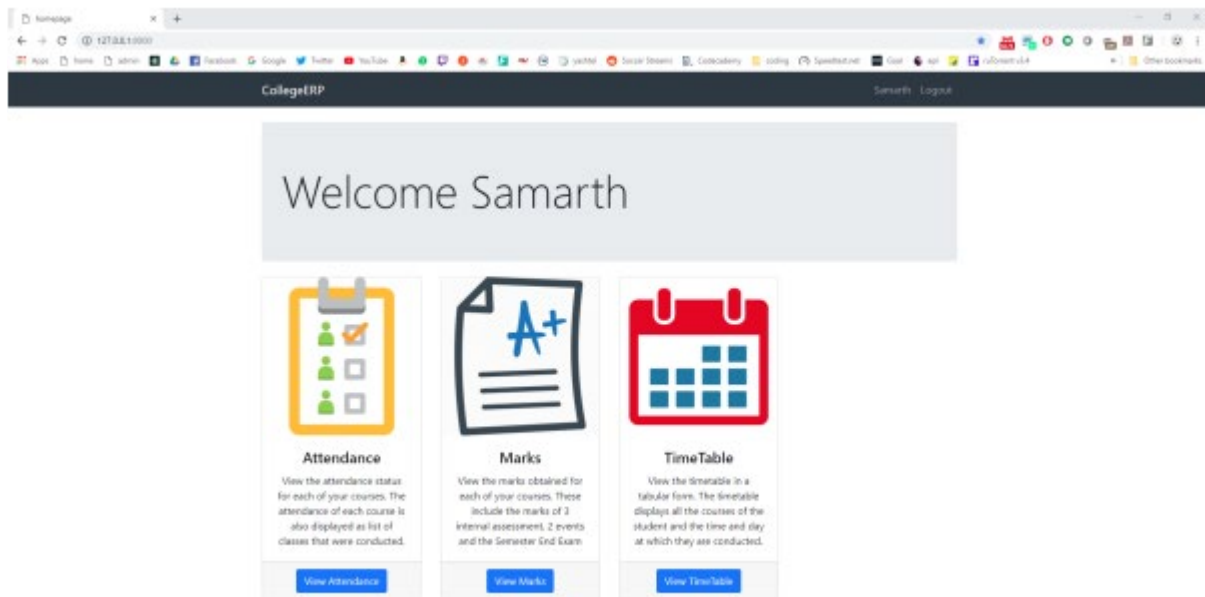
##### **4.1.1 Login**

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.



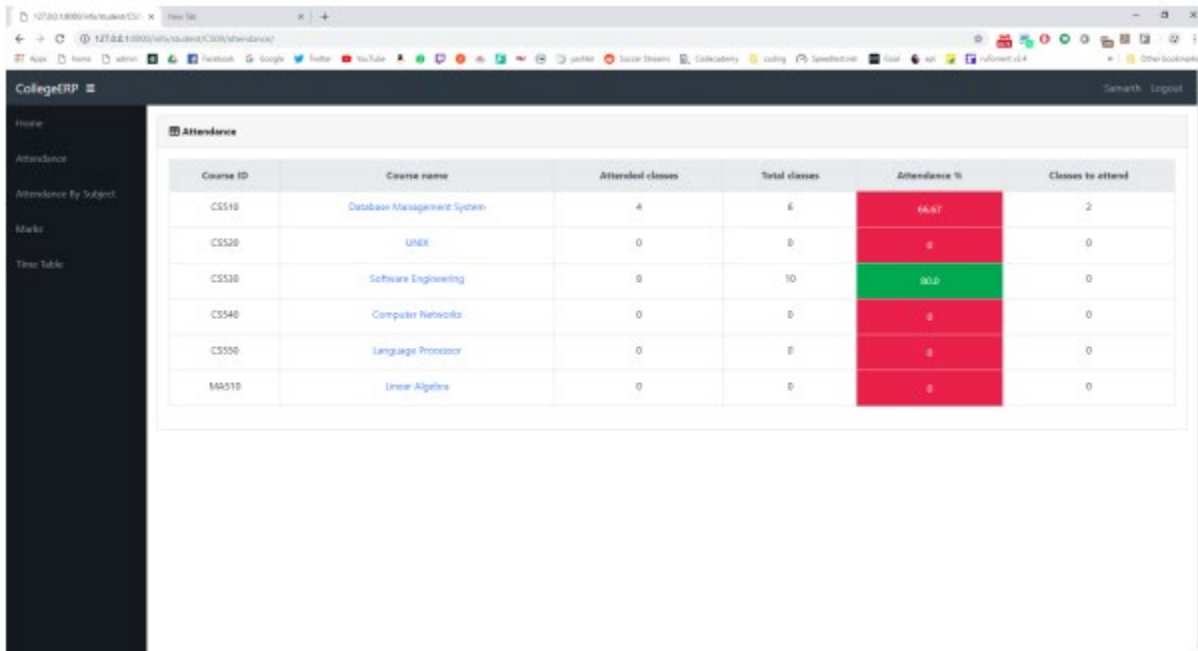
## 4.1.2 Homepage

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.



### 4.1.3 Attendance

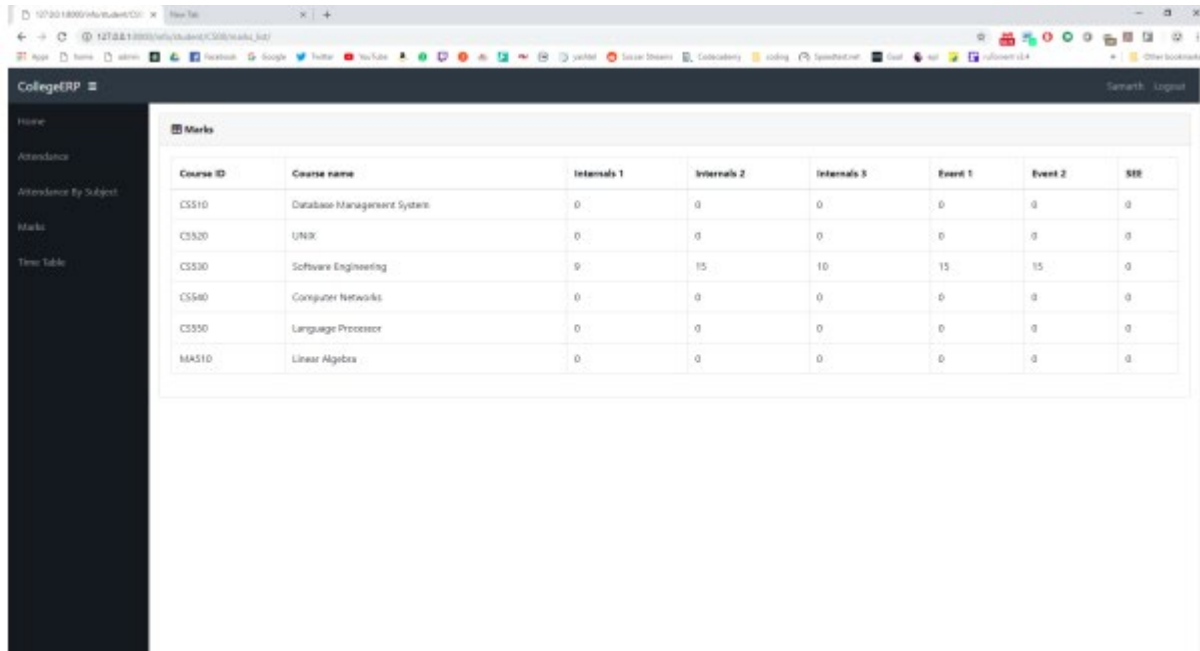
On the attendance page, there is a list of courses that is dependent on each student. For each course, the course id and name are display along with the attended classes, total classes and the attendance percentage for that particular course. If the attendance percentage is below 75 for any course, it 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.



Course ID	Course name	Attended classes	Total classes	Attendance %	Classes to attend
CS118	Database Management System	4	6	66.67	2
CS208	UNIX	0	0	0	0
CS338	Software Engineering	0	10	0%	0
CS348	Computer Networks	0	0	0	0
CS398	Language Processor	0	0	0	0
MA518	Linear Algebra	0	0	0	0

## 4.1.4 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.



The screenshot shows a web browser displaying the 'CollegeERP' interface. The main content area is titled 'Marks' and contains a table with the following data:

Course ID	Course name	Internals 1	Internals 2	Internals 3	Event 1	Event 2	SSE
CS510	Database Management System	0	0	0	0	0	0
CS520	UNIX	0	0	0	0	0	0
CS530	Software Engineering	0	15	10	15	15	0
CS540	Computer Networks	0	0	0	0	0	0
CS550	Language Processor	0	0	0	0	0	0
MA510	Linear Algebra	0	0	0	0	0	0

## 4.1.5 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.

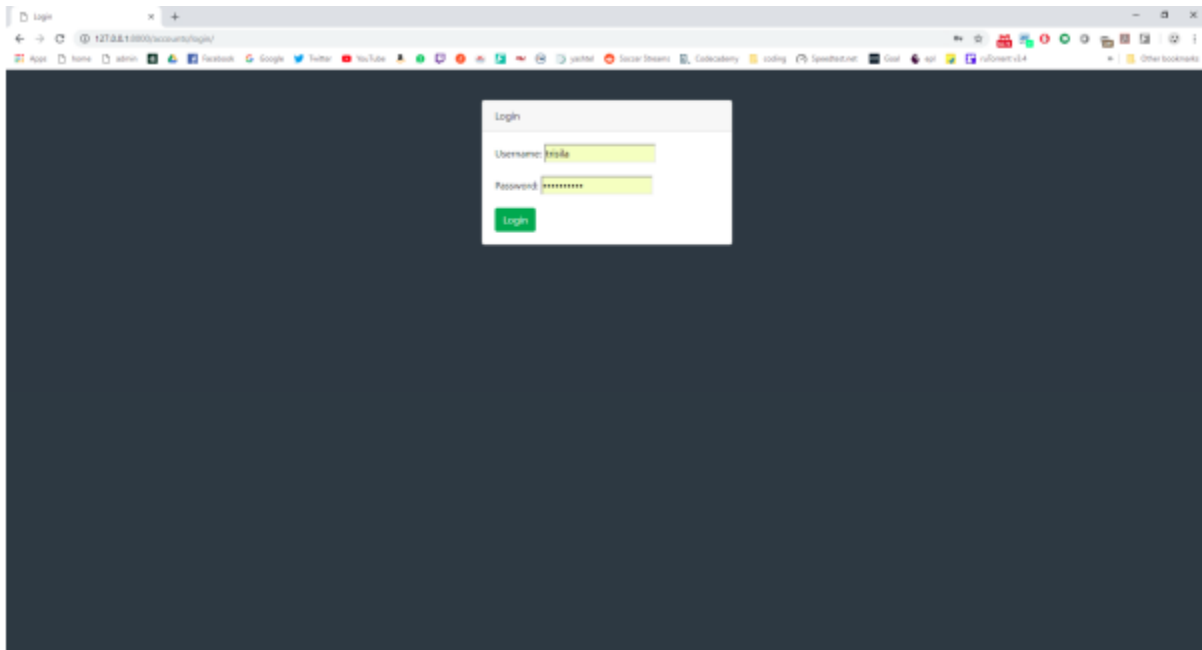
**Timetable**

	7:30 - 8:30	8:30 - 9:30	9:30 - 10:30	Break	11:00 - 11:50	11:50 - 12:40	12:40 - 1:30	Lunch	2:30 - 3:30	3:30 - 4:30	4:30 - 5:30
<b>Monday</b>			CS540		CS510	CS530	CS550				
<b>Tuesday</b>					MA510	CS510	CS540		CS520	CS550	
<b>Wednesday</b>					MA510	CS520	CS530				
<b>Thursday</b>									CS540	CS530	CS510
<b>Friday</b>			MA510		CS520	CS550	CS530				
<b>Saturday</b>		MA510	CS510		CS540	CS520	CS550				

## 4.2 Teacher

### 4.2.1 Login

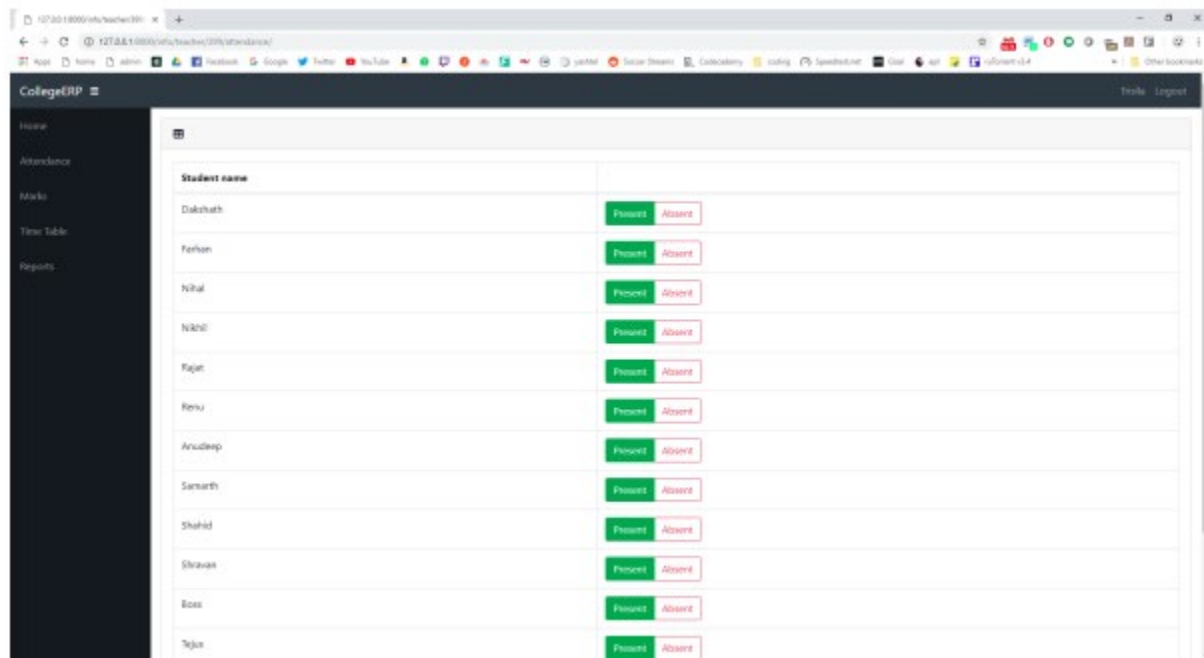
Each teacher in the college is assigned a unique username and password by the administrator. The username is their teacher ID and the same for password. The teacher may change the password later.





## 4.2.2 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.



The screenshot shows a web browser window displaying the 'CollegeERP' interface. The left sidebar contains navigation options: Home, Attendance, Marks, Time Table, and Reports. The main content area shows a table with the following data:

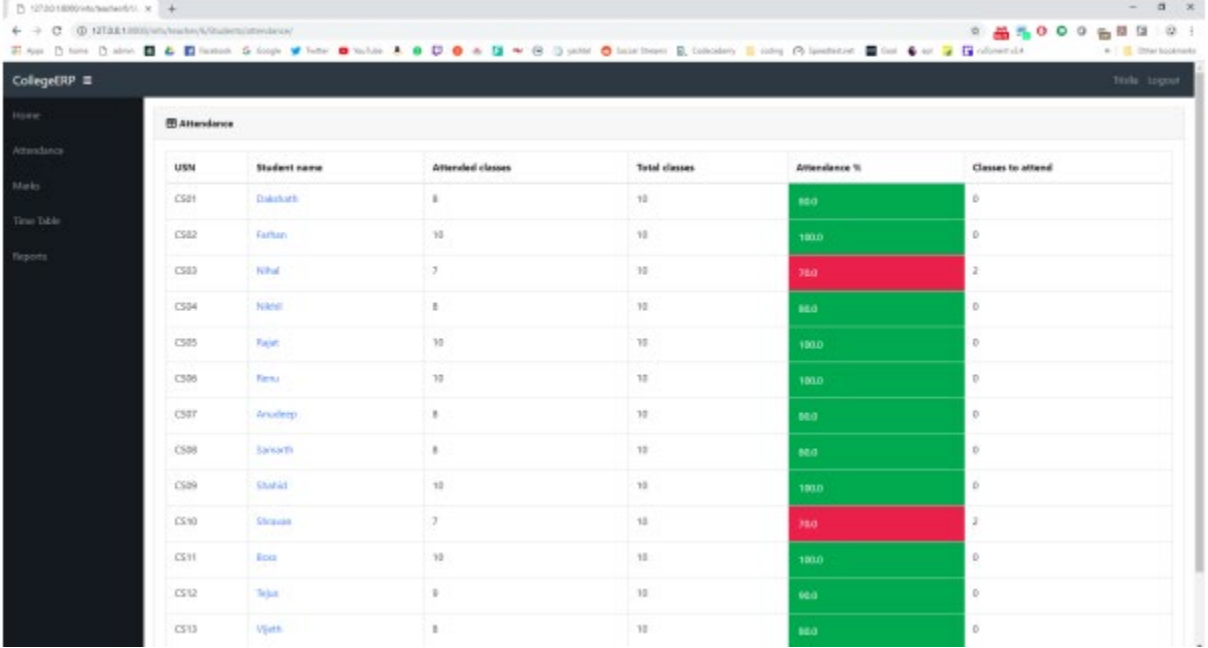
Student name	Present	Absent
Dakshath	Present	Absent
Ferhan	Present	Absent
Nihal	Present	Absent
Nikhil	Present	Absent
Rajat	Present	Absent
Renu	Present	Absent
Arundep	Present	Absent
Samarth	Present	Absent
Shahid	Present	Absent
Shravan	Present	Absent
Rishi	Present	Absent
Tyke	Present	Absent

## Extra Class

If a teacher has taken a class other than at the scheduled timings, they may enter the attendance for that as well. While entering the extra class, the teacher just needs to specify the date it was conducted and enter the attendance of each of the students. After submitting extra class, it will appear in the list of conducted classes and thus, it can be edited.

## Student Attendance

For each assigned class, the teacher can view the attendance status of the list of students. The number of attended classes, total number of classes conducted and the attendance percentage is displayed. If the attendance percentage of any of the students is below 75, it will be displayed in red. Thus, the teacher may easily find the list of students not eligible to take a test.

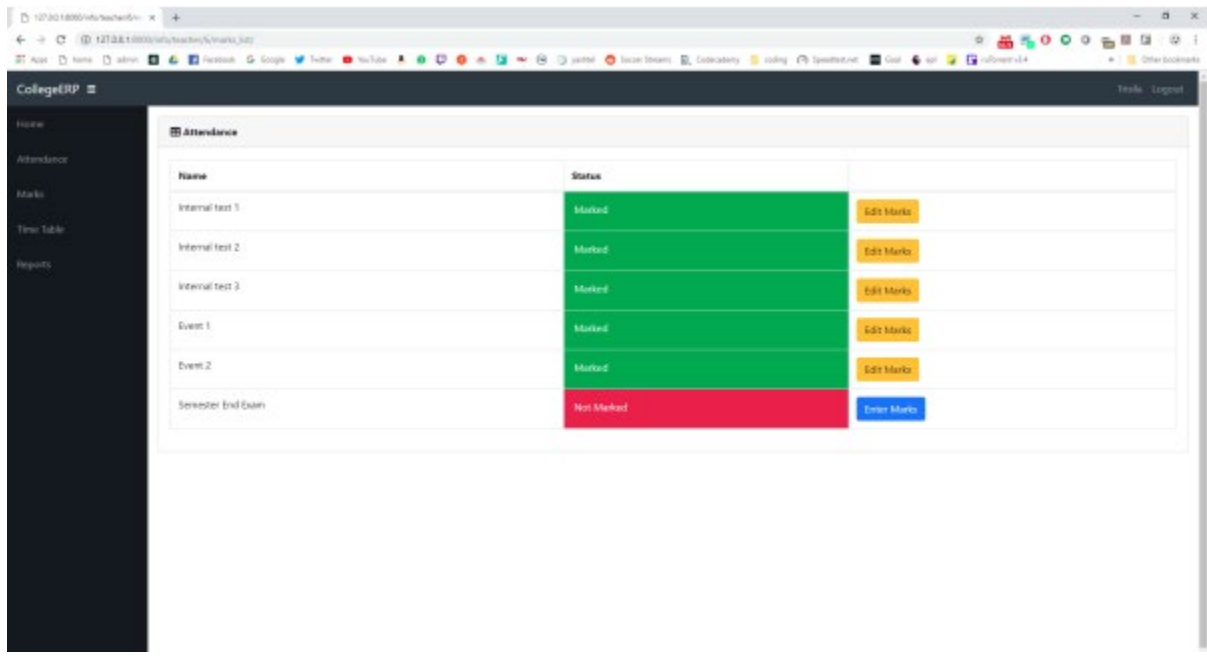


The screenshot shows a web browser displaying the 'CollegeERP' interface. The main content area is titled 'Attendance' and contains a table with the following columns: USN, Student name, Attended classes, Total classes, Attendance %, and Classes to attend. The table lists 13 students (USN CS01 to CS13). The 'Attendance %' column is color-coded: green for 80.0% or higher, and red for 70.0%. The 'Classes to attend' column shows 0 for all students except CS03 and CS10, which show 2.

USN	Student name	Attended classes	Total classes	Attendance %	Classes to attend
CS01	Dakshin	8	10	80.0	0
CS02	Fathin	10	10	100.0	0
CS03	Nihal	7	10	70.0	2
CS04	Nihal	8	10	80.0	0
CS05	Rajni	10	10	100.0	0
CS06	Renu	10	10	100.0	0
CS07	Anulekya	8	10	80.0	0
CS08	Sakshith	8	10	80.0	0
CS09	Shradha	10	10	100.0	0
CS10	Shravan	7	10	70.0	2
CS11	Easa	10	10	100.0	0
CS12	Tejas	8	10	80.0	0
CS13	Vijeth	8	10	80.0	0

### 4.2.3 Marks

On this page, the list of classes assigned to the teacher are displayed along with two actions for each class. These actions are, 40 Enter Marks On this page, the teacher can enter the marks for 3 internal assessments, 2 events and one semester end exam. Initially all of them are marked red to denote that the marks have not been entered yet. Once the marks for a test is entered, it turns green. While entering the marks for a particular test, the list of students in that class is listed and marks can be entered for all of them and submitted. Once, the marks are submitted, the students can view their respective marks. Incase if there is a need to change the marks of any student, it is possible to edit the marks



### Edit Marks

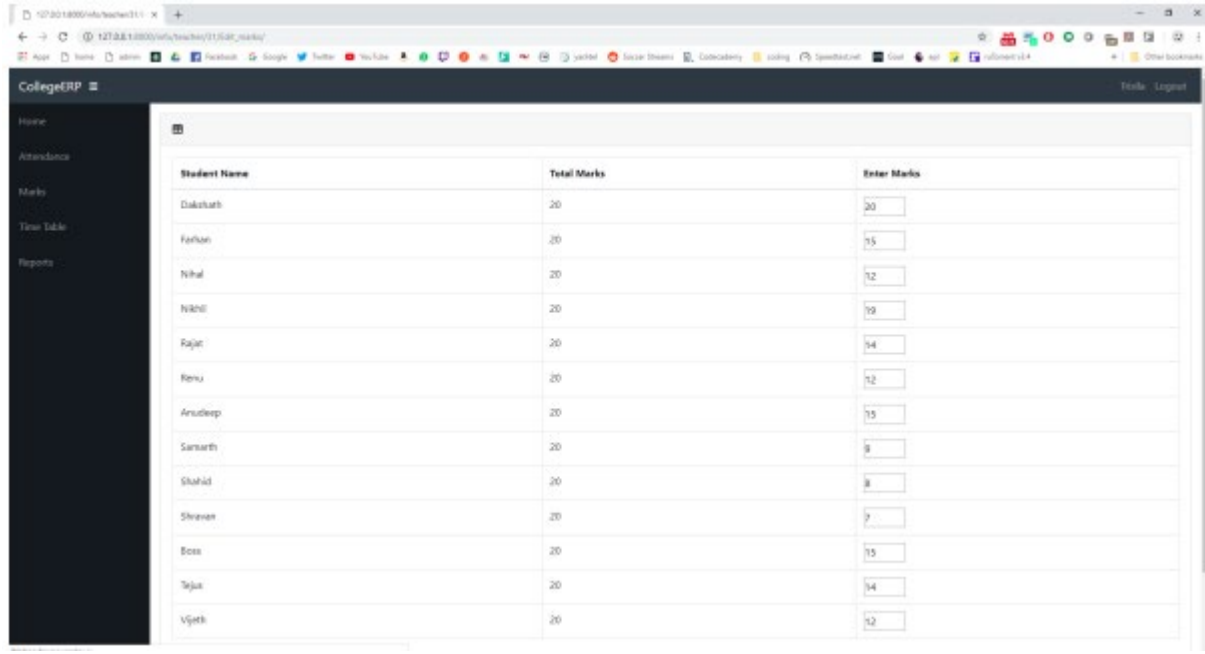
Marks for a test can be edited. While editing, the list of students in that class is displayed along with already entered marks. The marks to be updated can be changed and submitted. The students can view this change immediately.

### Student Marks

For each assigned class, the teacher has access to the list of students and the marks they obtained in all the tests. This is displayed in a tabular form.

## 4.2.4 Timetable

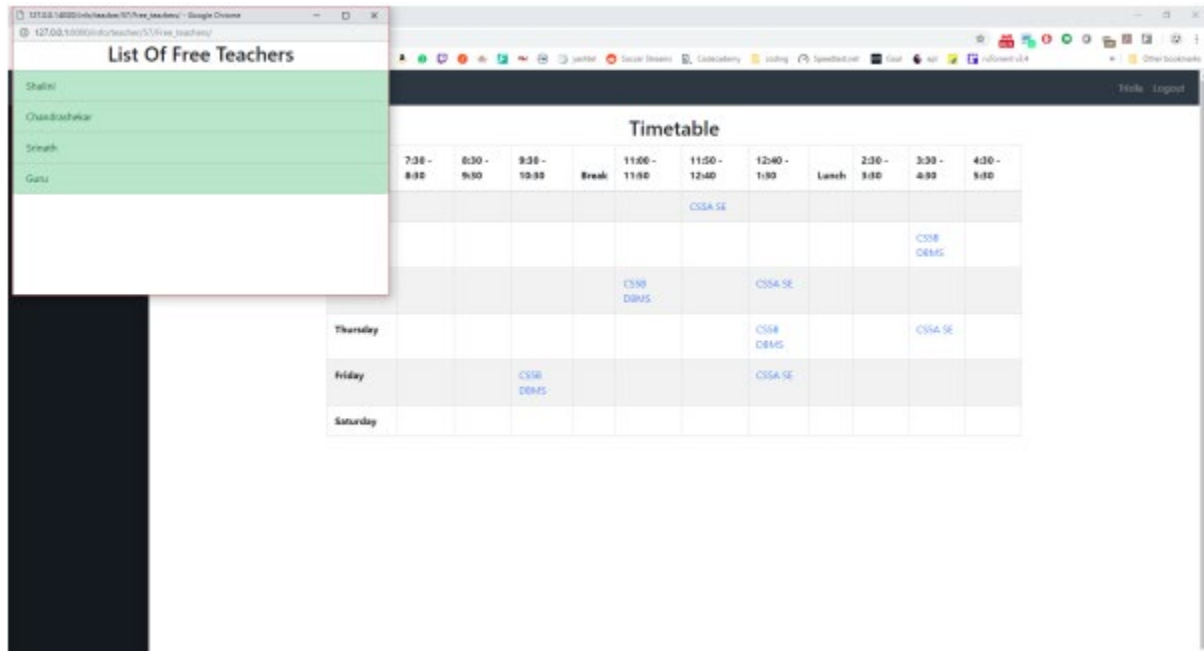
This page is a table which lists the day and timings of each of the classes assigned to the teacher. The row headers are the days of the week and the column headers are the time slots. So, for each day, it specifies the classes in the time slots. The timetable is generated automatically from the assign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



Student Name	Total Marks	Enter Marks
Dakshabh	20	20
Faizan	20	15
Nihal	20	12
Nishit	20	19
Rajit	20	14
Renu	20	12
Anudheep	20	15
Sansarth	20	8
Shahid	20	8
Shreya	20	7
Esan	20	15
Tejus	20	14
Vijeth	20	12

## Free teachers

For each entry in the table, the list of free teachers can be generated. Free teachers are the teachers who assigned to the class and are free for that time slot on that day. This is very useful for the teachers particularly when they are on leave as it helps them find a suitable replacement for that class.



The screenshot shows a web browser window with a 'List Of Free Teachers' modal open over a 'Timetable' table. The modal lists four teachers: Shashi, Chandrabhakar, Srinath, and Ganu. The timetable table shows the following data:

	7:30 - 8:30	8:30 - 9:30	9:30 - 10:30	Break	11:00 - 11:30	11:30 - 12:40	12:40 - 1:00	Launch	2:30 - 3:00	3:30 - 4:30	4:30 - 5:00
						CSSA SE				CSSB DEMS	
					CSSB DEMS		CSSA SE				
Thursday							CSSB DEMS			CSSA SE	
Friday			CSSB DEMS				CSSA SE				
Saturday											

## 4.3 Administrator

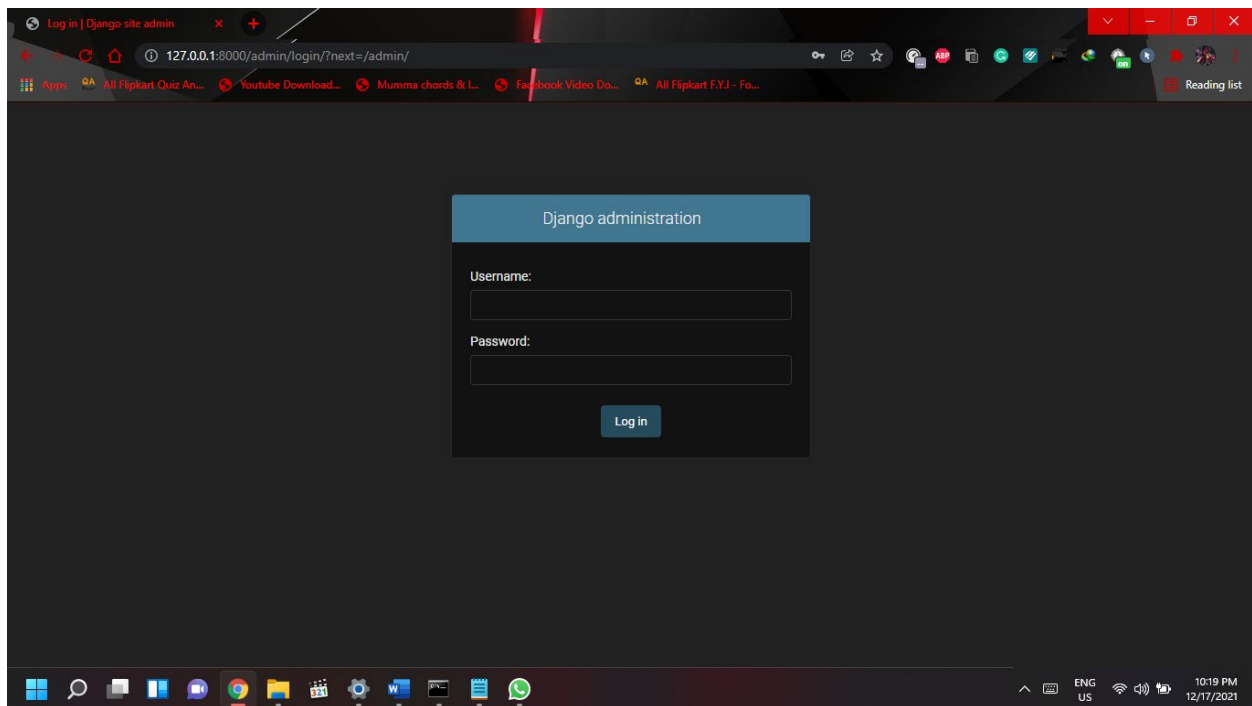
### 4.3.1 Login

We can create an administrator profile from the Django backend by using the given command

```
“python manage.py createsuperuser”
```

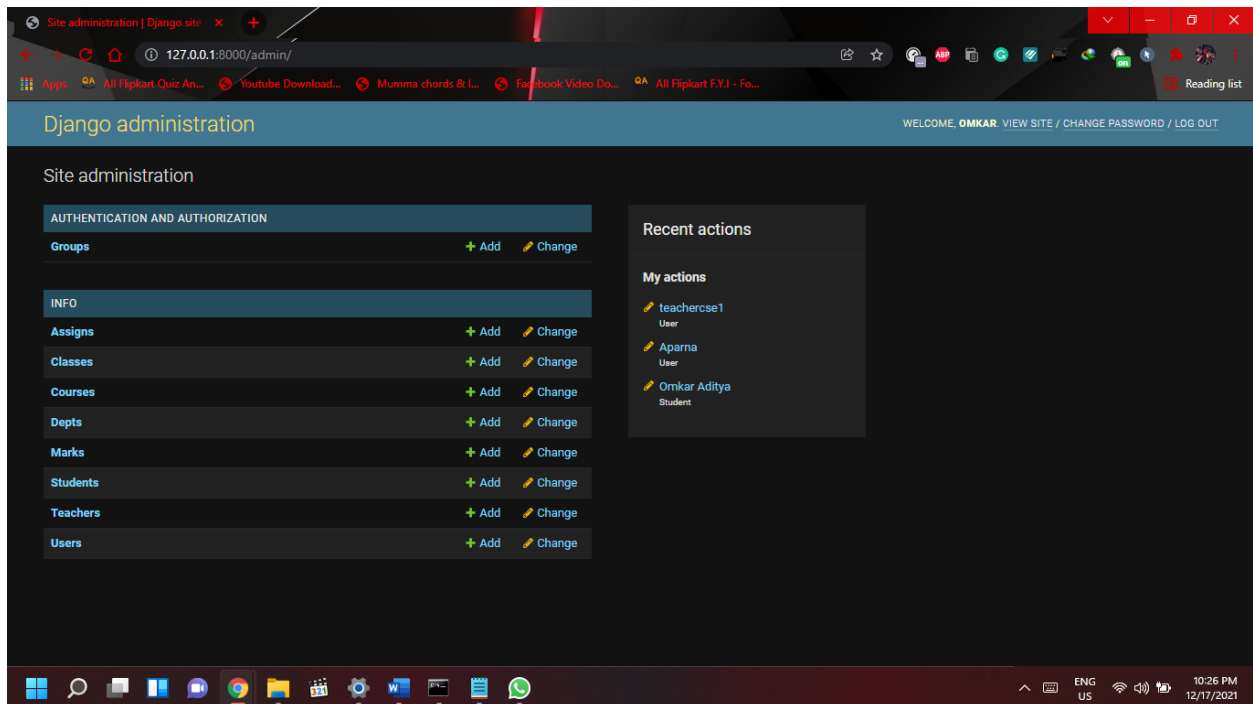
We can define the username and password of the super user there. After setting the details, we can access the administrator page by visiting the <http://127.0.0.1:8000/admin> page.

There is no option of forgot password yet because the server is not connected to the cloud yet. The account is only based on local hosted server.



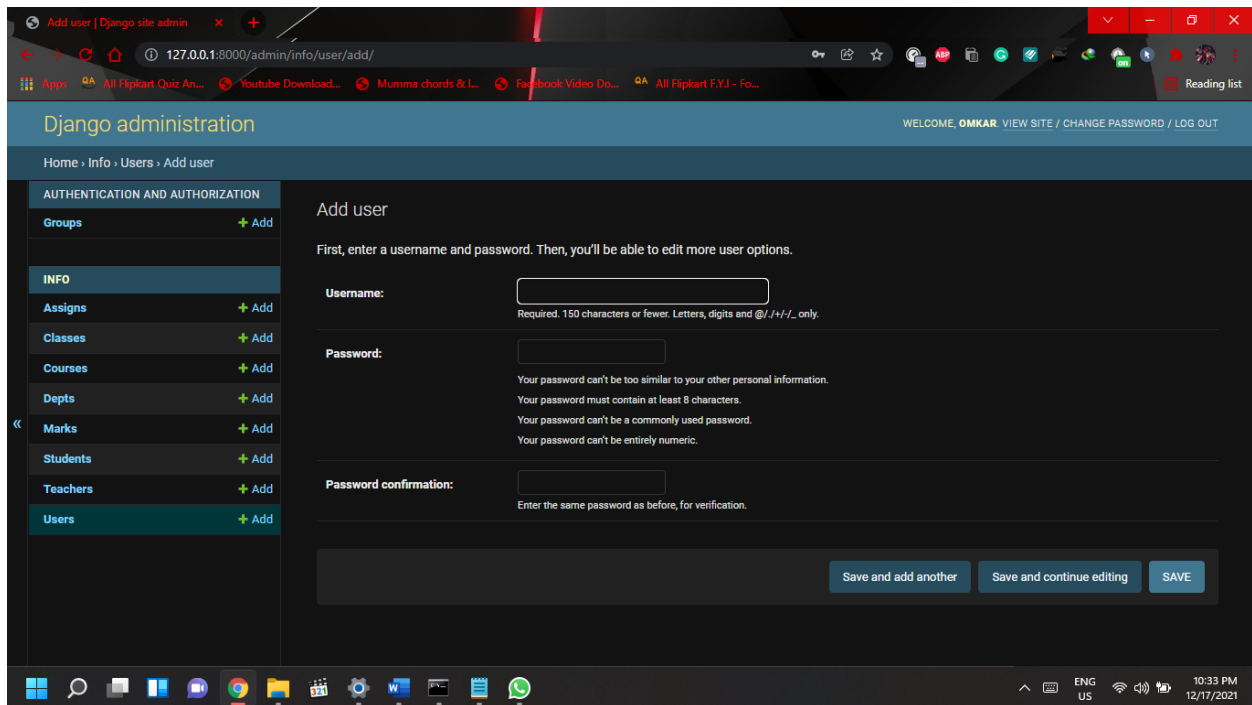
### 4.3.2 Homepage

After successful login, the administrator is presented a homepage giving them access of adding all information and modifying it as well. Administrator can add users and allot them classes. Administrator can create multiple departments in the organization and add professors in the respective courses and department. The administrator allocates the classes to the teachers and add students in the classes as well. Administrator has access to the backend of all teachers and student departments. He can modify attendance, marks as well, assign teachers to free classes, modify time table as well. These are the features in the homepage given to the administrator of the ERP.



### 4.3.3 Users

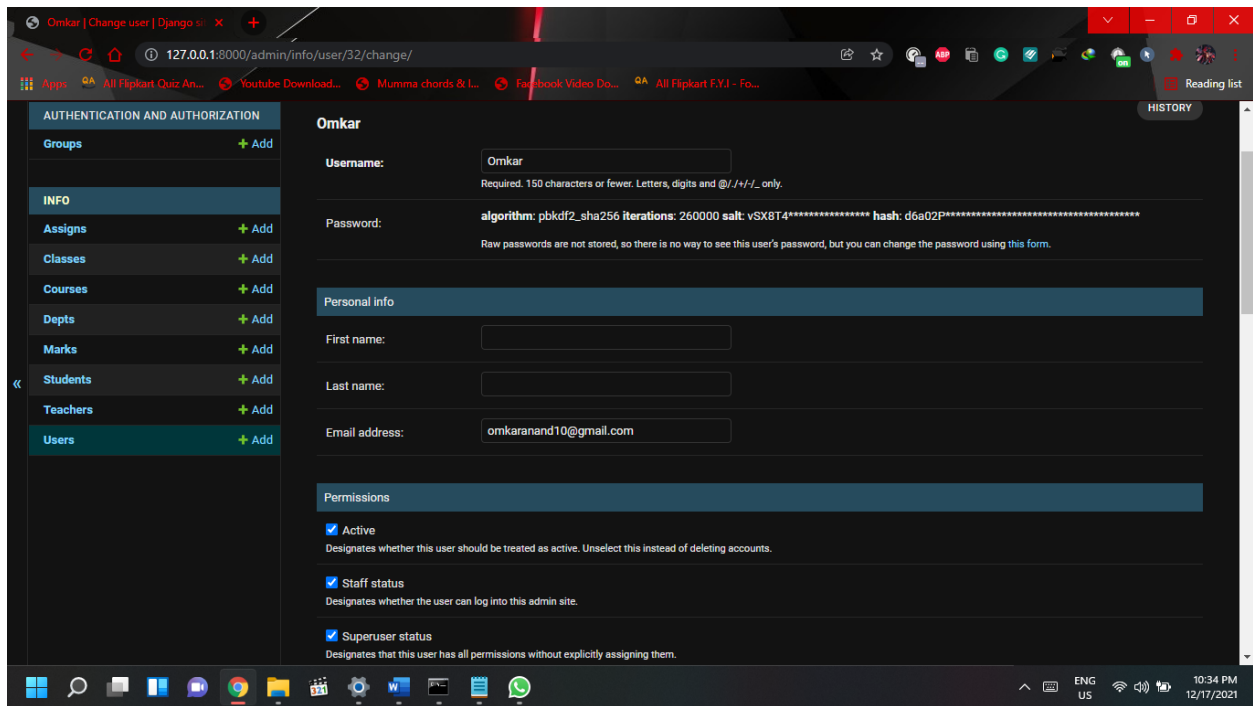
This is the first step to use the ERP i.e. Adding users in the University portal. We can add users in the server database by using the add users features. We have to assign the username which will be their university roll number and teachers ID in the case and the same as default password which they can change later by accessing their account.





## User List

We can see the list of users as well we have added in the system and categorize them by their status as students/teachers/super user. We can add their basic information in the system as well as their first and last name, phone numbers, email id for their account linking and recovery. And for keeping personal privacy of users, the administrator can't see their password but if the account face any issue, the password can be reset to default by the administrator. And for load sharing and ease of work, we can even assign specific permission to faculties and students so that administrator doesn't have to do everything.



## 4.4 Instructions

Go to the College-ERP folder and run

```
python manage.py runserver
```

Then go to the browser and enter the url <http://127.0.0.1:8000/>

You can access the django admin page at <http://127.0.0.1:8000/admin>

Also a new admin user can be created using

```
python manage.py createsuperuser
```

To end server press Ctrl+C

```
Administrator: Command Prompt - python manage.py runserver
Microsoft Windows [Version 10.0.22519.1012]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>cd .
C:\Windows>cd .
C:\>cd "College ERP"
C:\College ERP>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified some issues:

WARNINGS:
Info.Assign: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Info.AssignTime: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Info.Attendance: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Info.AttendanceClass: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Info.AttendanceTotal: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Info.Marks: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Info.MarksClass: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Info.StudentCourse: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Info.User: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
      HINT: Configure the DEFAULT_AUTO_FIELD setting or the InfoConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.

System check identified 9 issues (0 silenced).

You have 1 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): auth.
Run 'python manage.py migrate' to apply them.
December 17, 2021 - 23:29:18
Django version 3.2.8, using settings 'CollegeERP.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

[Taskbar icons: Start, Search, File Explorer, Edge, Teams, Chrome, OneDrive, Outlook, Word, Excel, PowerPoint, Teams, WhatsApp, Task View, Settings, Network, ENG US, 11:20 PM 12/17/2021]
```

## CHAPTER-5

### RESULT ANALYSIS

The completion of a system will be achieved only after it has been thoroughly tested. Though this gives a feel the project is completed, there cannot be any project without going through this stage. Hence in this stage it is decided whether the project can undergo the real time environment execution without any break downs, therefore a package can be rejected even at this stage.

#### **5.1 Testing methods**

Software testing methods are traditionally divided into black box testing and white box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.

##### **5.1.1 White Box Testing**

White box testing, by contrast to black box testing, is when the tester has access to the internal data structures and algorithms (and the code that implement these). White box testing methods can also be used to evaluate the completeness of a test suite that was created with black box testing methods. This allows the software team to examine parts of a system that are rarely tested and ensures that the most important function points have been tested. This project is implemented using python with the Django framework. The code consists of models and views which can be tested. Models define the tables stored in SQL and the relationship between the different tables using foreign keys. A view function, or “view” for short, is simply a Python function that takes a web request and returns a web response. This response can be the HTML contents of a Web page, or a redirect, or a 404 error, or an XML document, or an image, etc. Python also provides a file called test.py where we can write unit tests for the models and views. This is very useful as it automates the testing and we no longer have to manually test every page after there were any changes.

The python code is pasted below and each test is explained using comments in the code. from Django. Test import TestCase from info.models import Dept, Class, Course, User, Student, Teacher, Assign, Attendance from django.urls

```
import reverse from django.test.client import Client class InfoTest(TestCase):  
# function used to create test users
```

### **5.1.2 Black Box Testing**

Black box testing treats the software as a “black box,” without any knowledge of internal implementation. Black box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing and specification-based testing. We performed black box testing on the teacher page to make sure every page was working as desired. We took into consideration various test cases and noted down the results. Below we have recorded various test cases and their respective results.

#### **Test Case: 1**

Request the attendance page for a teacher with no assigned classes. The web page loaded with message “Teacher has no classes assigned”.

#### **Test Case: 2**

Request the attendance page for a teacher with 1 assigned class. The web page displayed the assigned class and options to enter attendance and view the students.

#### **Test Case: 3**

Request to enter the attendance for an assigned class with one test student 50 The web page displays the student with his/her details and an option to mark present or absent. On marking absent, it can be viewed by the student.

#### **Test Case: 4**

Request to edit the attendance for an assigned class with one test student The student is listed with his/her details and is initially marked as absent from the previous test. On marking present, the attendance for that student and can be viewed by the student.

#### **Test Case: 5**

Request to enter the marks for an assigned class with one student Initially, a list of tests is displays such as internals 1, SEE etc. On selecting one of internals 1, the teacher can enter the marks for the student out of 20. On submitting, the

status for that test turns green denoting that it has been successfully entered.

### **Test Case: 6**

Request to edit the marks for an assigned class with one student. For each class, there is a list of tests such as internals 1, SEE etc. As the marks for internals 1 was already entered in the previous test, it is marked green and there is an option to edit. When editing, the marks already stored is displayed and appropriate changes can be made and saved.

### **Test Case: 7**

Request to view the student information for an assigned class with no students. The requested page is displayed with no content and a message stating “This class has no students assigned”

### **Test Case: 8**

Request to view the student information for an assigned class with 1 student. The web page is the form of a table with entries for student name, USN and their attendance percentage, marks in each test including 3 internals, 2 events and 1 SEE. If the attendance status is below 75%, it is marked in red.

## **5.2 Results of testing**

After applying various testing methods such as black box testing, white box testing and acceptance testing, we can conclude that the testing for the software is completed. To summarize the testing phase, white box testing is done using the inbuilt feature of Django to apply unit tests to all the components in the software. After any changes to the software, we can run the tests on the software automatically and thus we can find and eliminate any bugs or errors in the system easily instead of performing rigorous manual testing after every change. In black box testing, we testing all the components and system as a whole. Several test cases were considered and extensive tests were conducted. The results of these tests were positive and any errors were fixed during the testing phase. For acceptance testing, we gave a demonstration of the software to our teacher, who is a key stakeholder. After several tests and questions, she was content with results of the tests and software.

We have already checked all the modules in the ERP by rigorous testing methods and we can say that our ERP can handle all type of work done in a University

without any issues. It is the very useful to the student as well as the faculties to easy access to finding the details. This project is successfully implemented with all the features and modules of the college management system as per requirements. The college ERP provides appropriate information to users based on their profiles and role in the system. This project is designed keeping in view the day to day problems faced by a college system.

## CHAPTER-6

### 6.1 CONCLUSION

A cloud-based university management system (UMS) is a digital solution that helps automate tasks. Ranging from registering students, admitting students, onboarding them, managing their fees, managing payroll for lecturers to helping students access online learning, classroom training etc. a university management system is a complete solution. Reducing manual labor and intervention, this system ensures accuracy, transparency, reliability, and integrity of records, information, intellectual property and data.

A UMS is an education ERP product that helps teachers, students, parents, external vendors, learning suppliers etc. come together via a platform and exchange information. The university administration can automate several of their processes such as attendance, announcements, results, campus updates, schedule changes etc. to all concerned persons. There is no limit on the number of features of the university management system, because every university will have its own set of processes that is different from another university. Therefore, a UMS should be able to help universities apply university-specific customizations.

College ERP system provides a simple interface for maintenance of different student, department, faculties, library and others information. All the colleges usually have a number of departments and educational modules such as courses, seminar hall, etc. Managing all these departments and other modules manually is a very difficult and hard, ineffective and expensive task. So here we propose an ERP system for college.

Our college ERP system has all the information about the students, teachers, events, library, departments and other respected information. The system allows the admin to add students, faculties and any other events. Our system allows a faculty to enter or input student's attendance into the database which can later be viewed by students and faculties. The students can view his/her attendance through a separate student login. The admin can upload the timetables for various departments for exam preparation. The time table is then available to be viewed by faculties and students on the web portal. These systems have easy user interface and have powerful data management system which makes this system is very

useful.

## **6.2 FUTURE WORK**

- **BI-powered ERP**

The future will most likely see a rush in demand for higher education BI-powered ERPs. Using business intelligence dashboards, it will be easy to track key performance indicators, real-time metrics, or any other key student data that could be easily tracked, allowing teachers to make smart decisions with all the information required. By using an ERP institution can track attendance and studying reports generated to help faculties and students to have better learning platforms.

- **Automation of complete activities**

Automation is one of the most important functions that an ERP provides, a higher education ERP automates the complete day-to-day tasks. Students in today's era demand easier access to everything like marking attendance, fetching timetables and all other tasks, and predicting the future. There are more chances that students will only like institutes that will provide their complete automation and real-time access to things from admission form to result in the announcement.

- **Advances in technology – teaching and learning patterns**

Opting for cloud-based higher education management software will allow institutions to adopt advances in technology that can help in teaching and learning patterns. With all these, it is also easier to predict that one can anticipate more options and more advanced features using e-learning ERPs in the future.

- **Customization and personalization**

According to the Education ERP Market Size and Forecast, growing demand for modern learning has improved the level of the education system, improved performance of the academic and administrative department, and enhanced connectivity have made ERP essential for higher education institutions. It is not wrong to estimate that modern trends and technologies are improving the chances of incredible future growth of institutions using ERP. A good and robust ERP provides the personalization and customization of tools and technologies that are needed by the institutions.

Commenting on the personalization of systems using ERP, I can conclude that an ERP will get more powerful having stronger implementation and customization of



each institution's process but for these, institutions must choose a flexible, robust, and well-integrated ERP system.

- **Better communication**

ERP software can help students and faculty to learn properly bridging the gap between them. It helps institutions with updating all the lessons, assignment details, faculty's availability and other related information, managing attendance, enrollment tasks, examinations, and all other tasks managing better communication.

## REFERENCE

1. Elmasri and Navathe: Fundamentals of Database Systems, 7th Edition, Pearson Education, 2016.
2. Ian Sommerville: Software Engineering, 10th edition, Person Education Ltd, 2015.
3. Roger S Pressman: Software Engineering- A Practitioners approach,8th edition, McGraw-Hill Publication, 2015.
4. <https://en.wikipedia.org/wiki/Requirements-engineering>
5. <https://web.cs.dal.ca/~hawkey/3130/srs-template-ieee.doc>
6. <http://www.ntu.edu.sg/home/cfcavallaro/Reports/Report%20writing.htmTop>
7. [https://en.wikipedia.org/wiki/Class diagram](https://en.wikipedia.org/wiki/Class_diagram)
8. <https://www.djangoproject.com/>
9. <https://getbootstrap.com/>
10. <https://www.tutorialspoint.com/>
11. <https://creately.com/>
12. <https://www.overleaf.com/project>