DUCoE Digital Class Management System

**A Project Report**

**In Partial fulfillment for the award of the degree of**

**Bachelor of Technology in**

**Computer Science and System Engineering**

**At**

****

**DEFENCE UNIVERSITY, COLLEGE OF ENGINEERING**

**BISHOFTU**

**NOVEMBER 2021 GC**

**DUCoE Digital Class Management System**

**A Project Report**

**By:**

1. ***Esubalew Gizaw-----------------------RDEG/142/09***
2. ***Ephrem Gemechu---------------------RDEG/442/09***
3. ***Zelalem Endeg-------------------------RDEG/480/09***
4. ***Abdi Birhanu --------------------------RDEG/401/09***
5. ***Mehamednur Bekele ---------------RDEG/151/09***

*Under the Guidance of*

***Capt. Wondimagegn Nigussie***

**In Partial fulfillment for the award of the degree of**

**Bachelor of Technology In**

**Computer Science and System Engineering**

**At**

****

**DEFENCE UNIVERSITY, COLLEGE OF ENGINEERING**

**BISHOFTU**

**November 2021GC**

DECLARATION

This is to declare that the project entitled **“DUCoE Digital Class Management System.”** submitted to Defense University College of Engineering department of Computer and Information Technology in partial fulfillment of the requirement for the award of the degree of Bachelor of technology in Computer Science and System Engineering is an original work carried out by ourselves

Student Team Approval

|  |  |
| --- | --- |
| **Student Name** | **Signature** |
| 1. Esubalew Gizaw | \_\_\_\_\_\_\_\_ |
| 2. Ephrem Gemechu | \_\_\_\_\_\_\_\_ |
| 3. Zelalem Endeg | \_\_\_\_\_\_\_\_ |
| 4. Abdi Birhanu | \_\_\_\_\_\_\_\_ |
| 1. Mehamednur Bekele | \_\_\_\_\_\_\_\_ |

CERTIFICATE

This is to certify that the project entitled **“DUCoE Digital Class Management System.**“ is the work carried out by Computer Science and Systems Engineering students of BTech, Defense Engineering College, Bishoftu, during the year 2021 in partial fulfillment of the requirement for the award of the Degree of Bachelor of Computer Science and Systems Engineering is an original work carried out by Esubalew Gizaw ,Ephrem Gemechu, Zelalem Endeg, Abdi Birhanu and Mehamednur Bekele under my guidance. The work in this project is reliable and is genuine work done by the students and has not been submitted to Defense University, College of Engineering.

Project advisor Head of the DepartmentCapt. Wondimagegn Nigussie Capt. Dr. Solomon Abera

Signature \_\_\_\_\_\_\_\_\_ Signature \_\_\_\_\_\_\_\_\_

**Examiners ‘NameSignature**

|  |  |
| --- | --- |
| 1.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  2.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 3.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# ABSTRACT

*This speciﬁes the various processes and techniques used in gathering requirements, designing, implementing and testing for the project on college management system. The problems regarding the current teaching-learning system in the college were analyzed and noted. This project aims to solve some of those problems and thus, add more value to the current system. The requirements were gathered from all the stakeholders and based on that we created a requirements models and designed the software based on the based. The project was implemented in the form of a website using django (python).*

*Using the various resources and tools we gathered along the way, we implemented the Digital Class Management system using some features that solve the current problems in the system such as a provision to edit the attendance and marks before locking it at the end. The software was also tested using the various testing methods and results were positive.*

*Thus, the results can be integrated in the current teaching-learning system to improve it’s working and solve some of the existing problems.*

# 

**ACKNOWLEDGEMENT**

Foremost, we would like to express our sincere gratitude to our project advisor Capt. Wondemagegn Nigussie for the continuous support of our BTech study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped us in all the time of research and writing of this thesis. we could not have imagined having a better advisor and mentor for our BTech study.

Besides our advisor, we would like to thank our Head of CIT Department Capt. Dr. Solomon Abera for his encouragement and insightful comments.  
  
Last but not the least, we would like to thank our family for their continuous support and good wishes.

# 

# **LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **DUCoE** | Defense University College of Engineering |
| **OOD** | Object Oriented Design |
| **OOA** | Object Oriented Analysis |
| **JS** | JavaScript |
| **HTML** | Hyper Text Markup Language |
| **CSSE** | Computer Science and System Engineering |
| **CSS** | Cascading style sheet |
| **CIT**  **MTV**  **DSF**  **ML**  **AI**  **IoT**  **Apps**  **SQL** | Computer and information technology  Model template View  Django software foundation  Machine learning  Artificial intelligence  Internet of things  Applications  Structure query language |

# **Table of Contents**

[DECLARATION I](#_Toc48029073)

[CERTIFICATE II](#_Toc48029074)

[ABSTRACT III](#_Toc48029075)

[ACKNOWLEDGEMENT IV](#_Toc48029076)

[LIST OF ABBREVIATIONS V](#_Toc48029077)

[Table of Contents VI](#_Toc48029078)

[List of Figure XI](#_Toc48029079)

[List of Table XII](#_Toc48029080)

[CHAPTER ONE 1](#_Toc48029081)

[1. Introduction 1](#_Toc48029082)

[1.1. Background Information 1](#_Toc48029083)

[1.2. Statement of the problem 1](#_Toc48029084)

[1.3. Objective of the project 2](#_Toc48029085)

[1.3.1. General Objective 2](#_Toc48029086)

[1.3.2. Specific Objective 2](#_Toc48029087)

[1.4. Significance of the project 2](#_Toc48029088)

[1.5. Beneficiaries of the project 3](#_Toc48029089)

[1.6. Application area 3](#_Toc48029090)

[1.7. Methodology 3](#_Toc48029091)

[1.7.1. Data collection methodology 3](#_Toc48029092)

[1.7.2. Development Environment and programming tool 3](#_Toc48029093)

[1.7.3. Testing Procedure 4](#_Toc48029094)

[1.8. Scope of the project 5](#_Toc48029095)

[1.9. Risk, Assumption and Constraints 6](#_Toc48029096)

[1.9.1. Risk 6](#_Toc48029097)

[1.9.2. Assumptions 6](#_Toc48029098)

[1.9.3. Constraints 6](#_Toc48029099)

[1.10. Project organization 7](#_Toc48029100)

[1.11. Feasibility study 8](#_Toc48029101)

[1.11.1. Economic feasibility 8](#_Toc48029102)

[1.11.2. Technical feasibility 9](#_Toc48029103)

[1.11.3. Operational feasibility 9](#_Toc48029104)

[1.11.4*.*  Schedule feasibility 9](#_Toc48029105)

[1.11.5. Legal and Contractual feasibility 10](#_Toc48029106)

[1.12. Estimated Project Cost and time table 10](#_Toc48029107)

[1.12.1. Estimated Project Cost 10](#_Toc48029108)

[1.12.2. Work plan of the project 10](#_Toc48029109)

[CHAPTER TWO 11](#_Toc48029110)

[2. Literature Review 11](#_Toc48029111)

[2.1 Introduction 11](#_Toc48029111)

[2.2. Effectiveness of using digital class system `11](#_Toc48029112)

[2.2.1. Literature Review of other system 12](#_Toc48029113)

[2.2.2*.*  PyCharm 12](#_Toc48029114)

[2.2.3. Django 13](#_Toc48029115)

[2.3. Proposed System 15](#_Toc48029116)

[2.4. Summary 16](#_Toc48029117)

[CHAPTER THREE 16](#_Toc48029118)

[3. Object Oriented Analysis and Design 16](#_Toc48029119)

[3.1. A brief history 16](#_Toc48029120)

[3.2. Object-oriented analysis 16](#_Toc48029121)

[3.3. Object-oriented design 16](#_Toc48029122)

[3.4. System use case model 17](#_Toc48029123)

[3.4.1. What is a Use case? 17](#_Toc48029124)

[3.4.2. What is a Use Case Diagram? 17](#_Toc48029125)

[3.4.3. How do you write a Use Case? 17](#_Toc48029126)

[3.4.4. Sequence Diagram 28](#_Toc48029127)

[3.4.5. Activity diagram 30](#_Toc48029128)

[3.4.6. Class diagram 34](#_Toc48029129)

[3.4.7. Purpose of Class Diagrams 34](#_Toc48029130)

[3.4.8. Deployment 35](#_Toc48029131)

[3.4.8.1. Purpose of Deployment Diagrams 35](#_Toc48029132)

[CHAPTER FOUR 36](#_Toc48029133)

[4. System Description 36](#_Toc48029134)

[4.1. Introductions 36](#_Toc48029135)

[4.1.1. Essential Navigations 36](#_Toc48029136)

[4.2. Users page 37](#_Toc48029137)

[4.2.1 Student Page 37](#_Toc48029144)

[4.2.1.1 Login 37](#_Toc48029144)

[4.2.1.2 Homepage 38](#_Toc48029144)

[4.2.1.3 Attendance 39](#_Toc48029144)

[4.2.1.4 Marks 40](#_Toc48029144)

[4.2.1.5 Time table 41](#_Toc48029144)

[4.2.1.6 Class Assignmnet 42](#_Toc48029144)

[4.2.1.7 Update Profile 43](#_Toc48029144)

[4.2.1.8 Change password 44](#_Toc48029144)

[4.2.2 Teacher Page 45](#_Toc48029144)

[4.2.2.1 Login 45](#_Toc48029144)

[4.2.2.2 Homepage 46](#_Toc48029144)

[4.2.2.3 Attendance 47](#_Toc48029144)

[4.2.2.4 marks 50](#_Toc48029144)

[4.2.2.5 Time table 53](#_Toc48029144)

[4.2.2.6 Repports 54](#_Toc48029144)

[4.2.2.7 Uppload assignment 55](#_Toc48029144)

[4.2.2.8 Write notice 56](#_Toc48029144)

[4.2.2.9 Update profile 57](#_Toc48029144)

[4.2.2.10 View assignment submitted 58](#_Toc48029144)

[4.2.2.11 Change password 59](#_Toc48029144)

[4.2.3 Admin page 60](#_Toc48029144)

[4.2.3.1 Login 60](#_Toc48029144)

[4.2.3.2 Adminstrator 61](#_Toc48029144)

[4.2.3.3 Add New User 62](#_Toc48029144)

[4.2.3.4 Change password 63](#_Toc48029144)

[4.3 User activity Discription 64](#_Toc48029144)

[4.4 Database design 65](#_Toc48029144)

[4.5 Sample code 66](#_Toc48029144)

[CHAPTER FIVE 36](#_Toc48029133)

[5. Conclusion and Recommendation 69](#_Toc48029147)

[5.1. Introduction 69](#_Toc48029148)

[5.2.Conclusion 70](#_Toc48029149)

[5.3.Recommendation 70](#_Toc48029150)

[References 71](#_Toc48029151)

# List of Figure

[Figure1-1 Project organization detail work structure with the team members. 8](#_Toc48028482)

[Figure1-2Project organization and role of team members. 8](#_Toc48028483)

[Figure3-1 A Use Case Diagram for Admin 23](#_Toc48028485)

[Figur3-2 A Use Case Diagram for Teacher 24](#_Toc48028486)

[Figure3-3 A Use Case Diagram for Student 25](#_Toc48028487)

[Figure3-4 Sequence diagram for login 39](#_Toc48028488)

[Figure 3-5 Sequence Diagram for Edit Profile 40](#_Toc48028489)

[Figure3-6 Sequence diagram for uppload assignment 41](#_Toc48028490)

[Figure3-7 Activity diagram for admin 43](#_Toc48028492)

[Figure3-8 Activity diagram for student 44](#_Toc48028493)

[Figure3-9 Activity diagram for teacher 45](#_Toc48028494)

[Figure3-10 Class diagram 47](#_Toc48028495)

[Figure3-11 Deployment diagram 48](#_Toc48028496)

[Figure4-1 Student login page 50](#_Toc48028497)

[Figure4-2 Student Home Page 51](#_Toc48028498)

[Figure4-3 Student Attendance Page 52](#_Toc48028499)

[Figure 4-4 Student Marks Page 53](#_Toc48028501)

[Figure4-5 Student Timetable Page 54](#_Toc48028502)

[Figure4-6 Class Assignment Page 54](#_Toc48028502)

[Figure4-7 Edit profile 54](#_Toc48028502)

[Figure4-8 Change password Page 54](#_Toc48028502)

[Figure4-9 Teacher login Page 58](#_Toc48028503)

[Figure4-10 Teacher homepage 58](#_Toc48028503)

[Figure4-11 Entering Attendance 58](#_Toc48028503)

[Figure4-12 Editing Attendance 58](#_Toc48028503)

[Figure4-13 Attendance of student in class 58](#_Toc48028503)

[Figure4-14 Entering Marks 58](#_Toc48028503)

Figure4-15 Editing Marks 58

Figure4-16 Marks of all students in class 58

Figure4-17 Teacher Timetable 58

Figure4-18 Grade and attendance for a class of students 58

[Figure4-19 Upload assignmnet 54](#_Toc48028502)

[Figure4-20 Class notice page 54](#_Toc48028502)

[Figure4-21 Update Teacher profile page 54](#_Toc48028502)

[Figure4-22 Submitted Assignment page 54](#_Toc48028502)

[Figure4-23 Change password page 54](#_Toc48028502)

[Figure4-24 Admin Login page 54](#_Toc48028502)

Figure4-25 Admin Homepage 58

Figure4-26 Admin add user page 58

[Figure4-27 Admin change password page 54](#_Toc48028502)

[Figure4-28 Design of database 54](#_Toc48028502)

# List of Table

[Table1-1Risk table 6](#_Toc83694951)

[Table1-2 Constraints Table 6](#_Toc83694952)

[Table1-3 Estimated Project Cost 10](#_Toc83694953)

[Table1-4 Work plan of the project 10](#_Toc83694954)

[Table3-1 Use case discription for login Project Cost 22](#_Toc83694953)

[Table3-2 Use case discription for upload assignment 22](#_Toc83694953)

[Table3-3 Use case discription for edit profile 23](#_Toc83694953)

[Table3-4 Use case discription for enter attendance 24](#_Toc83694953)

[Table3-5 Use case discription for view time table 24](#_Toc83694953)

[Table3-6 Use case discription for see students list 25](#_Toc83694953)

[Table3-7 Use case discription for write notice 25](#_Toc83694953)

[Table3-8 Use case discription for view time table 26](#_Toc83694953)

[Table3-9 Use case discription for view assignment 26](#_Toc83694953)

[Table3-10 Use case discription for take attendance 27](#_Toc83694953)

[Table3-11 Use case discription for create user 27](#_Toc83694953)

[Table4-1 User activity discription 64](#_Toc83694953)

**CHAPTER ONE**

**1. Introduction**

## 1.1. Background Information

“Defense Engineering College established in 1997, by Ministry of National Defense to produce highly professional and technically efficient military. The driving force for establishing DEC, among other things, was that Minster of National Defense did not have sufficient technical military personnel and that the higher institutions did not have programs compatible with the demands of Minster of National Defense. On its effort to achieve academic excellence, the college community goes through many days to day functions. In each department there are many students, teachers, assistants, staff members and leaders which are actively participating in the education tasks.”

A digital class management system is a web application to facilitate access to learning content and administration. digital class Management System consists of tasks such as registering students in class, assigning students and teacher to course, attendance record keeping to control absentees, uploading assignments, attaching modules, preparing timetable, exchanging information(class notice) between student and teachers, show marks of assignment, Quizzes and finals examination for students Online, and students give feedback to their respective teachers. The system connects daily activity in the class ranging from attendance management to communication means between teacher and student. This reduces data error ensures the information’s are always up-to-date throughout the class this insure that student s0pend less time to get any materials and also teachers spend less time to manage class

## 1.2. Statement of the problem

The problems regarding the current system in the college were analysed and noted. This project aims to solve some of the problems that faced the college due to traditional paper based system.

Such as:-

* + High cost
  + Time consuming
  + Damage class reputation
  + Storage problems
  + Slightest work efficiency

## 1.3. Objective of the project

## 1.3.1. General Objective

Using the various resources and tools we gathered along the way, we implemented the Digital Class system using some features that solve the current problems in the college such as a provision to edit the attendance and marks before locking it at the end. Thus, the results can be integrated in this digital system to improve its working performance and solve some of the existing problems mentioned in Statement of problem section

## 1.3.2. Specific Objective

* In order to minimize the cost of teaching material such as chalk, markers, projector, paper, and other related equipment
* To overcome wastage of time to see what is new on the notice board by going physically
* To overcome teacher’s time for the attendance paper to be filled by his/her students at the end of class session
* In order to adapt the digital world teaching mechanisms
* In order to remove any biased actions between teacher and student by favoring individuals to get extra class’s resource than the rest of the student
* To overcome any physical resources used for like exam paper, attendance sheet and other useful papers to examine the student at the end of the semester

## 1.4. Significance of the project

The development of this project is significant. Because the project allows teachers to save their golden time, the college resource for example chalk, paper, pen and etc.. student to see their result of exams , quiz, assignment , The system will help students to be connected to technology .

Other significance can be:

* The project overcomes all the manual approach of maintaining class information on paper by automating.
* As the system is fully automated, it does not require human efforts in calculating and maintaining student’s mark details.
* Also it maintains records of all the students along with their class information so there would be systematic way of teaching-learning process.
* Saves efforts and time and it is cost-effective
* Daily activities like attendance can be digitalized and automated.

## 1.5. Beneficiaries of the project

At the current stage of the project, those who benefited are DUCoE students and teachers

## 1.6. Application area

At the beginning, the main users of this system are DUCoE’s Students, and Teachers but the number of users will grow and encompass others who want to contribute for the community for future.

## 1.7. Methodology

## 1.7.1. Data collection methodology

This project will use the following methodologies for the following different tasks: -

Interview

On job observation

Review of relevant documents

Review of similar platforms

## 1.7.2. Development Environment and programming tool

The following development environments and programming tools are selected based on the following criteria:

Primarily team members experience with the tools and Software.

By the rate given to the software’s in different sites.

The features that the tools have to make the project more effective.

**Tools**

Operating system:

Windows 10

Database:

MySQL

Back end

Django

Python

Front end language:

HTML,

CSS,

JS

Bootstrap,

Server:

XAMPP

Text editor:

PyCharm

MS word 2016- for all word processing tasks and documentations.

Visio 2016 to draw diagrams.

## 1.7.3. Testing Procedure

**Software testing** is a process, to evaluate the functionality of a software application with an intent to find whether the developed software met the specified requirements or not and to identify the defects to ensure that the product is defect-free in order to produce a quality product.

We will conduct the following two tests:-

**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.

**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.

In testing the functionality of the website, the following will be tested:

**Link test**

Internal Links

External Links

Broken Links

**Forms test**

Field validation

Error message for wrong input

Optional and Mandatory fields

**Session Test**

Session handling

**Database test**

Testing will be done on the database integrity

## 1.8. Scope of the project

Digital Class Management is becoming a very essential component in education in this modern

day. With the help of College Automation System, we can gather all the useful information needed to the management in few clicks. The system now computerizes all the details that are maintained manually. Once the details are fed into the system or computer there is no need for various persons to deal with separate sections. At the current stage, the project is designed for student-teacher interaction only.

## 1.9. Risk, Assumption and Constraints

## 1.9.1. Risk

Risk Identification determines which risks might affect the project and documents their characteristics. A risk factor is a situation that may give rise to one or more project risks.

###### Table1-1­Risk table

|  |  |  |
| --- | --- | --- |
| Risk | Description | Mitigation |
| Risks | -To attain expected benefit from the project  -To estimate accurate project cost estimates  -To estimate accurate project duration | Understanding the users’ requirements very well will improve the benefits of the project,  To approach estimation for accuracies of the project cost and duration feasibility study is used. |

## 

## 1.9.2. Assumptions

Assumption is any project factor that is considered to be true, real, or certain without empirical proof or demonstration. Realistically speaking, it's impossible to plan a project without making a few assumptions.

## 1.9.3. Constraints

Project constraints are anything that restricts or dictates the actions of the project team. That can cover a lot of territory. The triple constraints—time, resources, and quality are the big hitters, and every project has one or two, if not all three, of the triple constraints as a project driver. Many projects in the Information Technology area, for instance, are driven by time.

###### Table1-2 Constraints Table

|  |  |
| --- | --- |
| Assumption and Constraints | Description |
| Constraint | . Time spent in normal class, quiz preparation, class assignments, final exam preparation and other factors.  . Cost that can be used to help the project when the team members are working outside the campus which requires several cost such as for internet cafe |
| Assumptions | There will not be any internet interruption for long period of time or days.  There will be continues electric power source. |

## 

**1.10. Project organization**

A project organization is a structure that facilitates the coordination and implementation of project activities. Its main reason is to create an environment that fosters interactions among the team members with a minimum amount of disruptions, overlaps and conflict.

Figure1-1 Project organization detail work structure with the team members.

**Figure1-2 Project organization and role of team members**.

## 1.11. Feasibility study

As the name implies, a feasibility analysis is used to determine the viability of an idea, such as ensuring a project is legally and technically feasible as well as economically justifiable. It tells us whether a project is worth the investment—in some cases, a project may not be doable. There can be many reasons for this, including requiring too many resources, which not only prevents those resources from performing other tasks but also may cost more than an organization would earn back by taking on a project that isn’t profitable.

The five types of feasibility are described as follows:

## 1.11.1. Economic feasibility

Economical Feasibility Development of this application is highly economically feasible .The college needed not spend much money for the development of the system already available. The only thing is to be done is making an environment for the development with an effective supervision. If we are doing so, we can attain the maximum usability of the corresponding resources. Even after the development, the college will not be in condition to invest more in the organization .Therefore, the system is economically feasible.

**Benefits**: Benefits are further divided into tangible and intangible benefits (Hoffer, 2000).

**Tangible benefits**: are those that can be measured in financial terms. Such as:

* Increased productivity
* Process improvements
* Very flexible and more convenient

### Intangible benefits: which cannot be quantified directly in economic terms, but still have a very significant business impact. Such as:

* Organizational strategy support
* Enhanced user experience
* Increased user satisfaction

Costs: The team may further classify these costs as tangible and intangible costs.

**Tangible costs**: are costs associated with an information system that can be measured in Birr

and with certainty such as:

Hardware costs

Labor costs

Operational costs

**Intangible costs**: are costs associated with an information system that can’t be easily measured in Birr and with certainty such as:

Loss of teachers’ moral to use the system

Loss of students’ moral to use the system

Loss of Administrator’s moral to use the system

## 

## 1.11.2. Technical feasibility

We can strongly says that it is technically feasible, since there will not be much difficulty in getting required resources for the development and maintaining the system as well. All the resources needed for the development of the software as well as the maintenance of the same is available in the organization here we are utilizing the resources which are available already.

## 1.11.3. Operational feasibility

In order users like students, teachers, and administrators use the full features of the system effectively; they need training on how to use the system. More than the training they get it is known that using the system and exploring it repeatedly will help them to know it fast. Having this in mind we believe that the system is operationally feasible.

## 

## 1.11.4*.* Schedule feasibility

According to our time plan, all of the objective of the project will be done in line with time schedule. Even though it is difficult to forecast the things that are going to happen in the feature, based on the time taken to accomplish every module of the project we believe that it is schedule feasible.

## 

## 1.11.5. Legal and Contractual feasibility

The agreement that we have regarding this system is with all departments in Defense University, College of Engineering. Because of this the system is free from any legal contractual risks.

## 1.12. Estimated Project Cost and time table

## 1.12.1. Estimated Project Cost

###### Table1-3 Estimated Project Cost

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Material | | Minimum Requirement | Quantity | Estimated price (ETB) |
| 1 | Computer | | -500 GB Hard Disk  -4GB RAM  -Core i3 processor (x64 based)  -2 GB Nvidia Graphics Card | 1 | 20,000.00 |
| 2 | Paper | | A4 size | 5000 | 150.00 |
| 3 | Typing Cost | | | 200.00 | |
| 4 | | Printing cost | | 300.00 | |
| 5 | | Binding cost | | 200.00 | |
| 6 | | Proposal | | 500.00 | |
| 7 | | Data collection | | 500.00 | |
| 8 | | System analysis and design | | 500.00 | |
| 9 | | Implementation | | 1000.00 | |
|  | | **TOTAL** | | **23,700.00** | |

## Table1-4 Work Plan of the Project

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **NO** | **Activity** | **Duration**  **Of**  **Week** | **Month1** | | | | **Month2** | | | | **Month3** | | | |
| **Week** | | | | **Week** | | | | **Week** | | | |
| **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| 1 | Title Selection | 2 |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Proposal Preparation | 1 |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Data Analysis | 4 |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Design | 8 |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Implementation | 6 |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Documentation | 9 |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Testing | 5 |  |  |  |  |  |  |  |  |  |  |  |  |

**CHAPTER TWO**

**2. Literature Review**

**2.1. Introduction**

This chapter review different journals, papers, literature's written about effectiveness of using digital class system and as Development of systems needs a detailed and deep analysis and study of other systems which are similar to the platform that is going to be developed.

### 2.2. Effectiveness of using digital class management system

In the current world, technology has changed the ways in which people learn. The main question does not lie upon whether a person should get classroom learning or be offered with classes online, by what matters is the manner in which it is implemented. Digital Classroom Management refers to the wide variety of skills and techniques that teachers use to keep students organized, orderly, focused, attentive, on task, and academically productive during a class. When classroom-management strategies are executed effectively, teachers minimize the behaviors that impede learning for both individual students and groups of students, while maximizing the behaviors that facilitate or enhance learning. Generally speaking, effective teachers tend to display strong classroom-management skills

While a limited or more traditional interpretation of effective digital classroom management system may focus largely on “compliance”—rules and strategies that teachers may use to make sure students are sitting in their seats, following directions, listening attentively, etc.—a more encompassing or updated view of classroom management extends to everything that teachers may do to facilitate or improve student learning, which would include such factors as *behavior* (a positive attitude, happy facial expressions, encouraging statements, the respectful and fair treatment of students, etc.), *environment* (for example, a welcoming, well-lit classroom filled with intellectually stimulating learning materials that’s organized to support specific learning activities), *expectations* (the quality of work that teachers expect students to produce, the ways that teachers expect students to behave toward other students, the agreements that teachers make with students), *materials* (the types of texts, equipment, and other learning resources that teachers use), or *activities* (the kinds of learning experiences that teachers design to engage student interests, passions, and intellectual curiosity).

### 2.2.1. Literature Review of other system

### 2.2.2*.* PyCharm

What is pycharm?

PyCharm is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) used in [computer programming](https://en.wikipedia.org/wiki/Computer_programming), specifically for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) language. It is developed by the [Czech](https://en.wikipedia.org/wiki/Czech_Republic) company [JetBrains](https://en.wikipedia.org/wiki/JetBrains) (formerly known as IntelliJ). It provides code analysis, a graphical debugger, an integrated unit tester, integration with [version control systems](https://en.wikipedia.org/wiki/Revision_control) (VCSes), and supports web development with [Django](https://en.wikipedia.org/wiki/Django_(web_framework)) as well as [data science](https://en.wikipedia.org/wiki/Data_science) with [Anaconda](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)).

PyCharm is [cross-platform](https://en.wikipedia.org/wiki/Cross-platform), with [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux) versions. The Community Edition is released under the [Apache License](https://en.wikipedia.org/wiki/Apache_License), and there is also Professional Edition with extra features – released under a [proprietary license](https://en.wikipedia.org/wiki/Proprietary_software).

**Features of PyCharm**

* Coding assistance and [analysis](https://en.wikipedia.org/wiki/Code_analysis), with [code completion](https://en.wikipedia.org/wiki/Autocomplete), syntax and error highlighting, [linter integration](https://en.wikipedia.org/wiki/Lint_(software)), and quick fixes
* Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages
* Python [refactoring](https://en.wikipedia.org/wiki/Refactoring): includes rename, extract method, introduce variable, introduce constant, pull up, push down and others
* Support for web frameworks: [Django](https://en.wikipedia.org/wiki/Django_(web_framework)), [web2py](https://en.wikipedia.org/wiki/Web2py) and [Flask](https://en.wikipedia.org/wiki/Flask_(web_framework)) [professional edition only][[8]](https://en.wikipedia.org/wiki/PyCharm#cite_note-8)
* Integrated Python [debugger](https://en.wikipedia.org/wiki/Debugger)
* Integrated [unit testing](https://en.wikipedia.org/wiki/Unit_testing), with line-by-line [code coverage](https://en.wikipedia.org/wiki/Code_coverage)
* [Google App Engine](https://en.wikipedia.org/wiki/Google_App_Engine) Python development [professional edition only]
* Version control integration: unified user interface for [Mercurial](https://en.wikipedia.org/wiki/Mercurial), [Git](https://en.wikipedia.org/wiki/Git_(software)), [Subversion](https://en.wikipedia.org/wiki/Apache_Subversion), [Perforce](https://en.wikipedia.org/wiki/Perforce) and [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) with change lists and merge
* Support for scientific tools like matplotlib, numpy and scipy [professional edition only]

It competes mainly with a number of other Python-oriented IDEs, including [Eclipse](https://en.wikipedia.org/wiki/Eclipse_(software))'s [PyDev](https://en.wikipedia.org/wiki/PyDev), and the more broadly focused [Komodo IDE](https://en.wikipedia.org/wiki/Komodo_IDE).

**Pros:**

* A plethora of productive shortcuts
* Ability to view the entire Python source code with a single click
* Availability of an array of plugins
* Easy-to-use
* Excellent community support
* Facilitates faster code development
* More powerful, commercial version available
* Straightforward installation process

**Cons:**

* Costly paid version
* May pose issues when trying to fixing tools like venv
* Not suitable for Python beginners
* Resource-intensive application, i.e., requires plenty of memory and storage space

### 2.2.3. Django

Django is a [Python](https://en.wikipedia.org/wiki/Python_(programming_language))-based [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source_software) [web framework](https://en.wikipedia.org/wiki/Web_framework) that follows the model–template–views (MTV) [architectural pattern](https://en.wikipedia.org/wiki/Architectural_pattern_(computer_science)). It is maintained by the [Django Software Foundation](https://en.wikipedia.org/wiki/Django_Software_Foundation) (DSF), an American independent organization established as a [501(c)(3)](https://en.wikipedia.org/wiki/501(c)(3)) non-profit.

Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes [reusability](https://en.wikipedia.org/wiki/Reusability) and "pluggability" of components, less code, low coupling, rapid development, and the principle of [don't repeat yourself](https://en.wikipedia.org/wiki/Don%27t_repeat_yourself). Python is used throughout, even for settings, files, and data models. Django also provides an optional administrative [create, read, update and delete](https://en.wikipedia.org/wiki/Create,_read,_update_and_delete) interface that is generated dynamically through [introspection](https://en.wikipedia.org/wiki/Type_introspection) and configured via admin models.

Some well-known sites that use Django include [Instagram](https://en.wikipedia.org/wiki/Instagram), [Mozilla](https://en.wikipedia.org/wiki/Mozilla_Foundation), [Disqus](https://en.wikipedia.org/wiki/Disqus), [Bitbucket](https://en.wikipedia.org/wiki/Bitbucket), [Nextdoor](https://en.wikipedia.org/wiki/Nextdoor) and [Clubhouse](https://en.wikipedia.org/wiki/Clubhouse_(app)).

**Pros:**

### ‘Batteries included’ philosophy

Django positions itself as a ‘batteries-included’ framework. This ‘batteries-included’ philosophy means that Django comes with a wide range of features and functionalities. This framework comes with a lot of stuff out of the box. So, instead of writing your own code, you just need to import the packages for adding functionalities.

This not only saves a lot of time for the developers but also allows them to focus on adding advanced functionalities. Moreover, since it is open-source, one can use the packages made by world-class community members. By simply importing the packages, developers can start building applications in no time.

### Intelligent application development capabilities

Django is a core solution in various internet of things (IoT) applications. It is preferred by developers working on Machine Learning (ML), Artificial Intelligence (AI), and IoT algorithms. Its computational capabilities make it an ideal platform for machine learning apps. It is also compatible with some powerful machine learning libraries like PyTorch, NumPy, etc.

Since Django provides regression and prediction capabilities, it’s ideal for developing intelligent applications be that on IoT, AI, or ML.

### Flexibility

Since Django is Python-based which a versatile programming language is, it provides more flexibility and dynamism. The framework can be configured quickly to match specific needs.

### Security

Made by the world’s best web developers, this framework leaves a very small possibility of security loopholes. It also includes prevention of common attacks like SQL injections and cross-site request forgery.

**Cons:**

### Lack of conventions

A big reason why many programmers dislike Django is that it lacks conventions. As opposed to other frameworks like Ruby on Rails, everything needs to be explicitly defined in Django. This leads to configuration boilerplate and, therefore, a slowed development process. It makes it difficult for developers who are used to working with frameworks with convention over configuration.

### Not suitable for smaller projects

The Django web framework comes with lots of code that takes the server’s processing and time. This can take its toll on low-end websites that run on very little bandwidth. That is why it is not suitable for projects and products with only a few features and requirements.

### Inability to simultaneously handle multiple requests

Unlike many modern web frameworks, Django is incapable of enabling individual processes to handle multiple requests simultaneously. This puts the pressure on developers to come up with ways to make individual processes handle multiple requests efficiently.

### Difficult to learn

One big disadvantage of Django is that it has a steep learning curve. It comes with a lot of features that cannot be easily understood by developers. This can be really tough for developers who are making a switch from other languages. Having a different syntax also makes it hard for developers to quickly get a hang of it. Python can thus be hard to master.

### 2.3. Proposed System

The proposed system has basically three types of user which have different privilege and role. Teacher, Student, and Admin. We have developed a system in which student can interact with their respective instructors digitally through our system. The system also provides a platform for any announcement that was found on the Notice Board during the manual approach that required to go directly to the board and attendance system. Other advantage of the proposed system includes:

* User friendly interface
* Fast access to database
* Less error
* More Storage Capacity
* Look and Feel Environment
* Smooth interaction

### 2.4. Summary

In general, we have reviewed some related published journals on the effectiveness of implementing digital class system over the traditional (manual) one. The studies also show that the digital class system has the potential to improve engagement, to provide depth class resources, saves teacher’s time, etc.

**CHAPTER THREE**

**3. Object Oriented Analysis and Design**

#### 3.1. A brief history

The object-oriented paradigm is based on the concept of "objects", which can contain data and code: data in the form of fields (often known as attributes or properties), and code, in the form of procedures (often known as methods).

#### 3.2. Object-oriented analysis

Object–Oriented Analysis (OOA) seeks to understand (analyze) a problem domain (the challenge we are trying to address) and identifies all objects and their interaction

#### 3.3. Object-oriented design

Object–Oriented Design (OOD) then develops (designs) the solution.

#### 3.4. System use case model

#### 3.4.1. What is a Use case?

A use case is a description of how a person who actually uses that process or system will accomplish a goal. It's typically associated with software systems, but can be used in reference to any process.

#### 3.4.2. What is a Use Case Diagram?

Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. Use-case diagrams illustrate and define the context and requirements of either an entire system or the important parts of the system.

#### 3.4.3. How do you write a Use Case?

Use case diagrams are designed with 4 major elements

Actors

System

Use Cases

Relationships between actors and use cases

After a deep study of the system we found different system users and categorized them in the

Following actor

1. Administrator
2. Student
3. Teacher

The following use case are identified in our system

UC\_01 Login

UC\_02 Upload Assignment

UC\_03 Edit profile

UC\_04 enter attendance

UC\_05 View Time Table

UC\_06 See Student List

UC\_07 Write Notice

UC\_08 View Assignment

UC\_09 Take Attendance

UC\_10 Create User



Figure3-1 Use Case Diagram for Admin



Figure3-2 Use Case Diagram for Teacher



Figure3-3 Use Case Diagram for Student

###### Table3-1 Use case description for Login

|  |  |
| --- | --- |
| Use case name | Login |
| Number | UC\_01 |
| Description | This use case is used by Admin, Techer and Student to log in to the system according to their privilege. |
| Actors | Admin, Teacher, or Student according to their privilege |
| Precondition | The admin must have admin privilege User Name and password, Teacher also must have Teacher level privilege User Name and password, the Student also must have Student level privilege User Name and password. |
| Flow events | 1: All the users must access the system by browser.  2: The system display home page.  3: The home page displays the login modal link.  4: The users Click on the login link.  5. The system displays the modal login form.  6: The System User Enters user Name and password.  7: Send in to the database by pressing login button.  8: The system authenticates the User Name and password.  9. The system redirects to authorized page according to the privilege specified in the database and display Dashboard based on their privilege.  End use case |
| Post condition | The system should display to Admin, student or Teacher page |
| Alternative flow of events | The Admin, Teacher or Student enters wrong user Name or password displays an error message.  8.1. Go back to step 5 |

###### Table3-2 Use case description for Upload Assignment

|  |  |
| --- | --- |
| Use case name | Upload Assignment |
| Number | UC\_02 |
| Description | Teacher Upload Assignment |
| Actors | Teacher |
| Precondition | The Teacher must enter User Name and password to access the system. |
| Normal flow of events | 1.Teacher selects dashboard link.  2. Selects the Upload Assignment link.  3. Teacher select Assignment  4. Then teacher choose a file to be uploaded  5. Click upload button.  6. The system displays user assignment is uploaded successfully message.  End of use case. |
| Post condition | The Assignment is uploaded and Student can download the Assignment |

###### Table 3-3 Use case description for Edit profile

|  |  |
| --- | --- |
| Use case name | Edit profile |
| Number | UC\_03 |
| Description | Teacher, and Student edit their own profile |
| Actors | Teacher or student |
| Precondition | Actor must enter User Name and password to access the system. |
| Normal flow of events. | 1. Actor select dashboard link.  2. Selects edit profile button  3. The system displays a form to edit profile and enter info to form.  4. Press Submit button and click on it.  5. The system displays profile edited successfully.  End of use case. |
| Post condition | The profile is edited and redirect to the dashboard. |

###### Table3-4 Use case description for Enter Attendance

|  |  |
| --- | --- |
| use case name | Enter Attendance |
| Number | UC\_03 |
| Description | Teacher feel Attendance |
| Actors | Teacher |
| Precondition | The teacher must enter User Name and password to access the system. |
| Normal flow of events. | 1. Teacher select dashboard link.  2. Teacher selects add Enter Attendance link.  2. The system displays a form to feel attendance and enter info to form.  4. Press Submit button and click on it.  5. The system displays attendance feel successfully.  End of use case |
| Post condition | The attendance is filled redirect to the dashboard |
| Alternative flow of events | If the content not found.  Steps 1-2 remains the same  2.1 System displays Attendance is not feel successfully.  3.2 System enables user to try again |

###### Table3-5 Use case description for View Time Table

|  |  |
| --- | --- |
| use case name | View Time Table |
| Number | UC\_05 |
| Description | Teacher and Student see Time Table |
| Actors | Student or Teacher |
| Precondition | The user must enter user name and password to access the system. |
| Normal flow of events. | 1. Actor select dashboard link.  2. Actor selects View Time Table link.  3. The system displays a Time Table for student and teacher.  4. End of use case |
| Post condition | The Time Table is visible for student and teachers |
| Alternative flow of events | If there is no time table.  Steps 1-2 remains the same  2.1 System displays there is no time table yet. |

###### Table3-6 Use case description for See Student List

|  |  |
| --- | --- |
| Use case name | See student list |
| Number | UC\_06 |
| Description | Teacher see student list |
| Actors | Teacher |
| Precondition | The teacher must enter User Name and password to access the system. |
| Normal flow of events. | 1. Actor selects see student list link. 2. The system displays student list End of use case |

###### Table3-7 Use case description for write notice

|  |  |
| --- | --- |
| use case name | Write notice |
| Number | UC\_08 |
| Description | Teacher write notice |
| Actors | Teacher |
| Precondition | The user must enter user name and password to access the system. |
| Normal flow of events. | 1. Actor select dashboard link.  2. Actor selects add notice.  3. Actor click on submit.  4. The system displays the notice is added  5. End of use case |
| Post condition | The Notice is visible for student and teachers |
| Alternative flow of events | If there is no added notice.  Steps 1-2 remains the same  2.1 System displays there is no added notice yet. |
|  |  |

###### Table3-8 Use case description for View Time Table

|  |  |
| --- | --- |
| use case name | View Time Table |
| Number | UC\_05 |
| Description | Teacher and Student see Time Table |
| Actors | Student or Teacher |
| Precondition | The user must enter user name and password to access the system. |
| Normal flow of events. | 1. Actor select dashboard link.  2. Actor selects View Time Table link.  3. The system displays a Time Table for student and teacher.  4. End of use case |
| Post condition | The Time Table is visible for student and teachers |
| Alternative flow of events | If there is no time table.  Steps 1-2 remains the same  2.1 System displays there is no time table yet. |

###### Table3-9 Use case description for View Assignment

|  |  |
| --- | --- |
| use case name | View Assignment |
| Number | UC\_09 |
| Description | Teacher and Student view assignment |
| Actors | Student and Teacher |
| Precondition | The user must enter user name and password to access the system. |
| Normal flow of events. | 1. Actor select dashboard link.  2. Actor selects View assignment link.  3. The system displays a assignment form for student and teacher.  4. End of use case |
| Post condition | The assignment is visible for student and teachers |
| Alternative flow of events | If there is no assignment.  Steps 1-2 remains the same  2.1 System displays there is no assignment. |

###### Table3-10 Use case description for Take Attendance.

|  |  |
| --- | --- |
| use case name | Take Attendance |
| Number | UC\_10 |
| Description | Teacher take attendance |
| Actors | Teacher |
| Precondition | The user must enter user name and password to access the system. |
| Normal flow of events. | 1. Actor select dashboard link.  2. Actor selects take attendance link.  3. The system displays absent present form for a teacher.  4. Teacher fill the form.  5. Actor click submit button.  6. End of use case |
| Post condition | The Attendance is visible for student and teacher |
| Alternative flow of events | If there is no time table.  Steps 1-2 remains the same  2.1 System displays there is no time table yet. |

###### Table3-11 Use case description for create user

|  |  |
| --- | --- |
| use case name | Create user |
| Number | UC\_11 |
| Description | Admin create user |
| Actors | Admin |
| Precondition | The user must enter user name and password to access the system. |
| Normal flow of events. | 1. Actor select dashboard link.  2. Actor selects add button.  3. Actor fill the form.  4. Actor click on save button  5. The system displays user is created successfully.  6. End of use case |
| Post condition | The created users can access the database |
| Alternative flow of events | If there is no created user.  Steps 1-2 remains the same  2.1 System displays there is no created users yet. |

#### 

#### 3.4.4. Sequence Diagram

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.



Figure3-4 Sequence diagram for login



Figure3-4 Sequence Diagram for edit profile

 Figure3-6 Sequence diagram for Upload Assignment

**3.4.5. Activity diagram**

Activity Diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

Figure3-8Activity diagram for admin



Figure3-9 Activity diagram for Student



Figure3-10 Activity diagram for Teacher

#### 3.4.6. Class diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system.

#### 3.4.7. Purpose of Class Diagrams

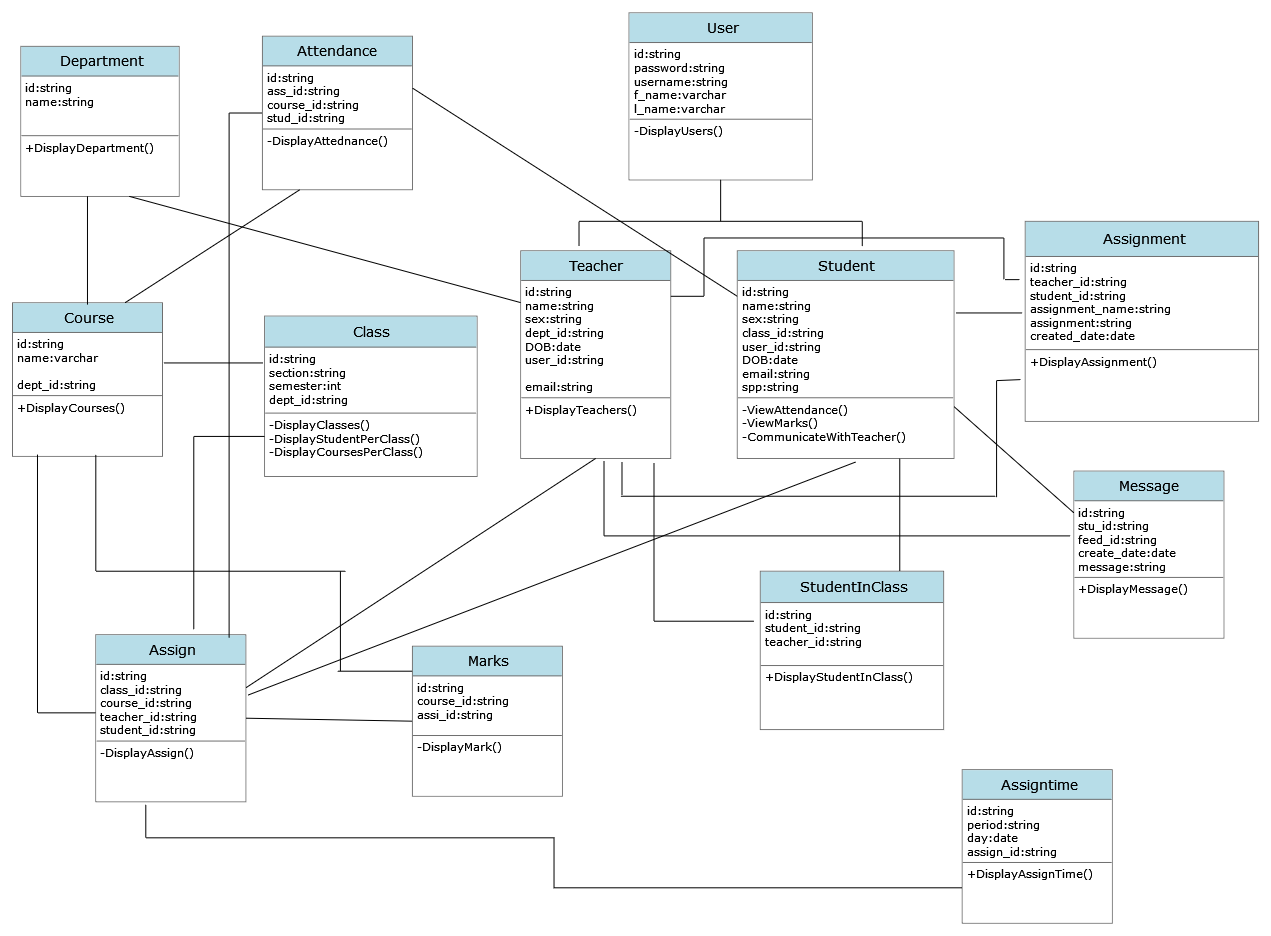
Class diagrams are the most important kind of UML diagram and are vitally important in software development. Class diagrams are the best way to illustrate a system’s structure in a detailed way, showing its attributes, operations as well as its inter-relationships.

Figure3-11 Class diagram of the system

#### 3.4.8. Deployment

Deployment Diagram is a type of diagram that specifies the physical hardware on which the software system will execute. It also determines how the software is deployed on the underlying hardware. It maps software pieces of a system to the device that are going to execute it.

#### 3.4.8.1. Purpose of Deployment Diagrams

Deployment diagrams are used with the sole purpose of describing how software is deployed into the hardware system. It visualizes how software interacts with the hardware to execute the complete functionality. It is used to describe software to hardware interaction and vice versa.

Figure3-12 Deployment diagram

##### 

##### CHAPTER FOUR

##### 4. System Description

##### 4.1. Introductions

The new proposed system named ‘DUCoE Digital Class Management System’ is a computer-based web application which digitizes the teaching-learning system.

The system provides:

* Each teacher will be able to enter attendance and marks for their respective students.
* Each student will be able to view the attendance status for their respective courses.
* The teachers will be able to apply for various types of leave directly through the system.
* The students will be able to Communicate and provide feedback to their teachers.
* The students will have access to a forum page where they are communicate will each other.
* The administrator will be able to view and update information such as departments, classes, teachers, students, courses.

This section of the document is aimed to describe the proposed system and its major functions in detail. The section also provides major navigation pages with their corresponding outputs and layouts in preferred format and describes the system architecture including the database, the server and the front-end framework.

##### 4.1.1. Essential Navigations

‘DUCoE Digital Class Management System’ is a computer-based system, and has multiple pages navigated by the user. In general, there are four main pages, namely

* Student page: where they can view the Attendance status of the courses to which they are enrolled, view the Marks of the courses to which they are enrolled and so on.

Students are end users of this system. The attendance, 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.

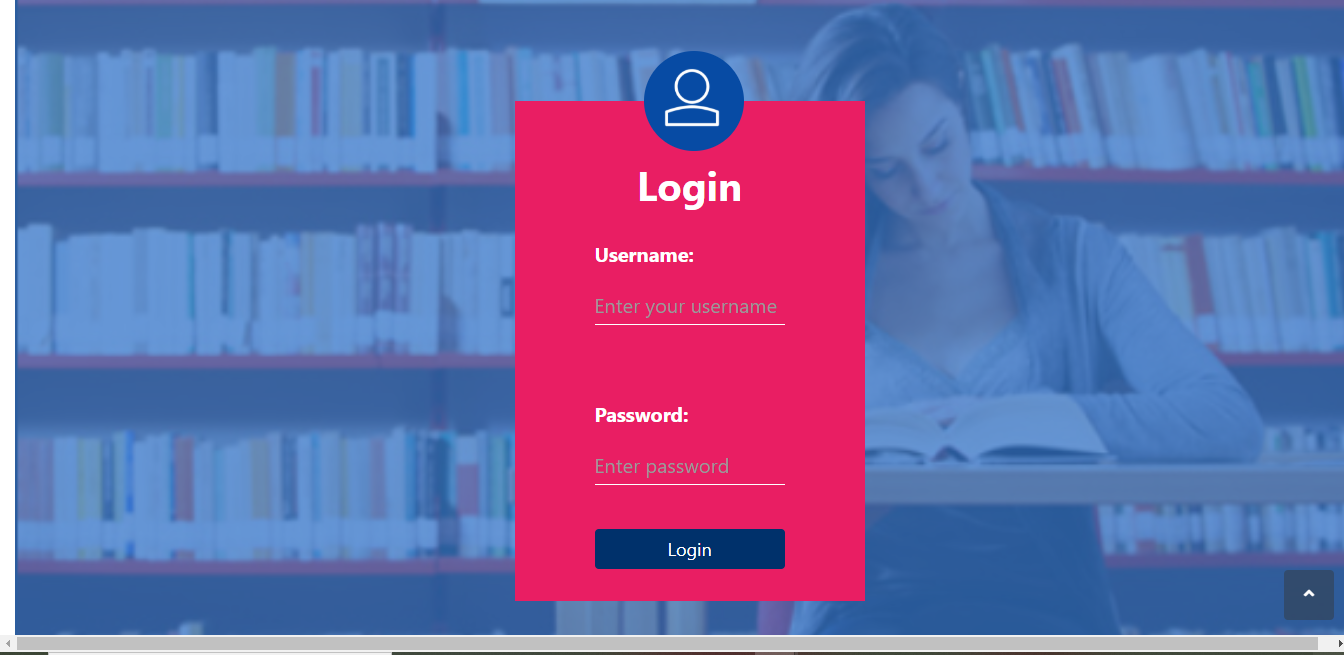
* Teacher page: where will have access to information regarding the courses and classes they are assigned to, will also have an access to information of students who belong to the same class as the teacher. Teachers are the key stakeholders of this system. Because they are the one who manage, edit, update the contents of the database of students such as attendance, marks, 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 help them reduce their overhead and make their tasks easier and simple.
* Administrator page: where they can Add and Update students, teachers and courses, assign teachers and students to courses. They will have the privilege to modify the database i.e., to add/remove students/teachers/, update information regarding each of these. It is their responsibility to maintain the database. So the Administrator play a major role in the system.

##### 4.2. Users page

##### 4.2.1 Student page

**4.2.1.1 Login**

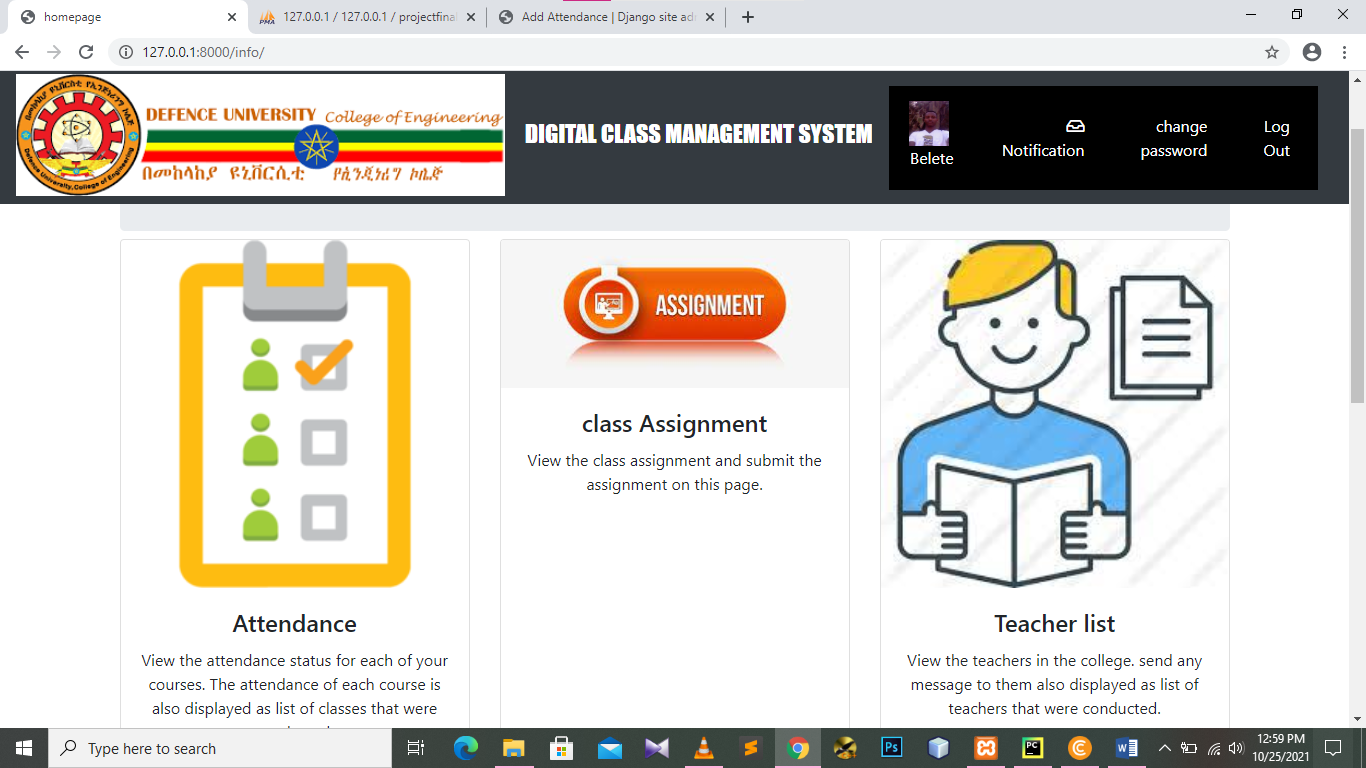
Each student in the college is assigned a unique username and password by the administrator. The user- name is the same as their USN and so is the password. They may change it later according to their wish.



**Figure4.1 Student Login Page**

**4.2.1.2 Homepage**

After successful login, the student is presented a homepage with their main sections, attendance, marks timetable, class assignment, class notice, notification, profile, change password, teacher list. 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 10 for 5 assessments, and semester end examination for 50 marks. in the timetable provides the classes assigned to that student and day and time of each in a tabular form. in the notification student see any class notice.in the class assignment section, the student can see assignments uploaded and submit those assignment to their teacher. In the profile section each student can edit their assignment.

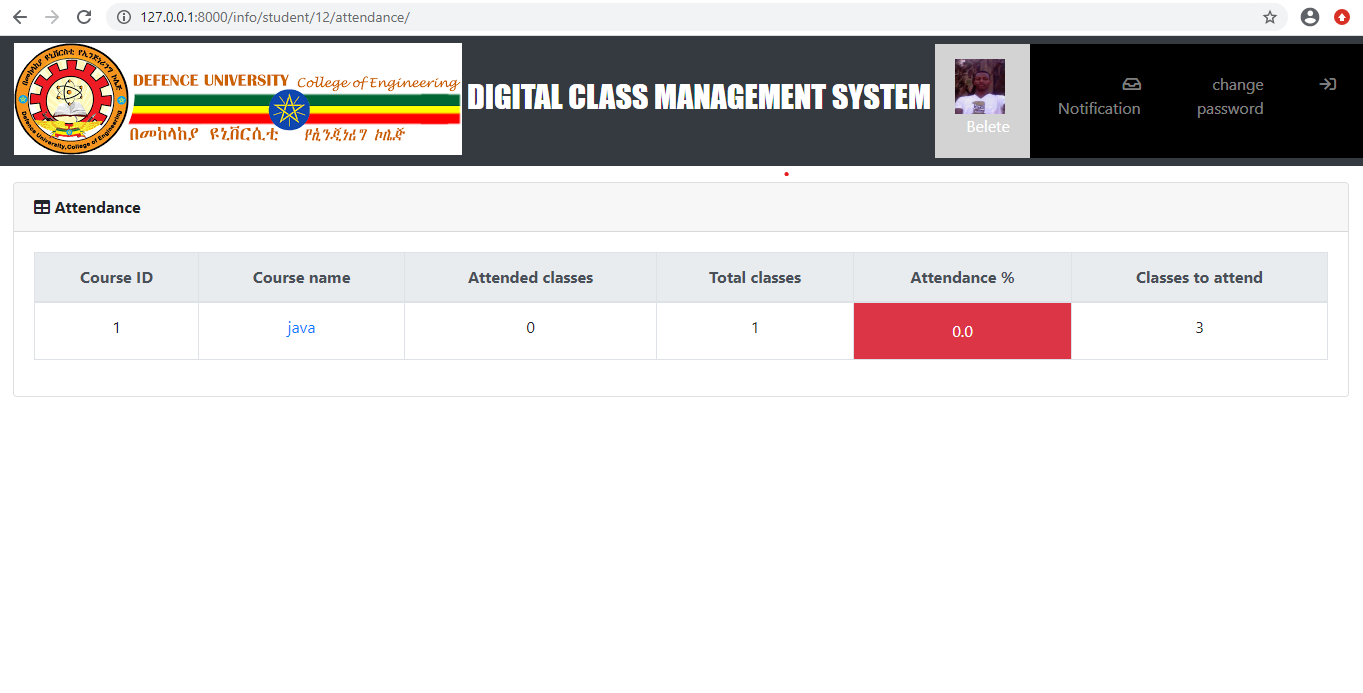
****

**Figure4.2 Student Home Page**

**4.2.1.3 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.

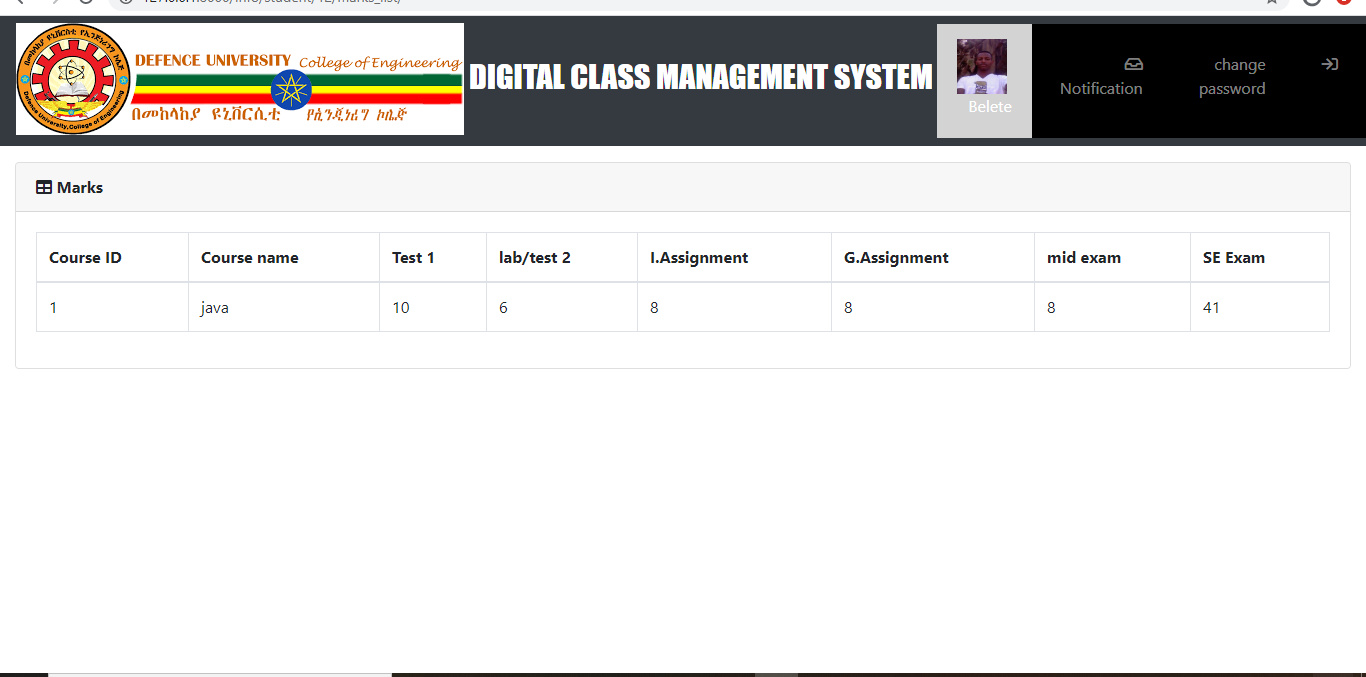
##### 



**Figure4.3 Student Attendance Page**

**4.2.1.4 Marks**

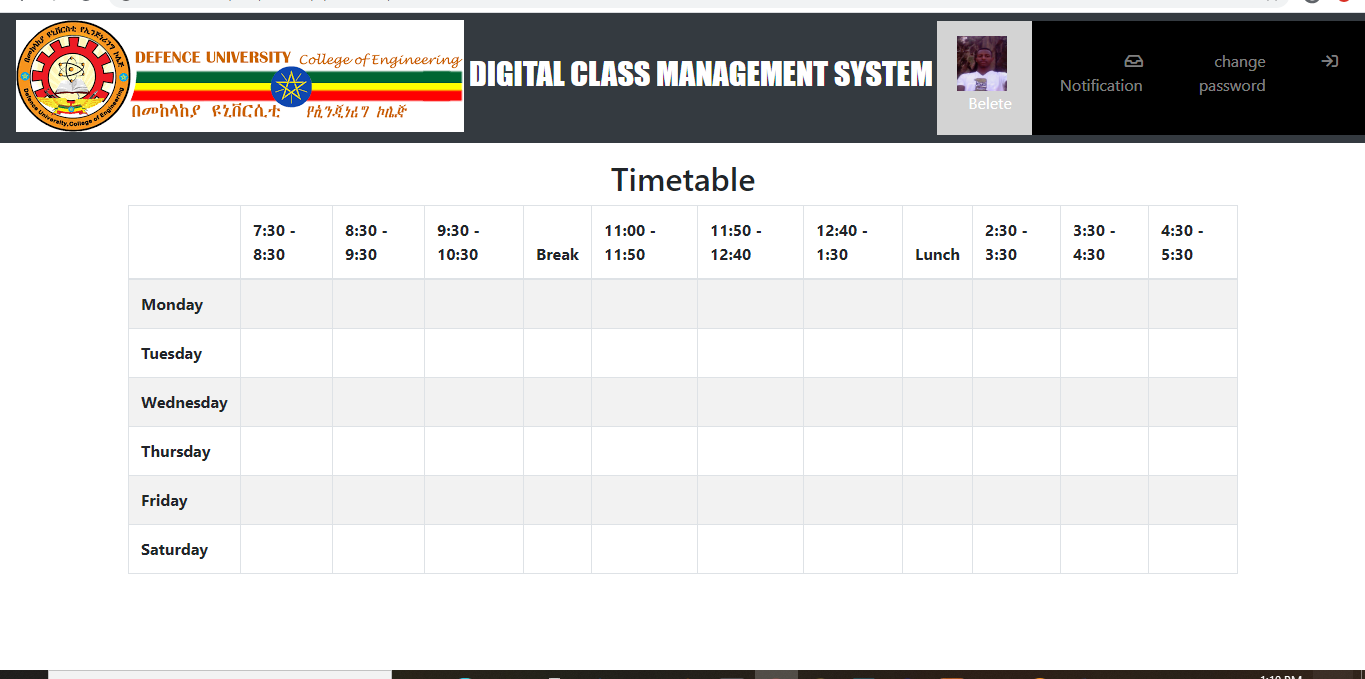
The Marks page is a table with an entry for each of their courses. The course id and name are speciﬁed along the marks obtained in each of the tests and exams. The tests include 5 assessments with marks obtained out of a total of 10 for lab, assignment, and quiz etc., one semester end exam with marks out of 50.



**Figure4.4 Student Marks Page**

##### 4.2.1.5 Timetable

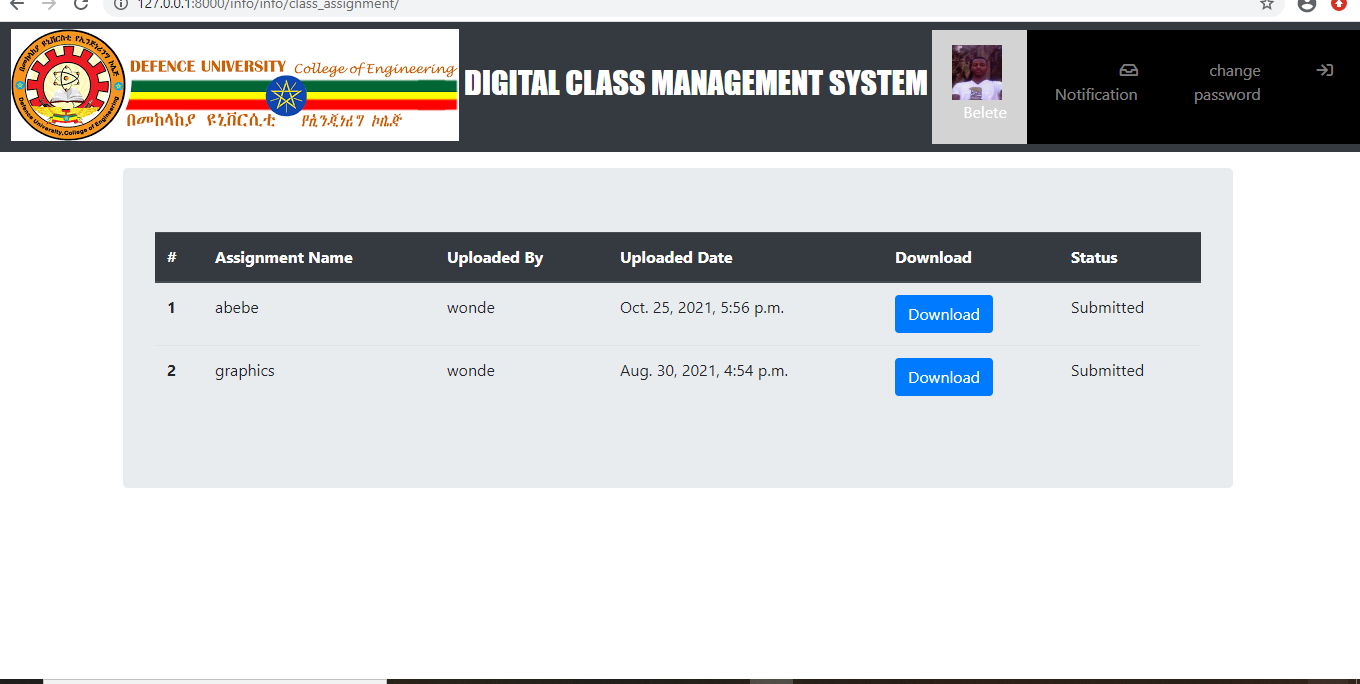
This page is a table which lists the day and timings of each of the classes assigned to the student. The row headers are the days of the week and the column headers are the time slots. So, for each day, it speciﬁes the classes in the time slots. The timetable is generated automatically from the assign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

****

**Figure4.5 Student Timetable**

**4.2.1.6 Class assignment**

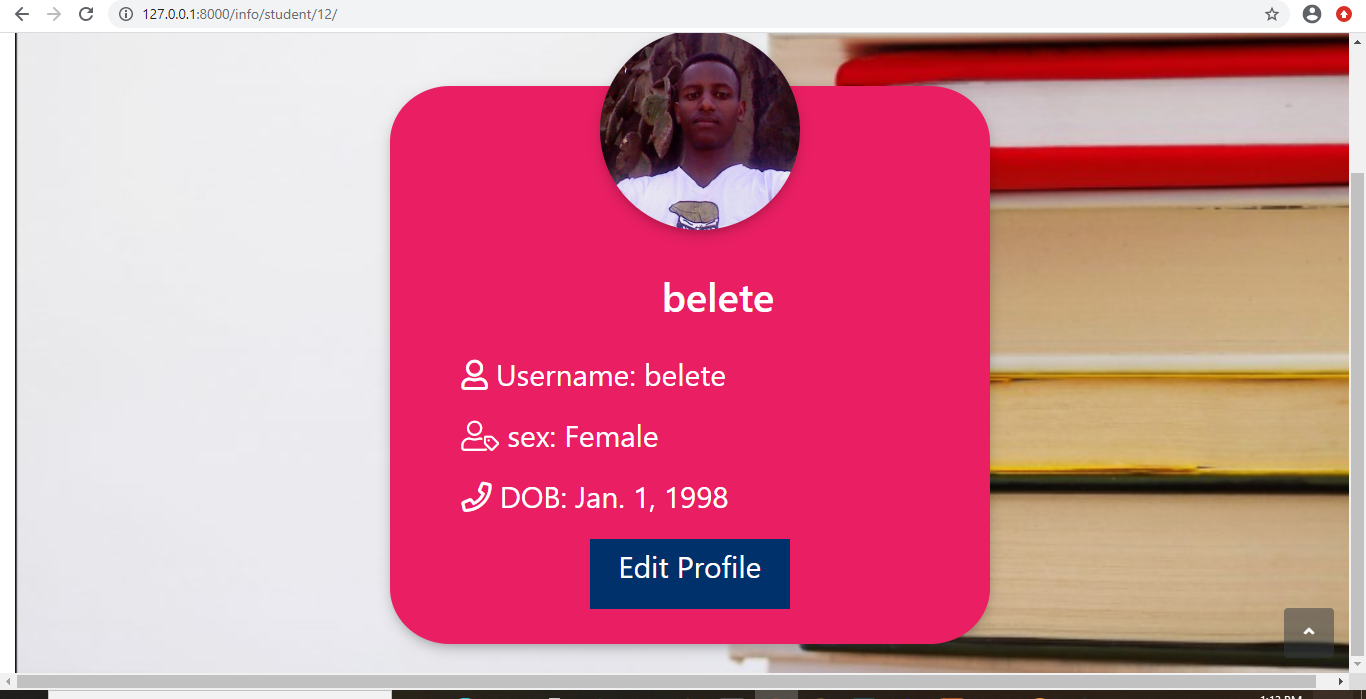
On this page, there is assignment name, teacher name who upload assignment, uploaded date download button and assignment content for each course. If there is no assignment uploaded yet it displays not assignment uploaded yet. If you click on submit button the page redirects to computer desktop to take the file content. Once assignment is submitted to teacher student can’t update it.

****

**Figure4.6 Class assignment**

**4.2.1.7 Update profile**

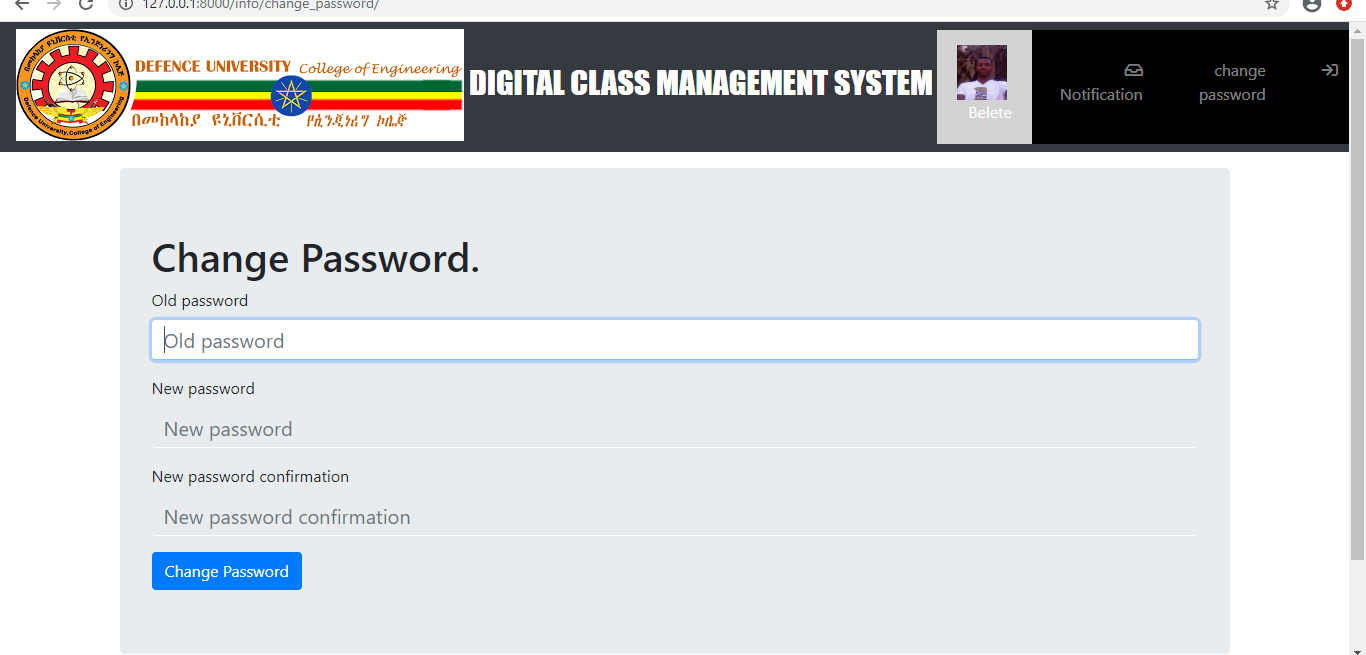
Update User Profile. Your user profile contains the personal information that is necessary for you using this Profile page you can update your profile photo, you can update your gender and other personal information.

****

**Figure4.7 Edit profile**

**4.2.1.8 Change password**

This page helps user to create their own password. Before you change password you will need to have your current password**.**

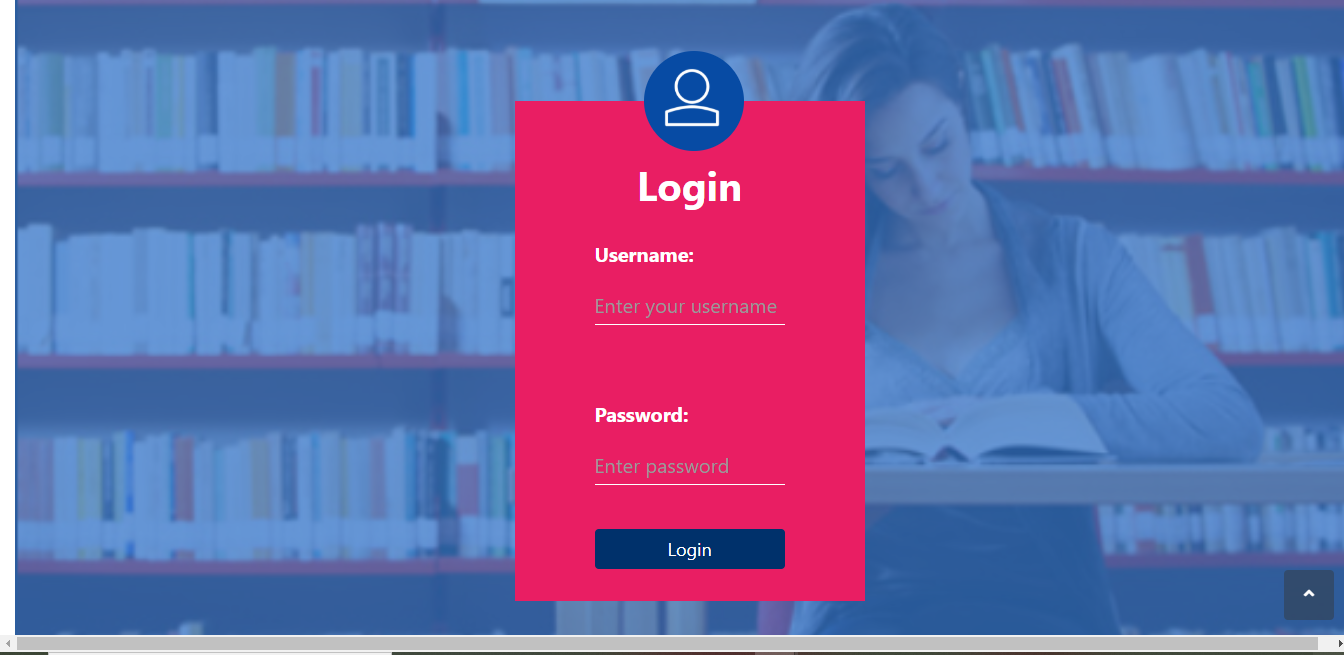


**Figure4.8 Change password page**

##### 4.2.2 Teacher page

**4.2.2.1 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.



**Figure4.9 Teacher login page**

**4.2.2.2 Homepage**

After successful login, the teacher is presented a homepage with their main sections, attendance, marks, timetable, edit profile, upload assignment, submitted assignment, message, change password 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 5 assessments 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.



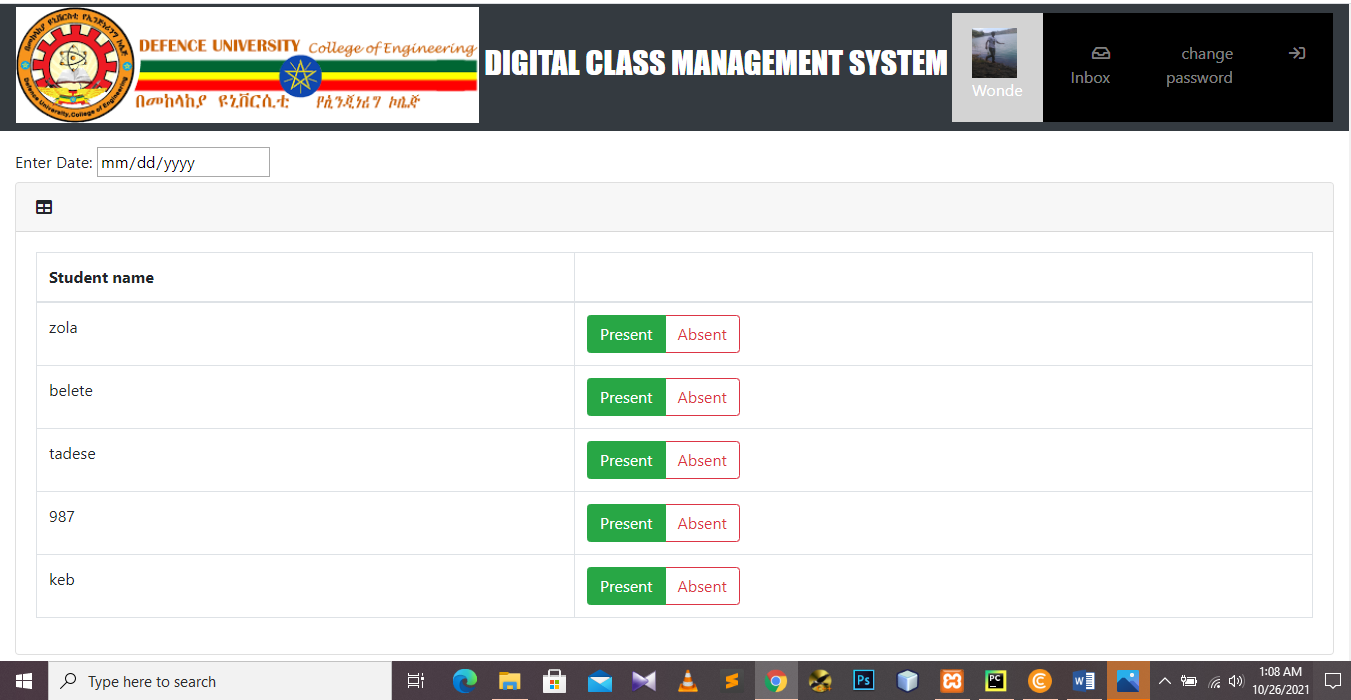
**Figure4.10 Teacher Homepage**

**4.2.2.3 Attendance**

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

**Enter Attendance**

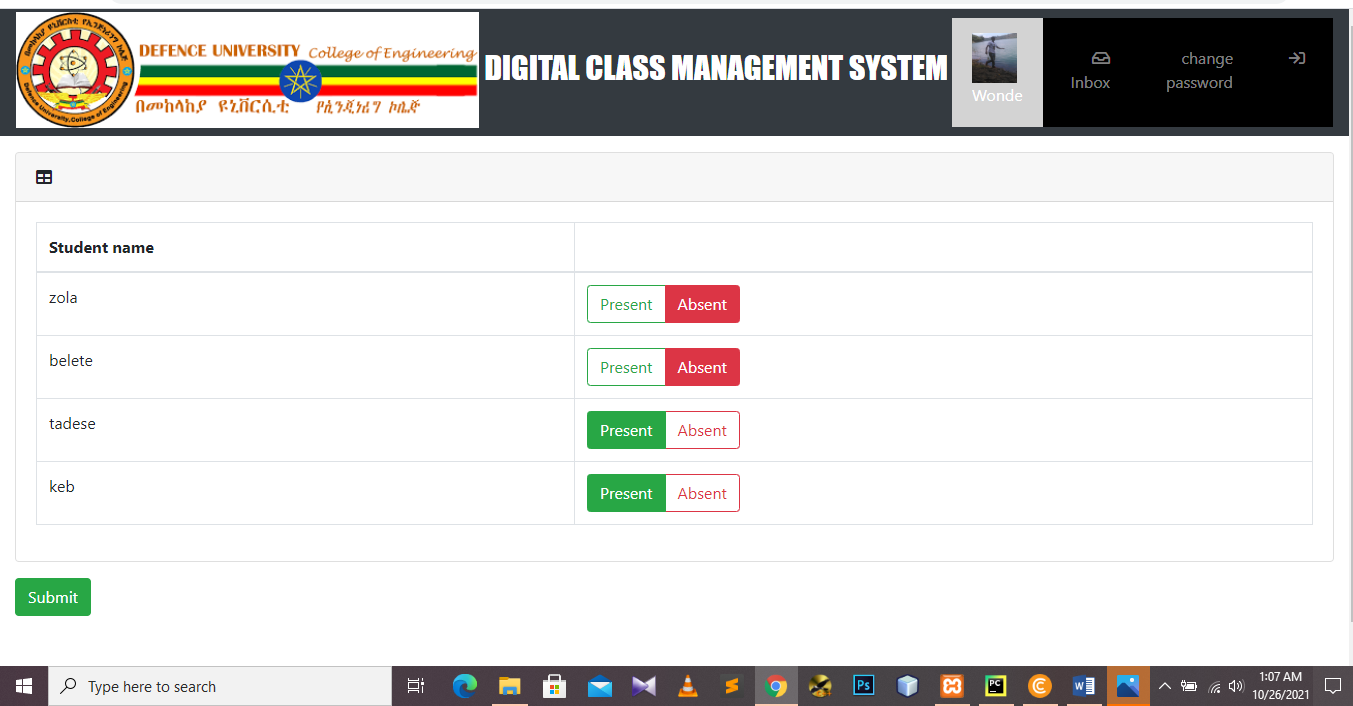
On this page, the classes scheduled or conducted is listed in the form of a list. Initially, all the scheduled classes will be listed from the start of the semester to the current date. Thus, if there is class scheduled for today, it will automatically appear on top of the list. If the attendance of 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 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

****

**Figure4.11 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.

****

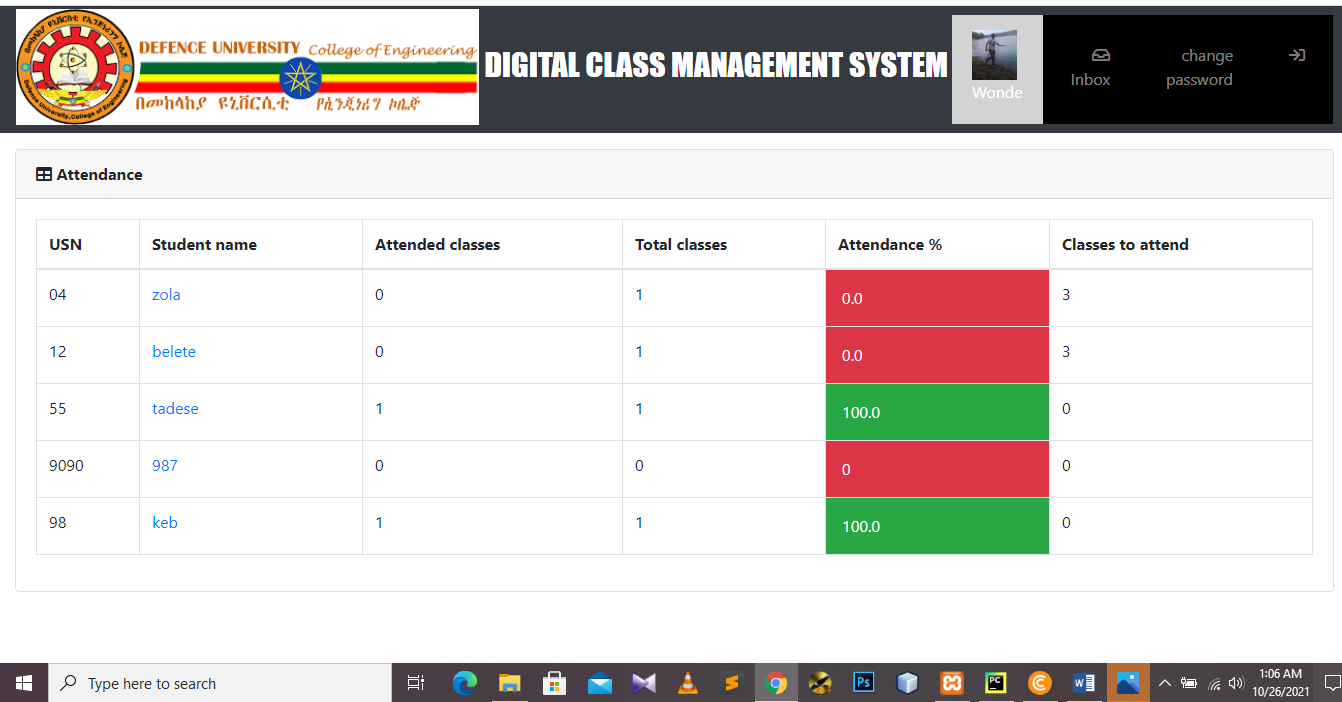
**Figure4.12 Editing Attendance**

**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.

**Student Attendance Details**

The teacher can view the attendance detail of all their assigned students individually. That is, for all the conducted classes, it will display whether that student was present or absent. The teacher can also edit the attendance of each student individually by changing the attendance status for each conducted class.

****

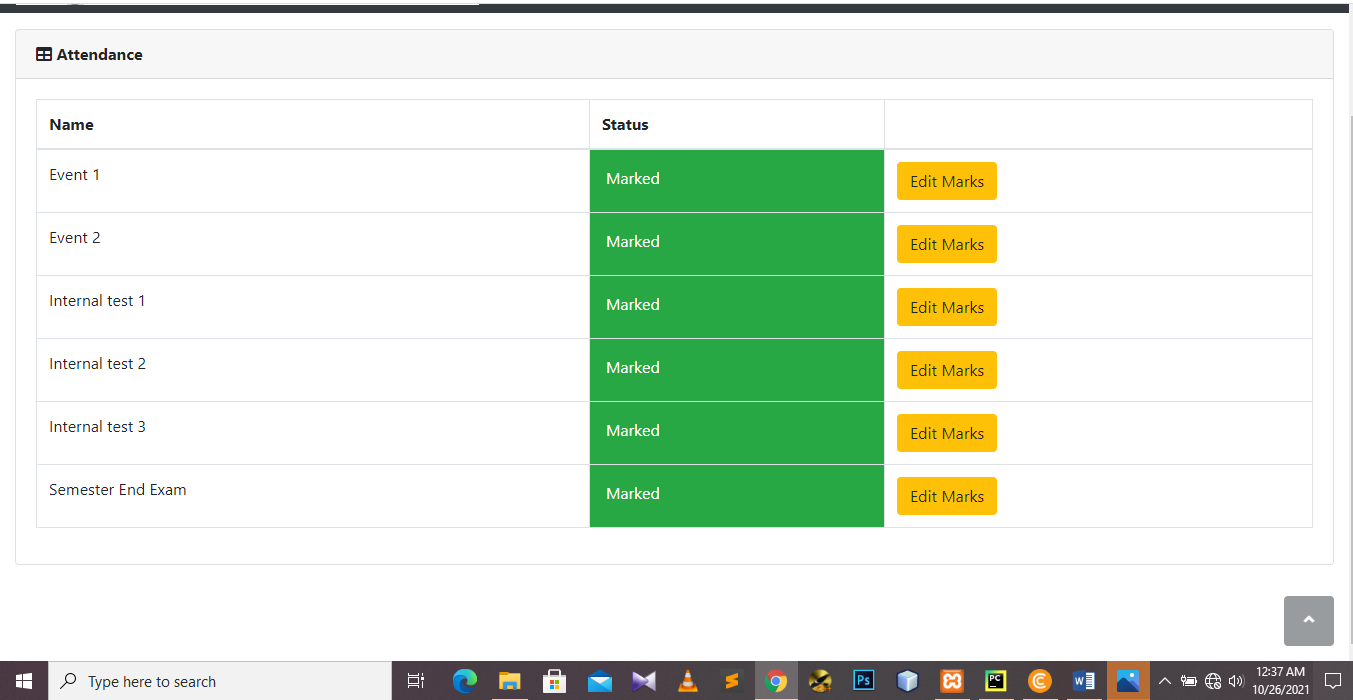
**Figure4.13 Attendance of Students in Class**

**4.2.2.4 Marks**

On this page, the list of classes assigned to the teacher are displayed along with two actions for eachclass. These actions are,

**Enter Marks**

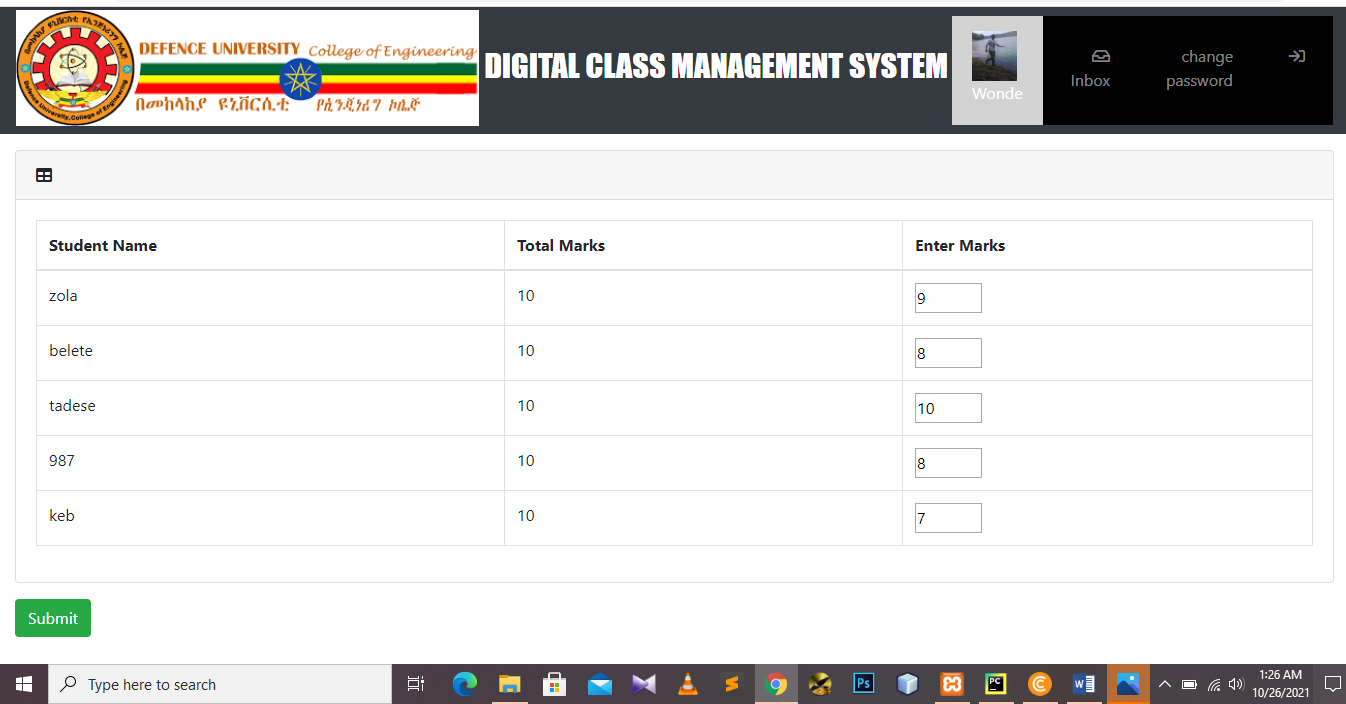
On this page, the teacher can enter the marks for 5 internal assessments, and 1 semester end exam. Initially all of them are marked red to denote that the marks have not been entered yet. Once the marks for a test is entered, it turns green. While entering the marks for a particular test, the list of 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.

****

**Figure4.14 Entering Marks**

**Edit Marks**

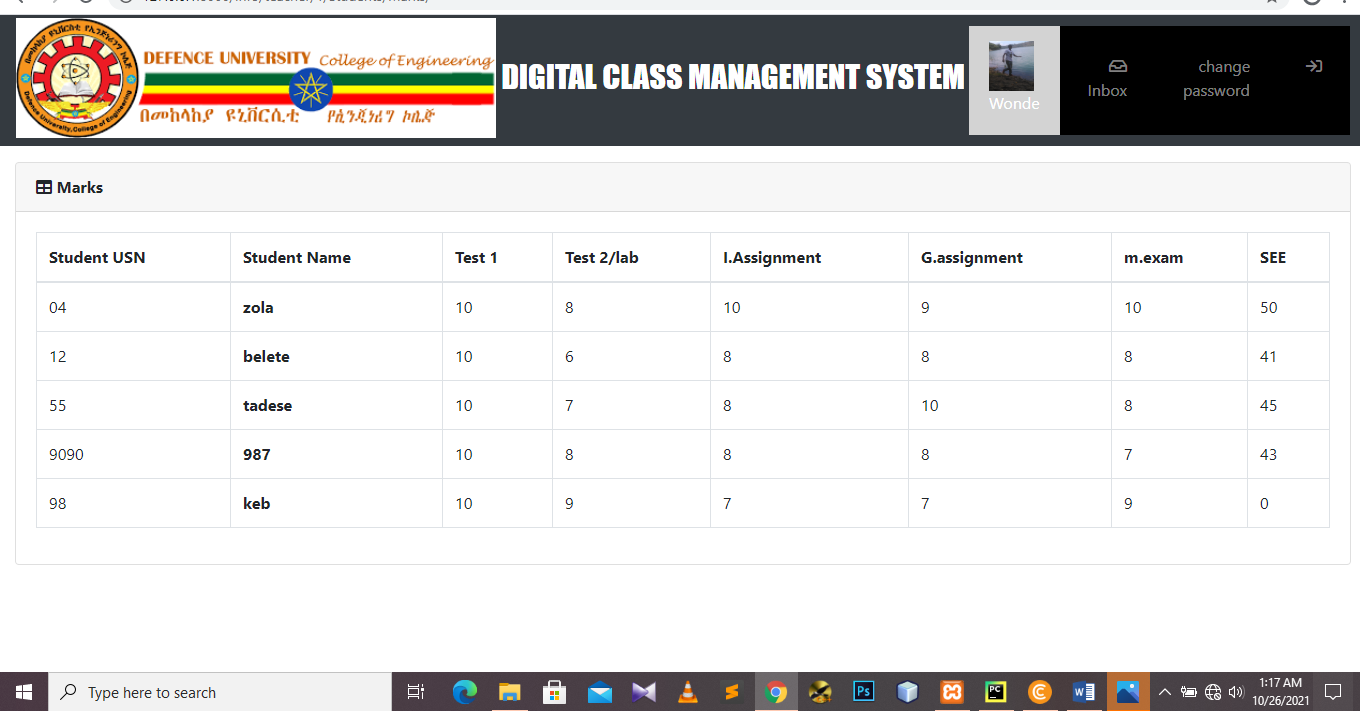
Marks for a test can be edited. While editing, the list of students in that class is displayed along with already entered marks. The marks to be updated can be changed and submitted. The students can view this change immediately.

****

**Figure4.15 Editing Marks**

**Student Marks**

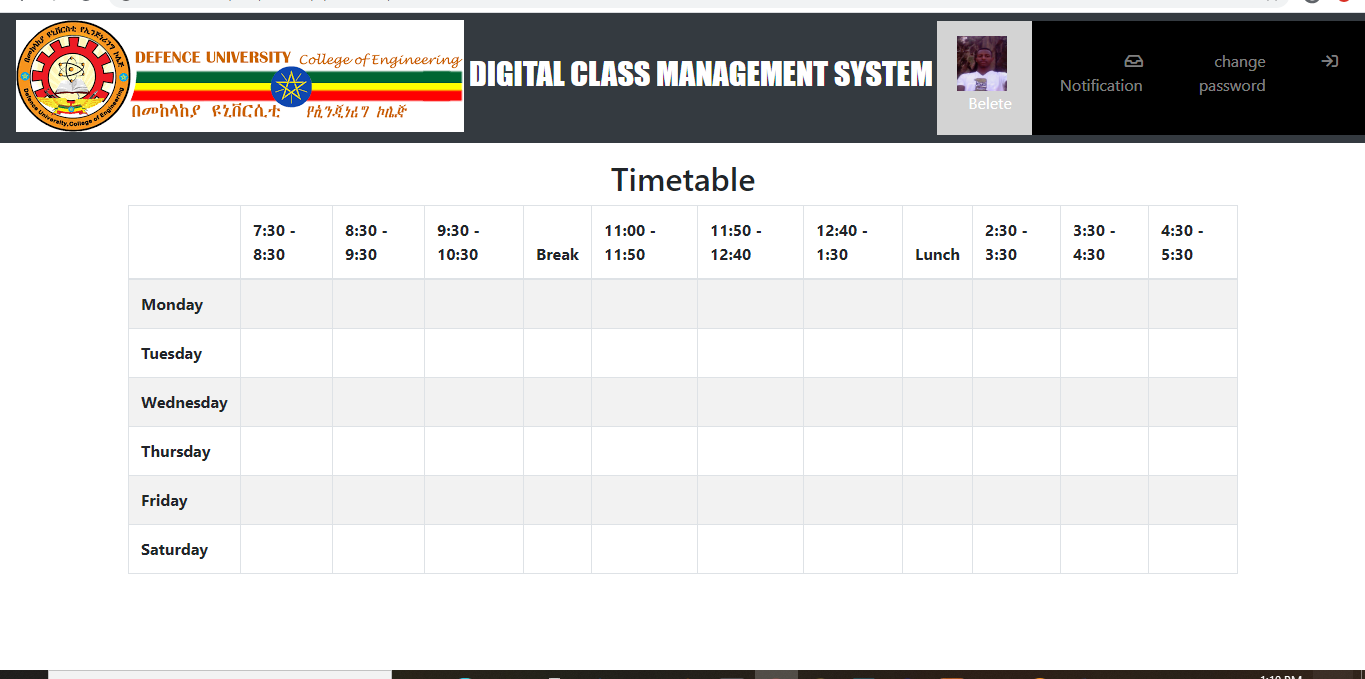
For each assigned class, the teacher has access to the list of students and the marks they obtained in all the tests. This is displayed in a tabular form.



**Figure4.16 Marks of all the students in a class**

**4.2.2.5 Timetable**

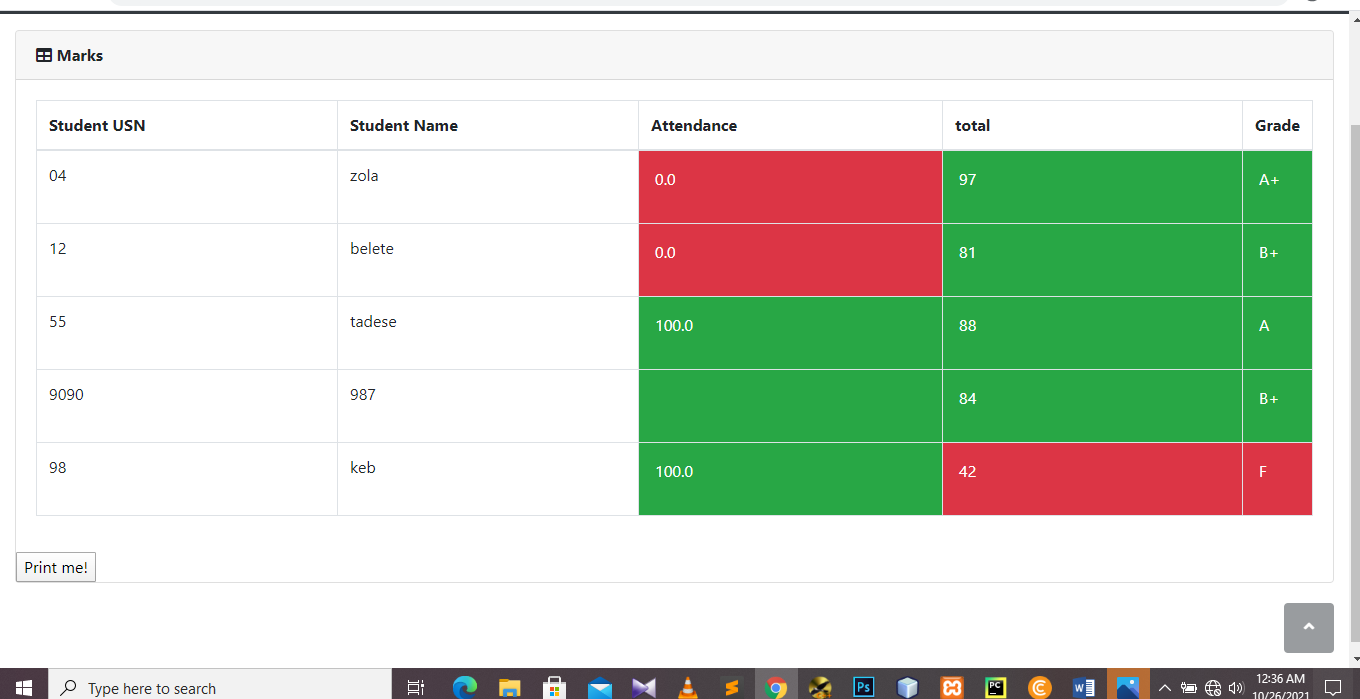
This page is a table which lists the day and timings of each of the classes assigned to the teacher. The row headers are the days of the week and the column headers are the time slots. So, for each day, it speciﬁes the classes in the time slots. The timetable is generated automatically from the assign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

****

**Figure4.17 Teacher Timetable**

**4.2.2.6 Reports**

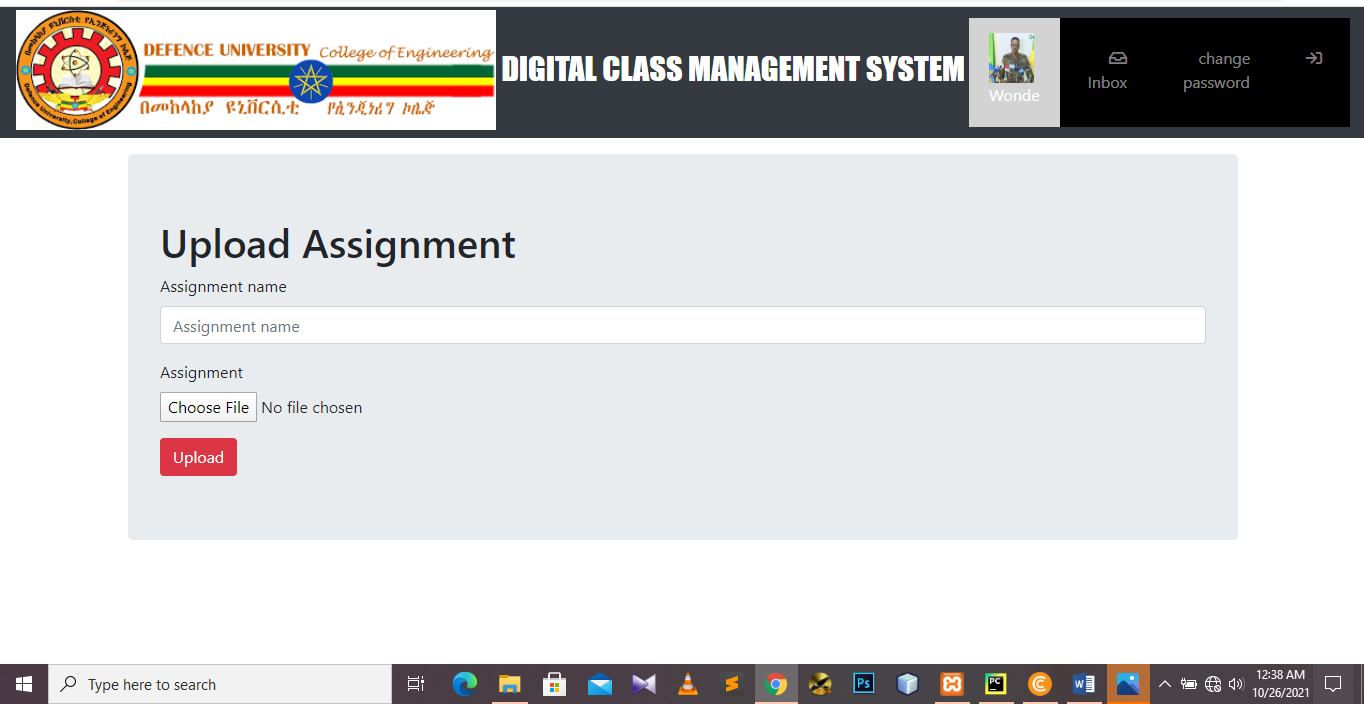
This page for the teachers is used to generate reports for each class. The report speciﬁes the list of students in that class and their respective total mark, Grade and attendance percentage. Total mark is the average of the marks obtained from the tests, 5 assessments. The SEE is out of 50 the attendance percentage is displayed with students below 75% marked in red. And if the total mark is below 44 the grade displayed in red



**Figure4.18 grade and attendance for a class of students**

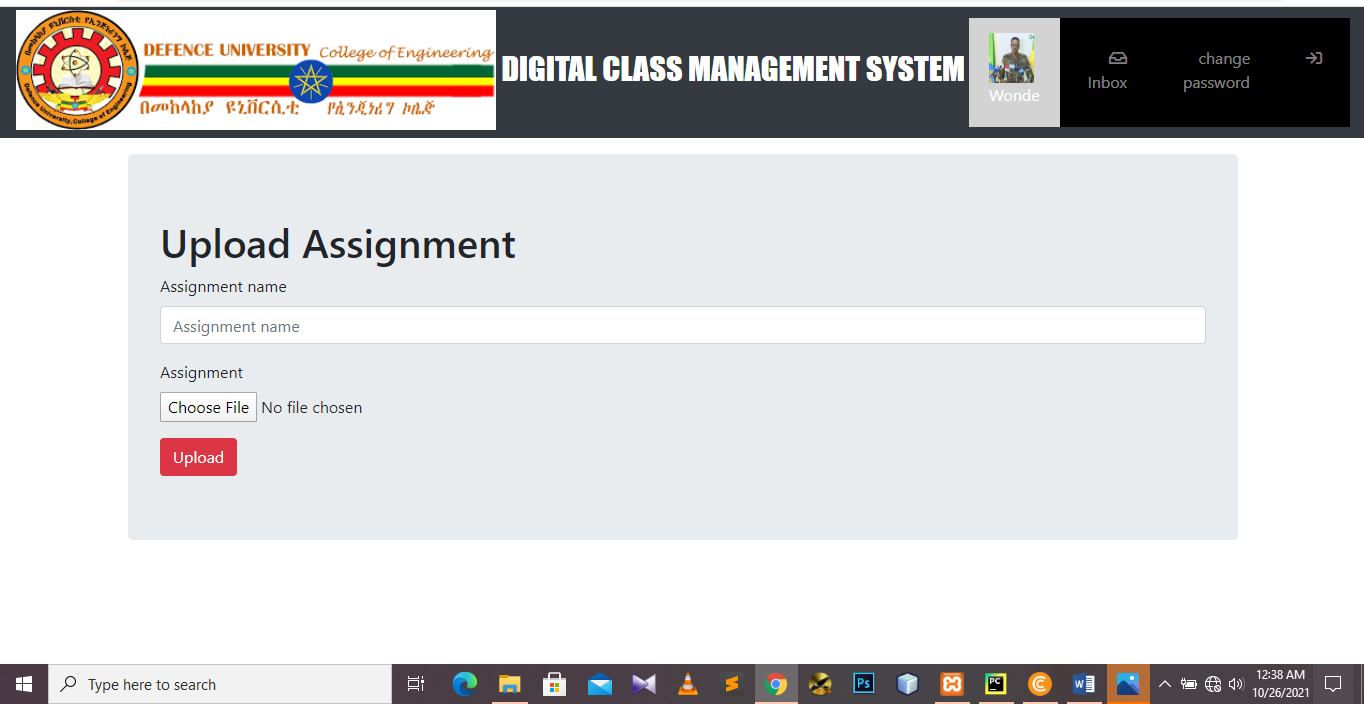
**4.2.2.7 Upload assignment**

This page helps teacher to upload assignment for their students.it contains assignment name and file content. If you click choose button it redirects you to a file storage. Click upload to send assignment for students in the class.



**Figure4.19 upload assignment for student**

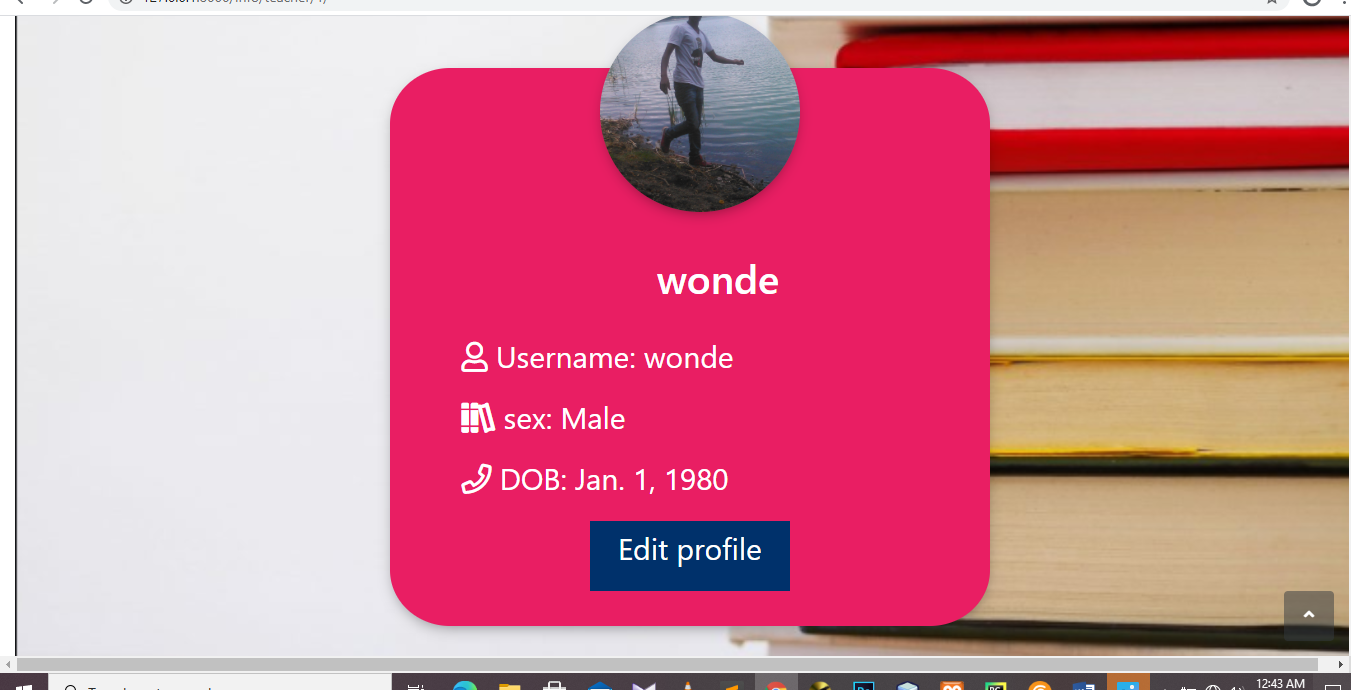
**4.2.2.8 Write notice**

Notice will be sent to every student in your class. From this page teacher can send any information to students.it contains text area and add notice button. If you click add notice the message sent successfully to all student in the class

**Figure4.20 Class notice page**

**4.2.2.9 Update profile**

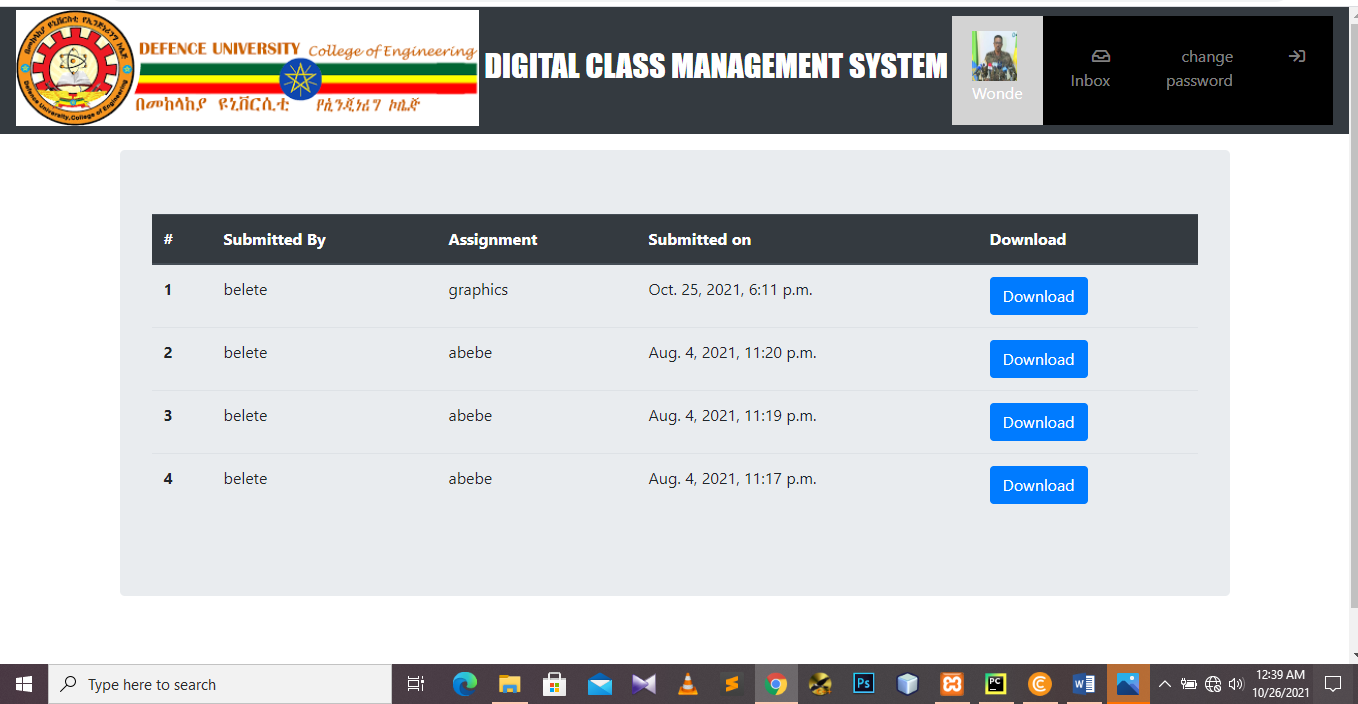
Update User Profile. Your user profile contains the personal information that is necessary for you using this Profile page you can update your profile photo, you can update your gender and other personal information



**Figure4.21 update teacher profile page**

**4.2.2.10 View assignment submitted**

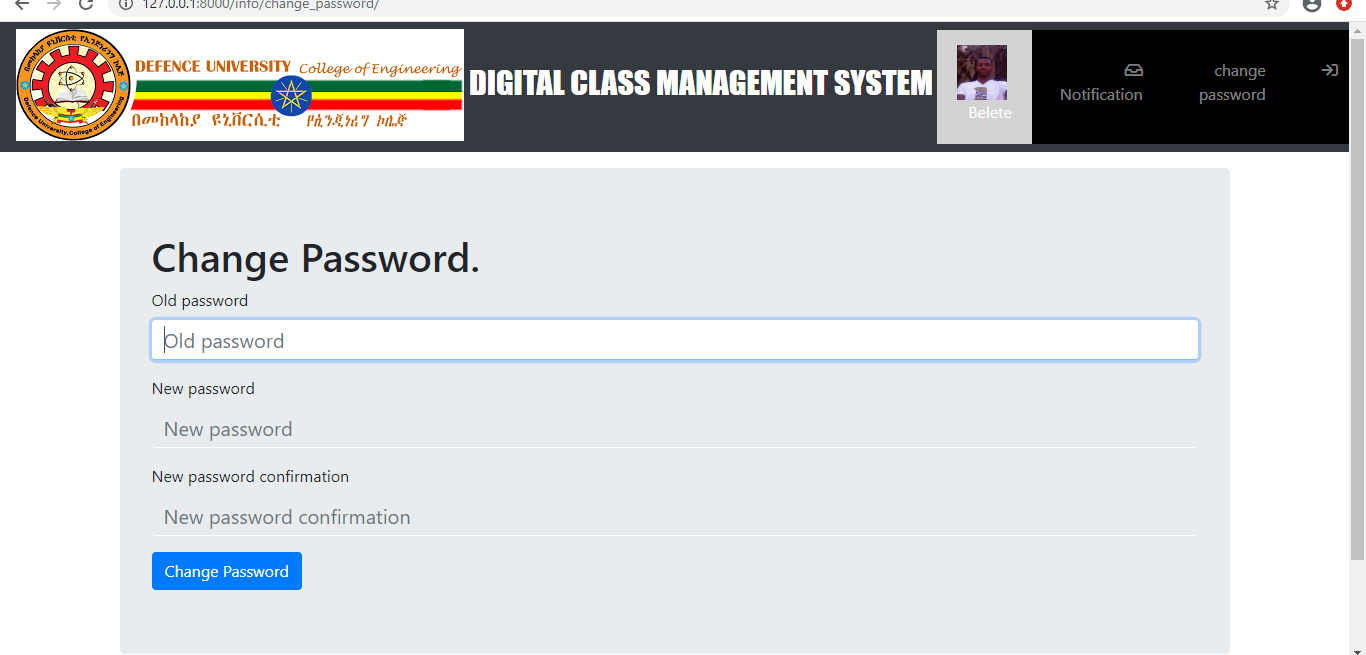
On this page, there is assignment name, student name who submitted assignment, submitted date download button and assignment content for each course. If there is no assignment uploaded yet it displays not assignment submitted yet.

****

**Figure4.22 submitted assignment list.**

**4.2.2.11 Change password**

This page helps user to create their own password. Before you change password you will need to have your current password**.**

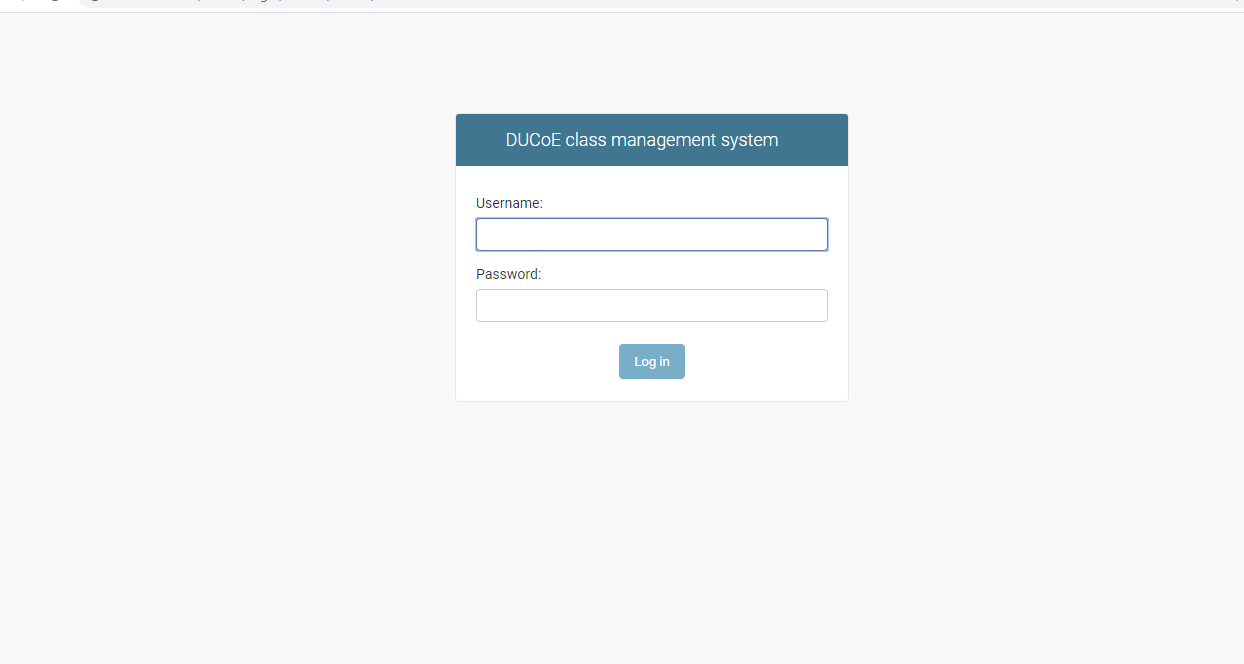
****

**Figure4.23 Change password.**

##### 4.2.3 Admin page

**4.2.3.1 Login**

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

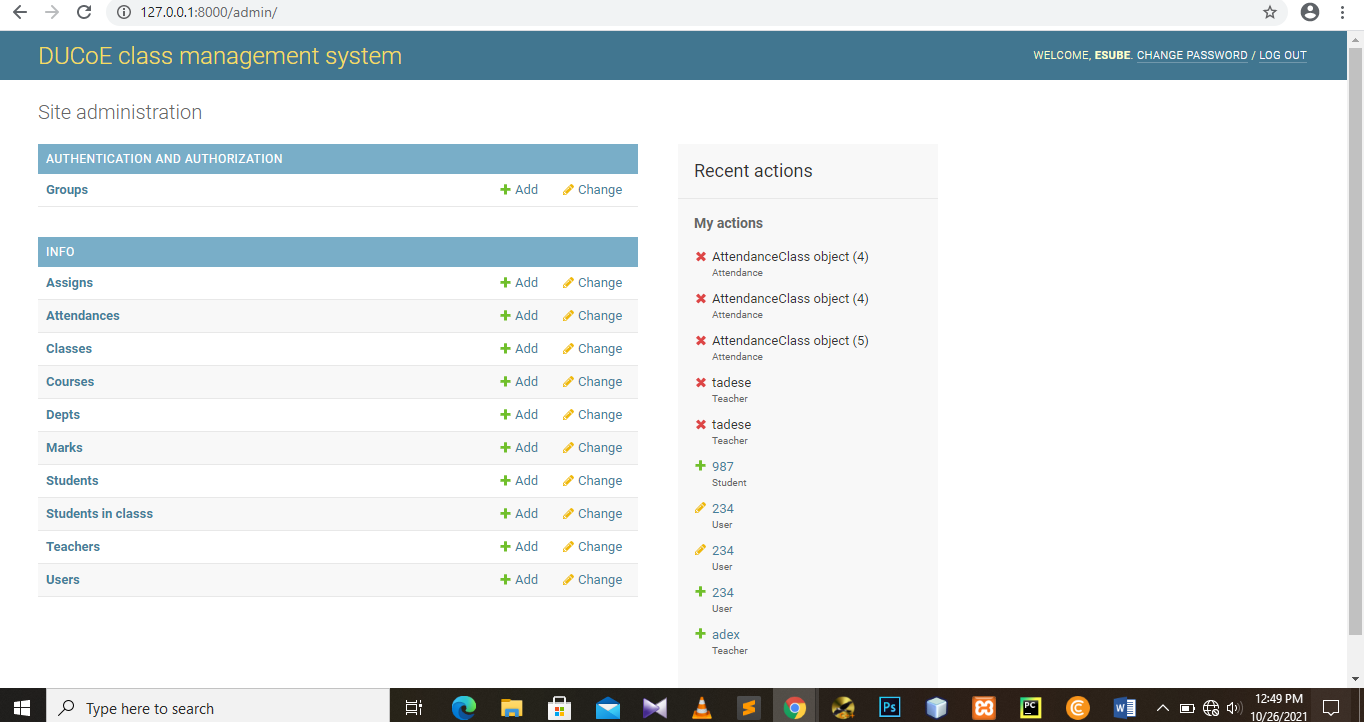
****

**Figure4.24 admin Login page.**

**4.2.3.2 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.

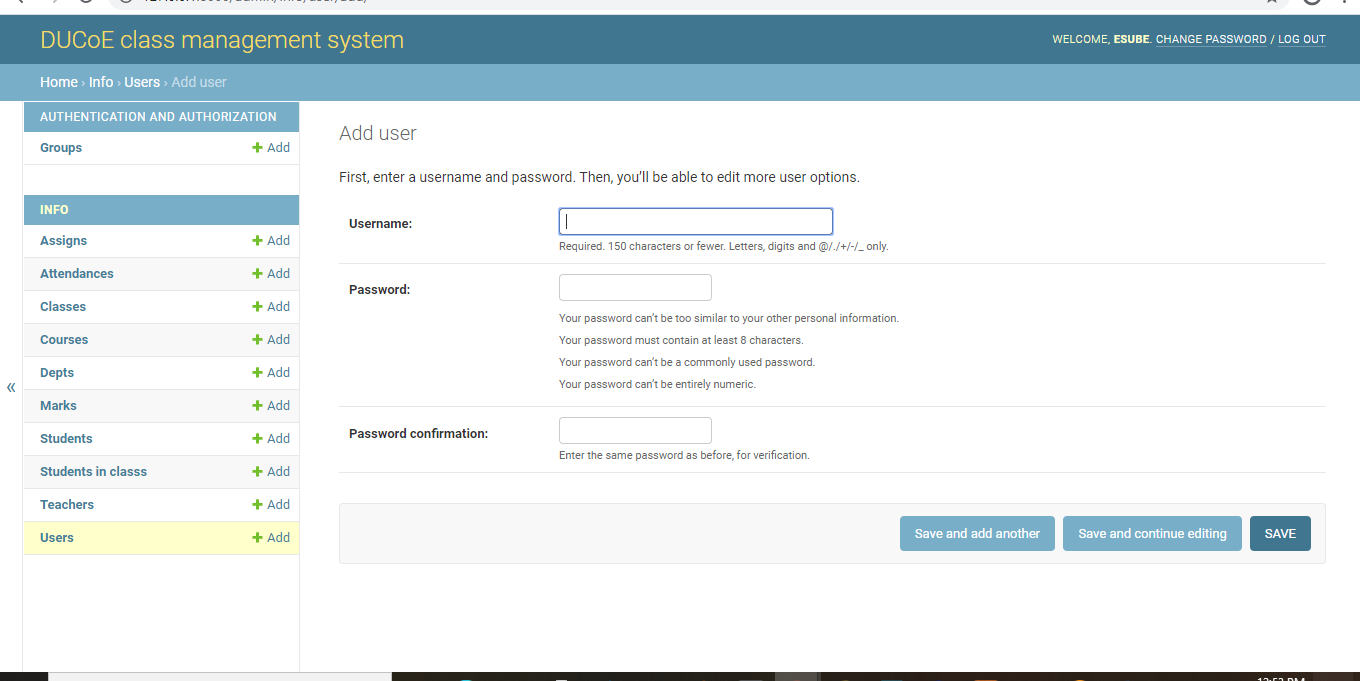
There are several features in place to ensure that querying the database is quick and eﬃcient for the administrator. As the database has the potential to become huge, there is a search feature for every table including student, teacher etc. The search has get a speciﬁc record based on name or id. Also,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

****

**Figure4.25 admin home page.**

**4.2.3.3 Add new user**

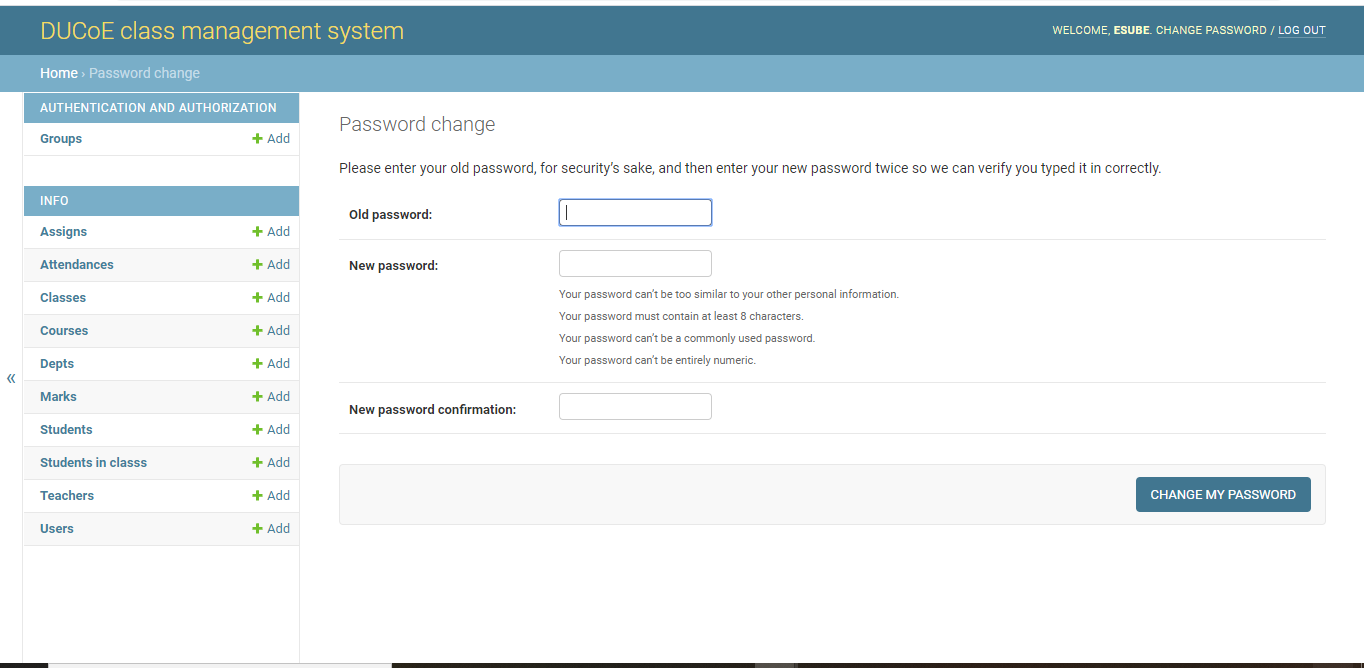
The administrator is responsible for adding and maintaining user. This page helps the admin to register new user

****

**Figure4.26 admin add user page.**

**4.2.3.4 Change password**

This page helps user to create their own password. Before you change password you will need to have your current password**.**

****

**Figure4.27 admin change password page.**

**Users- activity description table**

###### 

###### Table4-1 Users- activity description

|  |  |  |  |
| --- | --- | --- | --- |
| Activities | Admin | Teacher | Student |
| Manage Users Profile | yes | yes | yes |
| Create Department | yes | no | no |
| Create Class | yes | no | no |
| Create Teachers | yes | no | no |
| Manage Timetable | yes | no | no |
| Manage Assignment | yes | no | no |
| Change Users Role | yes | no | no |
| Update student result | yes | no | no |
| Upload assignment | no | yes | no |
| Take attendance | no | yes | no |
| Write message | no | yes | yes |
| Report | no | yes | no |
| Submit assignment |  |  | yes |
| Enter marks | yes | yes | no |
| Change password | yes | yes | yes |

##### 

##### 4.3 Database design

Database design is the organization of data according to a database model.

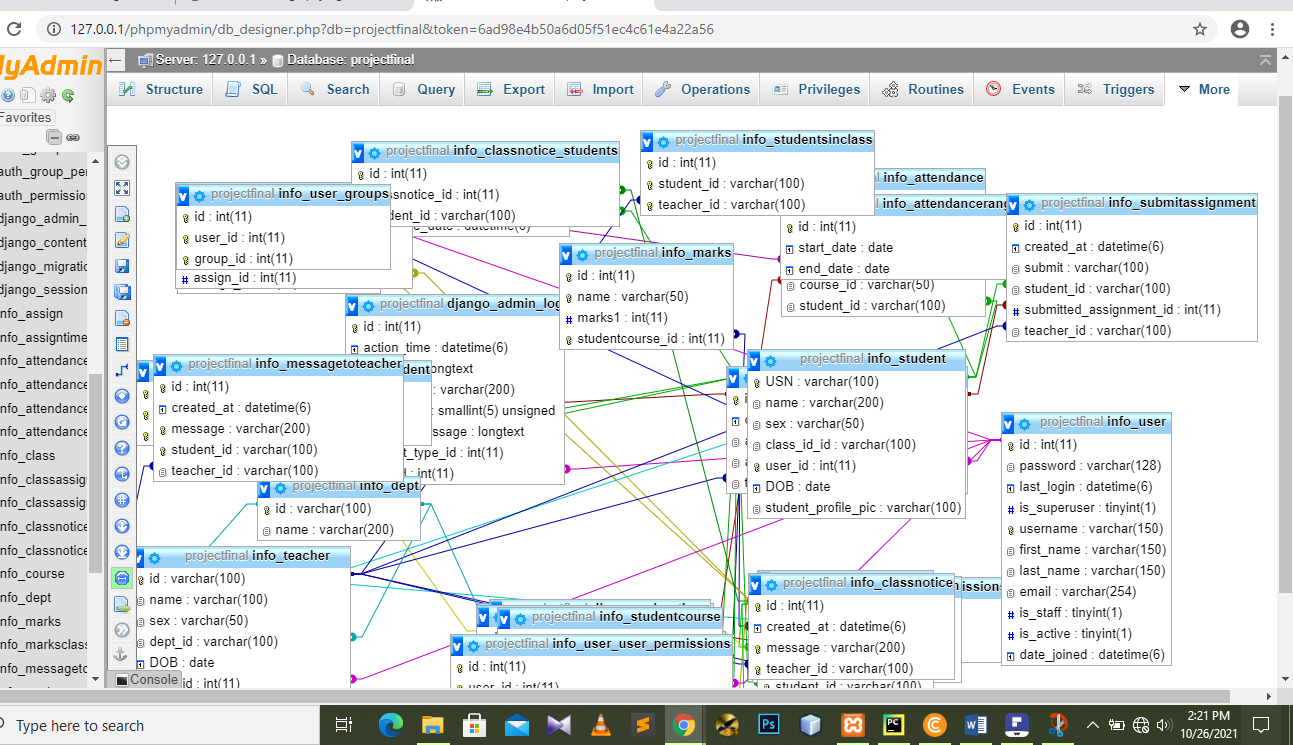


Figure4.28 Design of the database

##### 

##### 4.4 Sample Code

from django.shortcuts import render, get\_object\_or\_404  
from django.views import generic  
from django.views.generic import (View,TemplateView,  
 ListView,DetailView,  
 CreateView,UpdateView,  
 DeleteView)  
from django.utils.decorators import method\_decorator  
from django.contrib.auth.mixins import LoginRequiredMixin  
  
from info.forms import AssignmentForm,SubmitForm,NoticeForm,MessageForm,TeacherProfileUpdateForm,StudentProfileUpdateForm  
from django.http import HttpResponseRedirect  
from .models import Dept, Class, Student, Attendance, Course, Teacher, Assign, AttendanceTotal, time\_slots, \  
 DAYS\_OF\_WEEK, AssignTime, AttendanceClass, StudentCourse, Marks, MarksClass,SubmitAssignment,ClassAssignment,StudentsInClass  
from django.urls import reverse  
from info import models  
from info.models import StudentsInClass,ClassAssignment,SubmitAssignment,Student,Teacher  
from django.shortcuts import redirect  
from django.contrib import messages  
from django.contrib.auth import authenticate, login, logout,update\_session\_auth\_hash  
from django.contrib.auth.forms import PasswordChangeForm  
from django.utils import timezone  
from django.contrib.auth.decorators import login\_required  
from django.db.models import Q  
  
  
# Create your views here.  
  
  
@login\_required  
def index(request):  
 if request.user.is\_teacher:  
 return render(request, 'info/t\_homepage.html')  
 if request.user.is\_student:  
 return render(request, 'info/homepage.html')  
 return render(request, 'info/logout.html')  
  
  
@login\_required()  
def attendance(request, stud\_id):  
 stud = Student.objects.get(USN=stud\_id)  
 ass\_list = Assign.objects.filter(class\_id\_id=stud.class\_id)  
 att\_list = []  
 for ass in ass\_list:  
 try:  
 a = AttendanceTotal.objects.get(student=stud, course=ass.course)  
 except AttendanceTotal.DoesNotExist:  
 a = AttendanceTotal(student=stud, course=ass.course)  
 a.save()  
 att\_list.append(a)  
 return render(request, 'info/attendance.html', {'att\_list': att\_list})  
  
  
@login\_required()  
def attendance\_detail(request, stud\_id, course\_id):  
 stud = get\_object\_or\_404(Student, USN=stud\_id)  
 cr = get\_object\_or\_404(Course, id=course\_id)  
 att\_list = Attendance.objects.filter(course=cr, student=stud).order\_by('date')  
 return render(request, 'info/att\_detail.html', {'att\_list': att\_list, 'cr': cr})  
  
  
# Teacher Views  
  
@login\_required  
def t\_clas(request, teacher\_id, choice):  
 teacher1 = get\_object\_or\_404(Teacher, id=teacher\_id)  
 return render(request, 'info/t\_clas.html', {'teacher1': teacher1, 'choice': choice})  
  
  
@login\_required()  
def t\_student(request, assign\_id):  
 ass = Assign.objects.get(id=assign\_id)  
 att\_list = []  
 for stud in ass.class\_id.student\_set.all():  
 try:  
 a = AttendanceTotal.objects.get(student=stud, course=ass.course)  
 except AttendanceTotal.DoesNotExist:  
 a = AttendanceTotal(student=stud, course=ass.course)  
 a.save()  
 att\_list.append(a)  
 return render(request, 'info/t\_students.html', {'att\_list': att\_list})  
  
  
@login\_required()  
def t\_class\_date(request, assign\_id):  
 now = timezone.now()  
 ass = get\_object\_or\_404(Assign, id=assign\_id)  
 att\_list = ass.attendanceclass\_set.filter(date\_\_lte=now).order\_by('-date')  
 return render(request, 'info/t\_class\_date.html', {'att\_list': att\_list})  
  
  
@login\_required()  
def cancel\_class(request, ass\_c\_id):  
 assc = get\_object\_or\_404(AttendanceClass, id=ass\_c\_id)  
 assc.status = 2  
 assc.save()  
 return HttpResponseRedirect(reverse('t\_class\_date', args=(assc.assign\_id,)))  
  
  
@login\_required()  
def t\_attendance(request, ass\_c\_id):  
 assc = get\_object\_or\_404(AttendanceClass, id=ass\_c\_id)  
 ass = assc.assign  
 c = ass.class\_id  
 context = {  
 'ass': ass,  
 'c': c,  
 'assc': assc,  
 }  
 return render(request, 'info/t\_attendance.html', context)  
  
  
@login\_required()  
def edit\_att(request, ass\_c\_id):  
 assc = get\_object\_or\_404(AttendanceClass, id=ass\_c\_id)  
 cr = assc.assign.course  
 att\_list = Attendance.objects.filter(attendanceclass=assc, course=cr)  
 context = {  
 'assc': assc,  
 'att\_list': att\_list,  
 }  
 return render(request, 'info/t\_edit\_att.html', context)  
  
  
@login\_required()  
def confirm(request, ass\_c\_id):  
 assc = get\_object\_or\_404(AttendanceClass, id=ass\_c\_id)  
 ass = assc.assign  
 cr = ass.course  
 cl = ass.class\_id  
 for i, s in enumerate(cl.student\_set.all()):  
 status = request.POST[s.USN]  
 if status == 'present':  
 status = 'True'  
 else:  
 status = 'False'  
 if assc.status == 1:  
 try:  
 a = Attendance.objects.get(course=cr, student=s, date=assc.date, attendanceclass=assc)  
 a.status = status  
 a.save()  
 except Attendance.DoesNotExist:  
 a = Attendance(course=cr, student=s, status=status, date=assc.date, attendanceclass=assc)  
 a.save()  
 else:  
 a = Attendance(course=cr, student=s, status=status, date=assc.date, attendanceclass=assc)  
 a.save()  
 assc.status = 1  
 assc.save()  
  
 return HttpResponseRedirect(reverse('t\_class\_date', args=(ass.id,)))

##### CHAPTER FIVE

##### 5. Conclusion and Recommendation

##### 5.1. Introduction

This chapter is concerned with concluding the work that has been done, the recommendations that the users should follow when using the system and future woks that needs to be done to make the system more user friendly and available for all users.

##### Conclusion

##### By using Existing System 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 digital class management System 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 staff to easy access to ﬁnding the details. The college DCMS provides appropriate in- formation 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.

##### Recommendation

From the various facts that constitute the project, the following recommendations have been drawn that will be added in the system for future.

Exam scheduling

qr-code based attendance taking mechanism

Include registrar as a user

##### 

**References**

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

8. https://www.djangoproject.com/

9. https://getbootstrap.com/

10. https://www.tutorialspoint.com/

11. https://creately.com/

12. https://www.overleaf.com/project