PES UNIVERSITY

B.TECH. (CSE)

V SEMESTER

UE20CS303 –SOFTWARE ENGINEERING

PROJECT REPORT ON

STUDENT ATTENDANCE MANAGEMENT SYSTEM

SUBMITTED BY

1) SHIVANI ITAGI (PES1UG20CS398)
2) SHREYAS HIREMATH (PES1UG20CS406)
3) SHARATH KRISHNA (PES1UG20CS391)
4) SHRINIDHI KJ (PES1UG20CS412)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

TABLE OF CONTENTS

Synopsis/Project Proposal

Project description

Our project is a Student Attendance Management System. This system should help the institutional to streamline the administrative task and provide real-time access to the data. This software will be used by administrators and teachers to maintain and monitor attendance in educational institutions.

This project will enable the user to:

1) Mark if the Student is absent or present
2) Monitor Students attendance in each subject
3) Calculates the attendance percentage

Existing System:

The existing student attendance management system helps you mark if the student is present or absent. IN addition, the system displays all the available data such as instructor and student information, as well as their individual attendance. Admin Panel, Student Panel, and Teacher's Panel are the three sections of the project. In this web app's overview, the administrator has the ability to create users as well as insert student and teacher data. In terms of the project, the administrator has access to all student and teacher records. The teacher's account allows him or her to filter student data and keep track of his or her attendance for a certain subject.

Proposed System:

The proposed system aims to manage and view student records. The PhP based system will enable teachers to mark if a student is present or absent within a matter of seconds. The student records are maintained in a database, we have used SQL as our database.

Plan of work and product ownership:

The student attendance management system provides services to a large number of users. The system should identify individual students of the system by their roll number and name.

Input: Name, Roll number, Admission number, Class

Output: Attendance

Functional features planned to accomplish in the short term:

- Take attendance
- View class attendance
- View student attendance
- Download student’s attendance report
- Manage classes
- Manage teacher
- Manage student
- Manage session and term

SOFTWARE REQUIREMENTS SPECIFICATION

for

STUDENT ATTENDANCE MANAGEMENT SYSTEM

Version 1.0 approved

Prepared by

1) SHIVANI ITAGI (PES1UG20CS398)
2) SHREYAS HIREMATH (PES1UG20CS406)
3) SHARATH KRISHNA (PES1UG20CS391)
4) SHRINIDHI KJ (PES1UG20CS412)

PES UNIVERSITY

Table of Contents

1. INTRODUCTION

Student management system has become important factors in modern education field. This system should help the institutional to streamline the administrative task and provide real-time access to the data. The study findings enable the definition of the project problem statement, its objectives, scopes and advantages of the student management system.

PURPOSE:

The purpose this documents is to present a detailed description of the Student Management System. It will explain the purpose and features of the software, the interfaces of the software, what the software will do, the constraints under which it must operates and how the software will react to external stimuli. This document is intended for both the end users and the developers of the software.

SCOPE:

This document covers the requirements for the Student Management System. This software will provide a graphical environment in which the users of the system will be able to perform various operations that are associated with storing, marinating, updating and retrieving Student information. The purpose of this is to guide developers in selecting a design that will be able to accommodate the full-scale application. The system will capture information aboutstudent's personal details lectures and the courses. Storing updating and retrieving in a fast and accurate way.

2. OVERALL DESCRIPTION

The purpose this documents is to present a detailed description of the Student Management System. It will explain the purpose and features of the software, the interfaces of the software, what the software will do, the constraints under which it must operates and how the software will react to external stimuli. This document is intended for both the end users and the developers of the software.

PRODUCT PERSPECTIVE:

The product Student Management system, is an independent product and does not dependon any other product or system. The product will automate various tasks associated with handling student details and better organizing the stored information and optimum performance, thus helping the Colleges to ensure smooth working of these processes.

PRODUCT FUNCTIONS:

Our system has two types of accessing modes,

1) Administrator :

SMS is managed by Administrator. Administrator has to update and monitor the registered student details, add a new student, provide register number for all students, assign each student a course etc., Administrator can update his profile, and also can give help to the teachers and students.

2) Teacher :

User can add them onto the portal and view their schedules, marks attendance of the students, also can view the students details in graphical order, also of a single student andabout the views from the students.

USER CLASSES AND CHARACTERISTICS:

This software gives access to two kinds of users.

1) Administrator: The personnel and College administrator will have administrator access to add, delete and modify information stored in the database.

2) Authorized User: Teaching staff will have access to only view the data stored in the database and can update the student's attendance in the form of formatted reports.

ASSUMPTIONS AND DEPENDENCIES:

- We assume that the Office personnel do all the data entry based and the correct values obtained from forms and registers.
- We assume that the computers that will use the software will be part of the college LAN.
- Users with administrator access should be careful in deleting or modifying any information knowingly or unknowingly which will lead to inconsistency of the database.
- The end users of this software are assumed to have basic level of computer knowledge i.e. point and click.

3. EXTERNAL INTERFACE REQUIREMENTS:USER INTERFACE:

- GUI along with meaningful Frames and buttons.
- Reports are generated as per the requirement.

4. FUNCTIONAL REQUIREMENTS: STUDENT ATTENDANCE

MANAGEMENT:

- Easily track attendance information of students.

- Quickly produce single or multiple day attendance bulletins.

5. NON-FUNCTIONAL REQUIREMENTS:

STATIC REQUIREMENTS:

These requirements do not impose any constraints on the execution characteristics of the system. They are:

1) Number of Terminals: The software makes use of an underlying database that will reside at the server, while the front end will be available online to the administrative and departmental computers as well as students and teachers.

2) Number of Users: The number of users may vary, as this software finds applications in almost all department of the organization.

DYNAMIC REQUIREMENTS:

These specify constraints on the execution characteristics of the system. They typically include response time and throughout of the system. Since these factors are not applicable to the proposed software, it will suffice if the response tine is high and the transactions are carried out precisely and quickly.

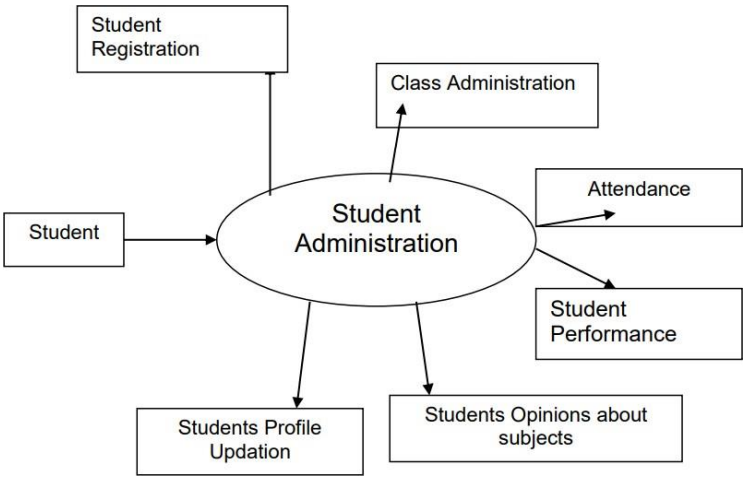
SECURITY:

The security requirements deal with the primary security. The software should be handled only by the administrator and authorized users. Only the administrator has right to assign permission like creating new accounts and generating password. Only authorized users can access the system with username and password.

DESIGN CONSTRAINTS:

This software provides security. The login form prevents the system from being misused by unauthorized users. Only an authorized operator will be granted rights to modify as per requirements. This software is also reliable and fault tolerant. The system developedis designed to handle invalid inputs. Since reliability is major area of concern the system has a backup to avoid data loss. The user should know the programming language very well that is used to develop a software.

6. ANALYSIS MODELS:



Project plan Document:

1: Identify the lifecycle to be followed for the execution of your project and justify why you have chosen the model.

We are using agile methodology in the software development lifecycle.

Agile SDLC methodology is based on collaborative decision making between requirements and solutions teams, and a cyclical, iterative progression of producing working software. Work is done in regularly iterated cycles, known as sprints, that usually last two to four weeks.

We are using this methodology because software is developed in incremental, rapid cycles. This results in small incremental releases with each release building on previous functionality. Each release is thoroughly tested to ensure software quality is maintained. It is used for time critical applications.

2: Identify the tools which u want to use it throughout the lifecycle like planning tool, design tool, version control, development tool, bug tracking, testing tool.

Tools used are:

JIRA for planning and bug tracking.
Github for version control.
Autodesk Product Design Suite for design,
VS code for development. (Language used Python)
Selenium IDE for testing.

3: Determine all the deliverables and categorise them as reuse/build components and justify the same.

Login module: Reusable. Can be used for different projects
Withdraw/deposit module: Build
Change PIN module: Reusable. Can be used for different projects as change password.
Home page module: Build
Transaction history: Build

4: Do a rough estimate of effort required to accomplish each task in terms of person months.

Approximately,

KLOC=1

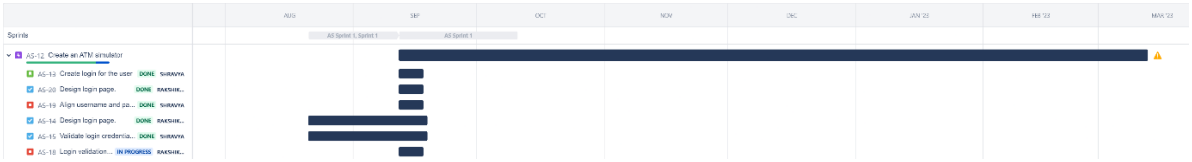
People doing the project: 4

Hence we project type is organic

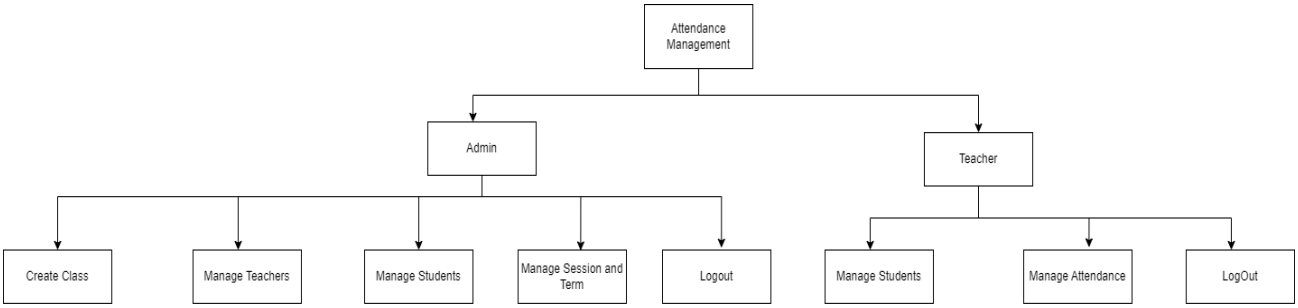
Estimation of effort re required to accomplish each task in terms of person months=

2.4 x (1) ^ 1.05 = 2.4 person months

5: Create the Gantt Chart for scheduling using any tool.



Design diagram



Test Plan Document

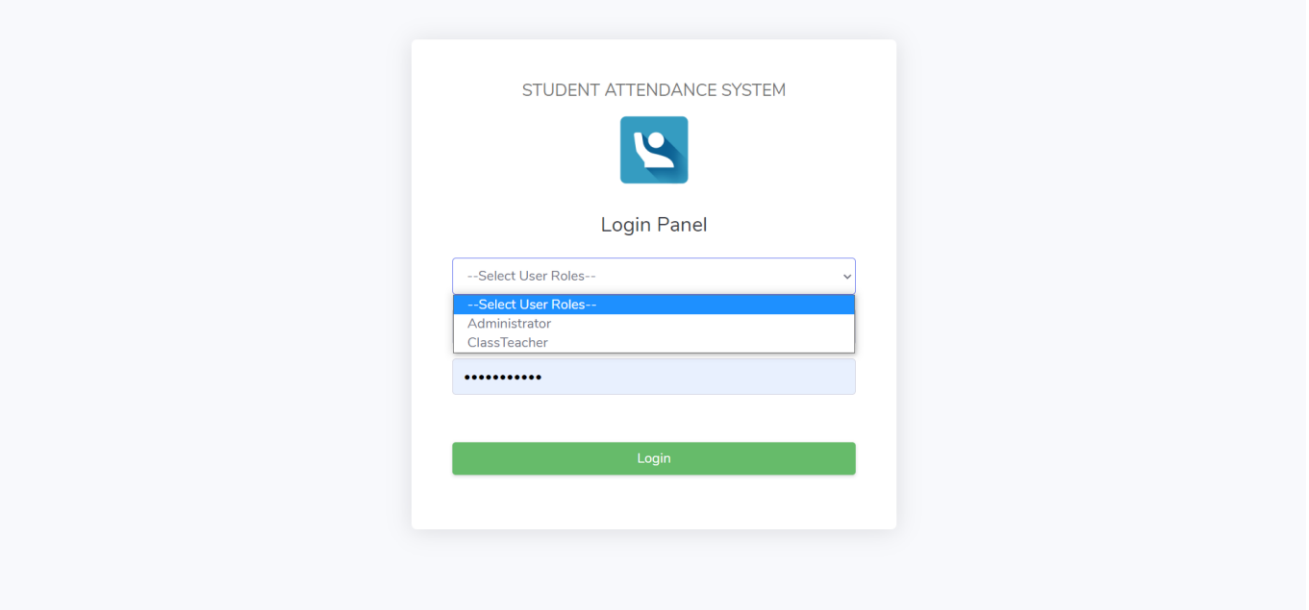
Test cases:

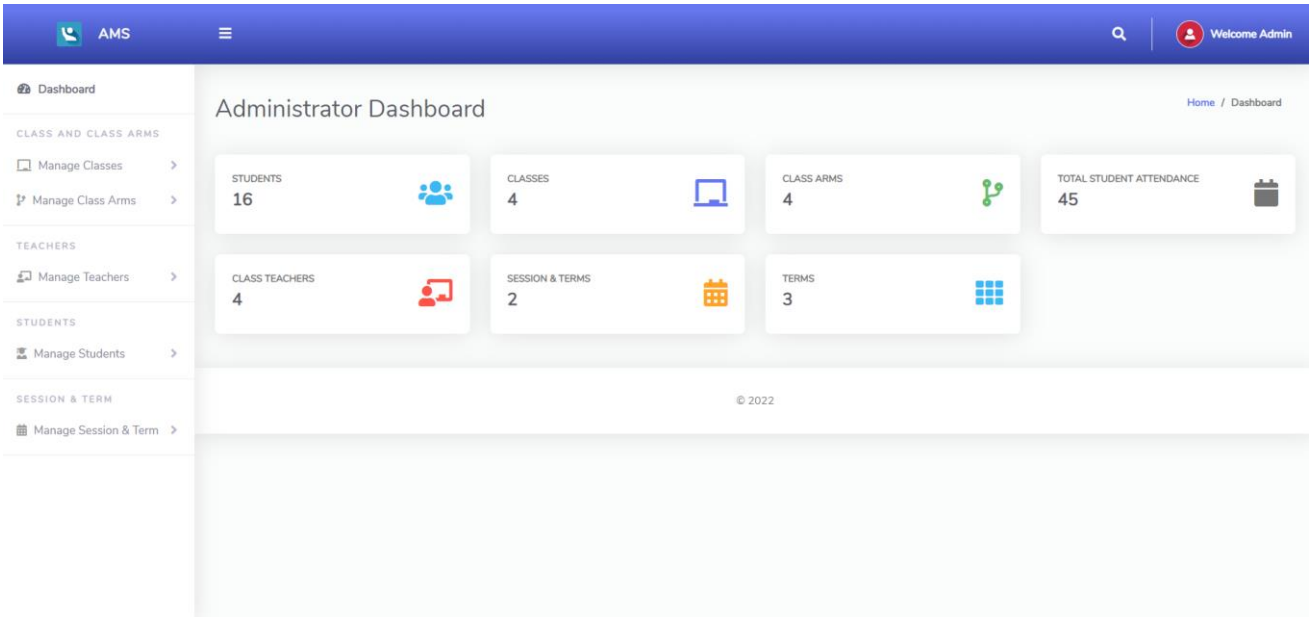
Test Case ID	Name of Module	Test case description	Pre-conditions	Test Steps	Test data	Expected Results	Actual Result	Test Result
1100	Check credentials module	To test the login functionality. When the password is invalid	Admin username and password should be entered	1) Choose teacher or administrator 2) Enter User name 3) Enter invalid password 4) Click enter	Email : admin@mail.com Password: Password@1234	Login should not be Successful. Display wrong password.	Login is not successful. Displays wrong password.	Pass
1234	Check credentialsmodule	To test the login functionality. When the details is valid.	Teacher username and password should be entered	1) Choose teacher or administrator 2) Enter Username Enter password 4) Click enter	Email : teacher@mail.com Password: pass123	Login should be Successful. Take to HomePage.	Login is Successful. Take to HomePage.	Pass
4537	Check credentials module	To test the login functionality. When the username isvalid.	Admin username and password should be entered.	1) Choose teacher or administrator 2) Enter Username Enter password 4) Click enter	Email : admin1@mail.com Password: Password@123	Login should not be Successful. Display wrong username.	Login should not be Successful. Display wrong username.	Pass
2839	Create Class Module.	Should Add a Class to the existing list of classes.	Change password option must be selected.	1) Navigate to Manage Classes. 2) Choose Create Class. 3) Enter Class Name. 4) Click on Save.	Class Name: Nine	Class Nine should be created.	Class Nine should be created.	Pass

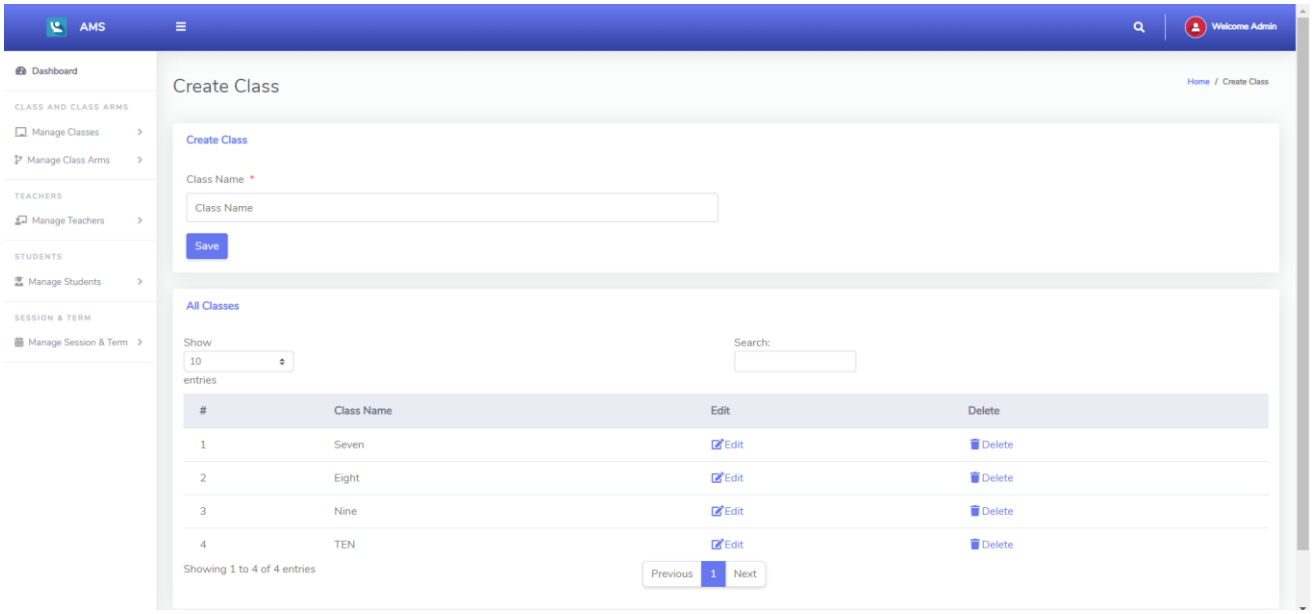
2543	Add Class Teachers.	Should be able to add Class Teachers.	Teacher details should be in the database.	1)Navigate to Manage Teachers. 2)Click on Create New Teachers. 3)Add Teacher Credentials. 4)Click on Save.	FirstName: Surabhi LastName: N Email: surabhi@gmail.com PhoneNo: 9343837654 Select Class: Nine	Surabhi N should be allotted as the class teacher for Nine.	Surabhi N should be allotted as the class teacher for Nine.	Pass
9043	Admission of Students.	Should be able to add students details to existing database.	Student Details should not exist in database.	1)Navigate to Manage Students. 2)Click on Create New Students. 3)Add Student Credentials. 4)Click on Save.	FirstName: Shivani LastName: Itagi OtherName: Gowri Admission Number: A1 Select Class: Nine ClassArm: N1	Shivani Itagi Gowri Should be added to student database and allotted to class Nine.	Shivani Itagi Gowri Should be added to student database and allotted to class Nine.	Pass
1267	Take Attendance.	Teachers should be able to take attendance of students.	Students Name should be in the attendance sheet.	1)Navigate to Manage Attendance. 2)Choose Take Attendance. 3)Update Student Attendance by crossing the checklist. 4)Click on Update Attendance	Click on Checkbox.	Student Attendance should get updated.	Student attendance gets updated.	Pass
9753	View Class Attendance	Teachers should be able to view entire class attendance.	Must be a teacher.	1)Navigate to Manage Attendance 2) View Class Attendance.	Select Date.	Must be able to see the class attendance for the particular day selected.	Able to see the class attendance for the particular day selected	Pass

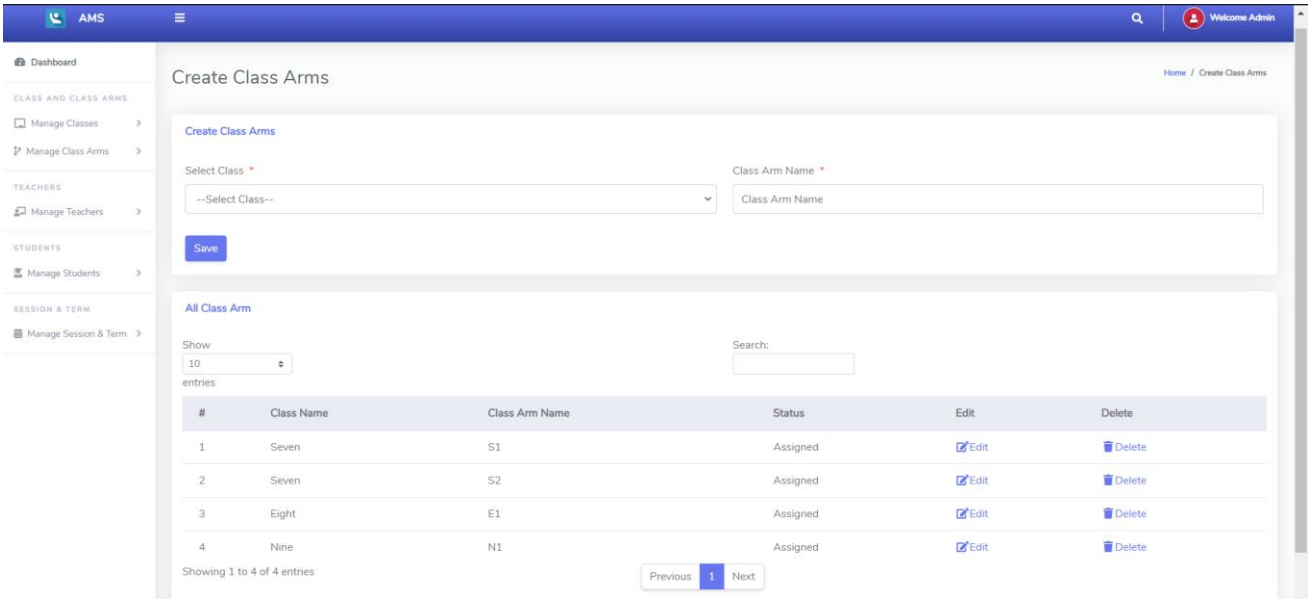
- **Test Case ID :** Each test case should be represented by a unique ID. To indicate test types, follow some convention like "UT_01" indicating "Unit Testing - Test Case#1."
- **Name of the module :** Specify the name of the **main module or sub module** being tested
- **Test Case Description :** Specify the summary or test purpose in brief
- **Pre- Conditions :** Any requirement that needs to be done before execution of this test case.
- **Test Steps** : Mention all the steps in detail and specify the order in which it is to be executed.
- **Test Data :** Input for the test case to be executed. Specify different data sets with precise values to be used as input. (create test case for both valid and invalid inputs)
- **Expected Results :** Mention the expected results including error or precise messages that should be displayed on screen
- **Actual Results :** After execution of test case fill this column with the result obtained
- **Test Result (Pass/Fail) :** Mark this field as “fail” if the actual result is not same as expected result else mark as “pass”.

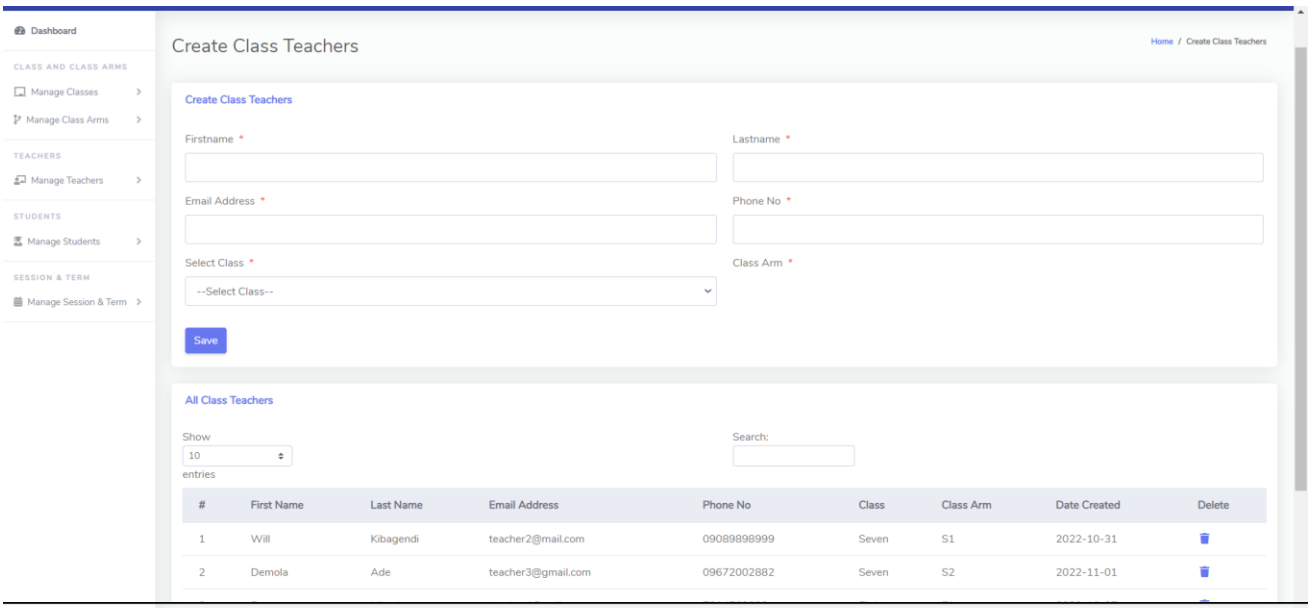
Screenshots of the output:

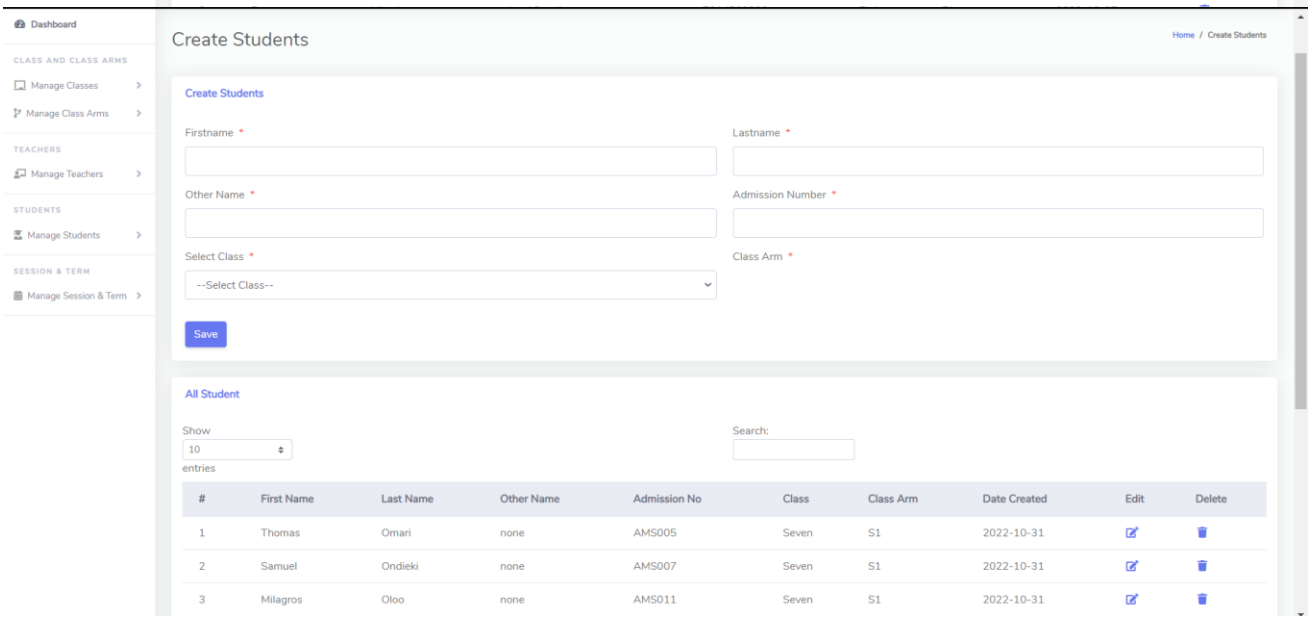


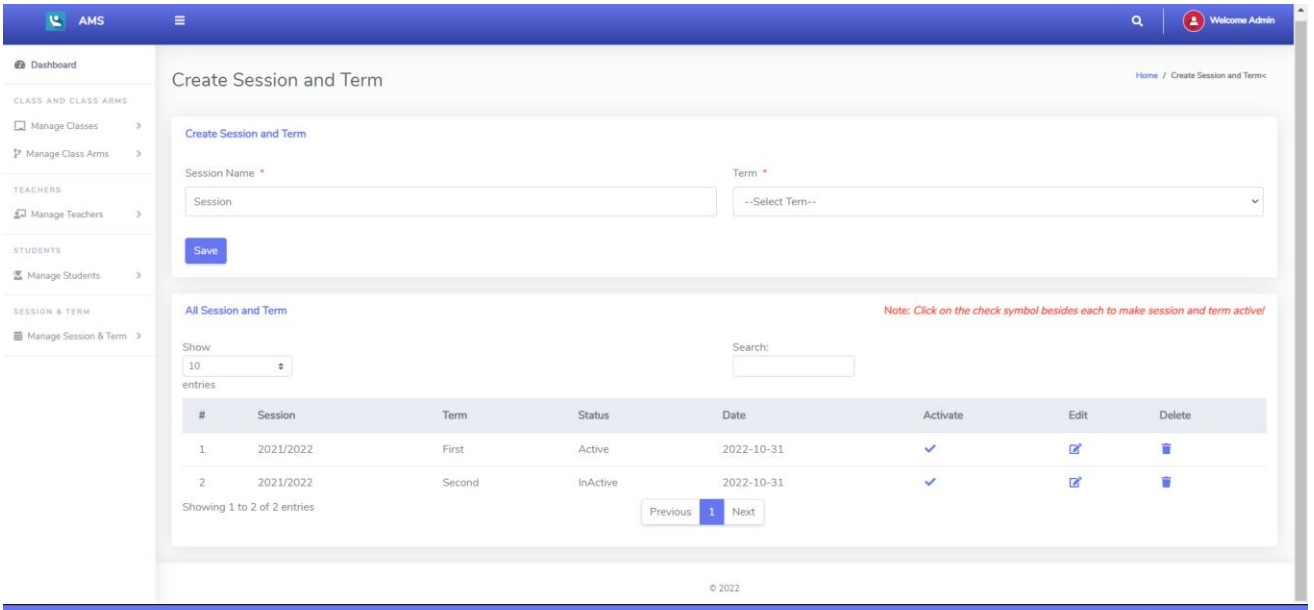


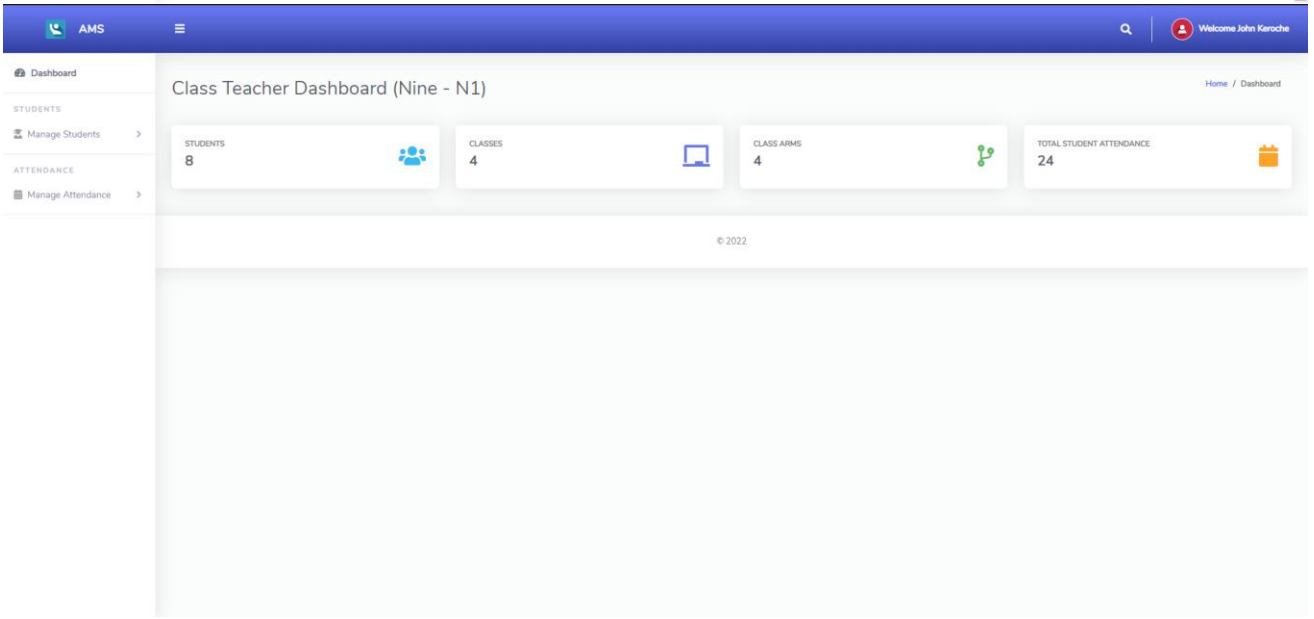

AMS
Welcome Admin
Dashboard
CLASS AND CLASS ARMS
Manage Classes
Manage Class Arms
TEACHERS
Manage Teachers
STUDENTS
Manage Students
SESSION & TERM
Manage Session & Term
Create Class
Home / Create Class
Create Class
Class Name *
Class Name
Save
All Classes
Show
10
entries
Search:
Class Name Edit Delete
1 Seven Edit Delete
2 Eight Edit Delete
3 Nine Edit Delete
4 TEN Edit Delete
Showing 1 to 4 of 4 entries
Previous 1 Next

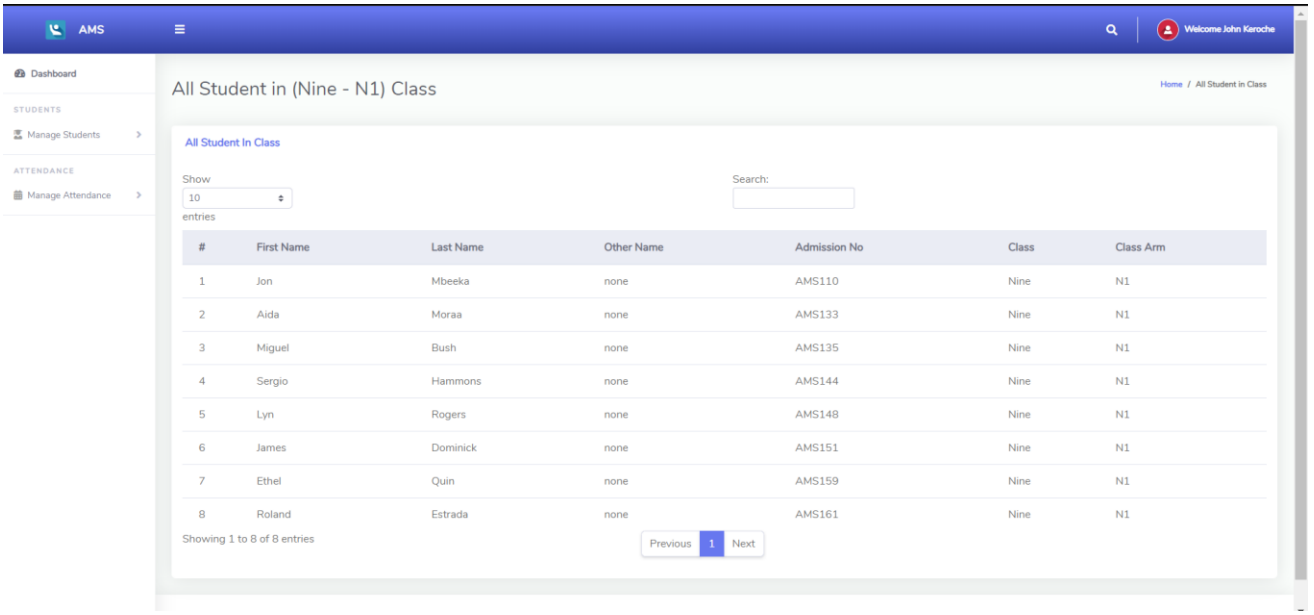

AMS
Welcome Admin
Dashboard
CLASS AND CLASS ARMS
Manage Classes
Manage Class Arms
TEACHERS
Manage Teachers
STUDENTS
Manage Students
SESSION & TERM
Manage Session & Term
Create Class Arms
Home / Create Class Arms
Create Class Arms
Select Class *
--Select Class--
Class Arm Name *
Class Arm Name
Save
All Class Arm
Show
10
entries
Search:
Class Name Class Arm Name Status Edit Delete
1 Seven S1 Assigned Edit Delete
2 Seven S2 Assigned Edit Delete
3 Eight E1 Assigned Edit Delete
4 Nine N1 Assigned Edit Delete
Showing 1 to 4 of 4 entries
Previous 1 Next

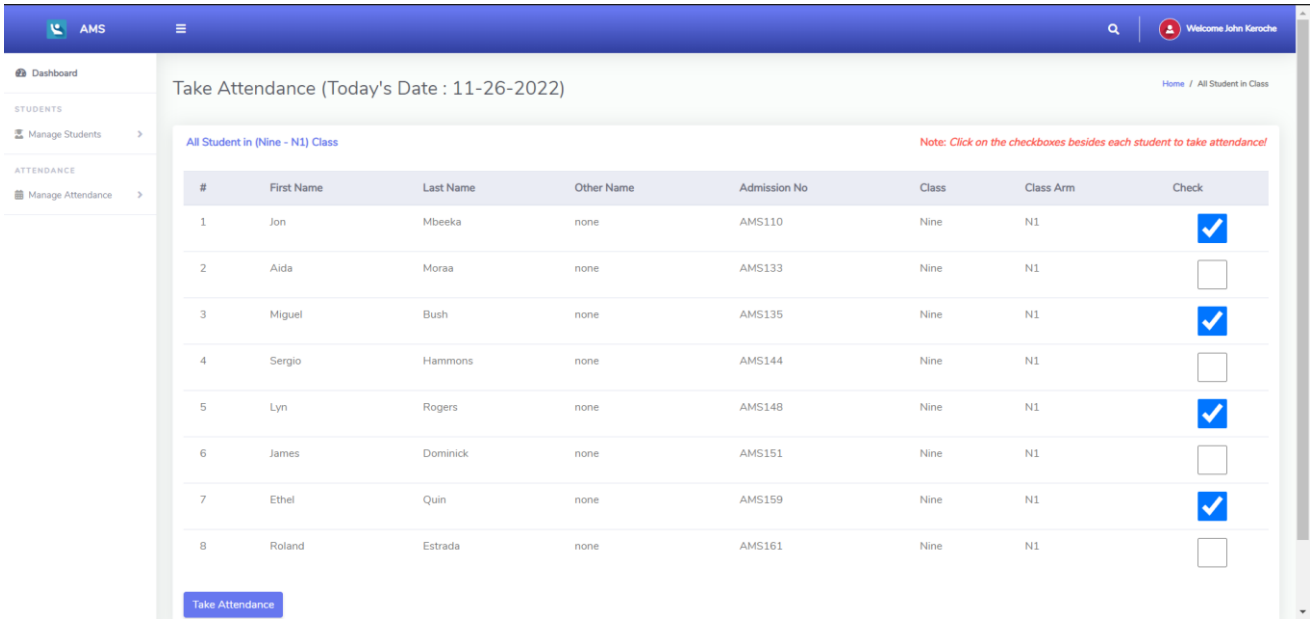

Dashboard
CLASS AND CLASS ARMS
Manage Classes
Manage Class Arms
TEACHERS
Manage Teachers
STUDENTS
Manage Students
SESSION & TERM
Manage Session & Term
Create Class Teachers
Home / Create Class Teachers
Create Class Teachers
Firstname *
Lastname *
Email Address *
Phone No *
Select Class *
--Select Class--
Class Arm *
Save
All Class Teachers
Show
10
entries
Search:
First Name Last Name Email Address Phone No Class Class Arm Date Created Delete
1 Will Kibagendi teacher2@mail.com 09089898999 Seven S1 2022-10-31
2 Demola Ade teacher3@gmail.com 09672002882 Seven S2 2022-11-01

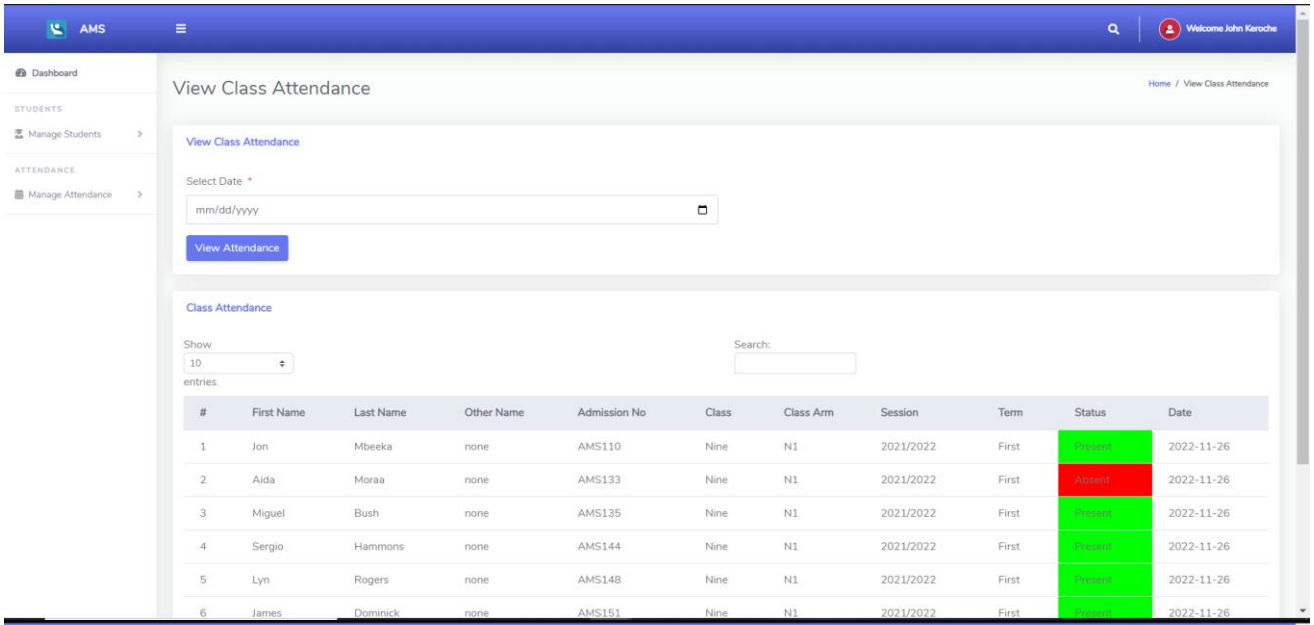

Dashboard
CLASS AND CLASS ARMS
Manage Classes
Manage Class Arms
TEACHERS
Manage Teachers
STUDENTS
Manage Students
SESSION & TERM
Manage Session & Term
Create Students
Home / Create Students
Create Students
Firstname *
Lastname *
Other Name *
Admission Number *
Select Class *
--Select Class--
Class Arm *
Save
All Student
Show
10
entries
Search:
First Name Last Name Other Name Admission No Class Class Arm Date Created Edit Delete
1 Thomas Omari none AMS005 Seven S1 2022-10-31
2 Samuel Ondieki none AMS007 Seven S1 2022-10-31
3 Milagros Oloo none AMS011 Seven S1 2022-10-31

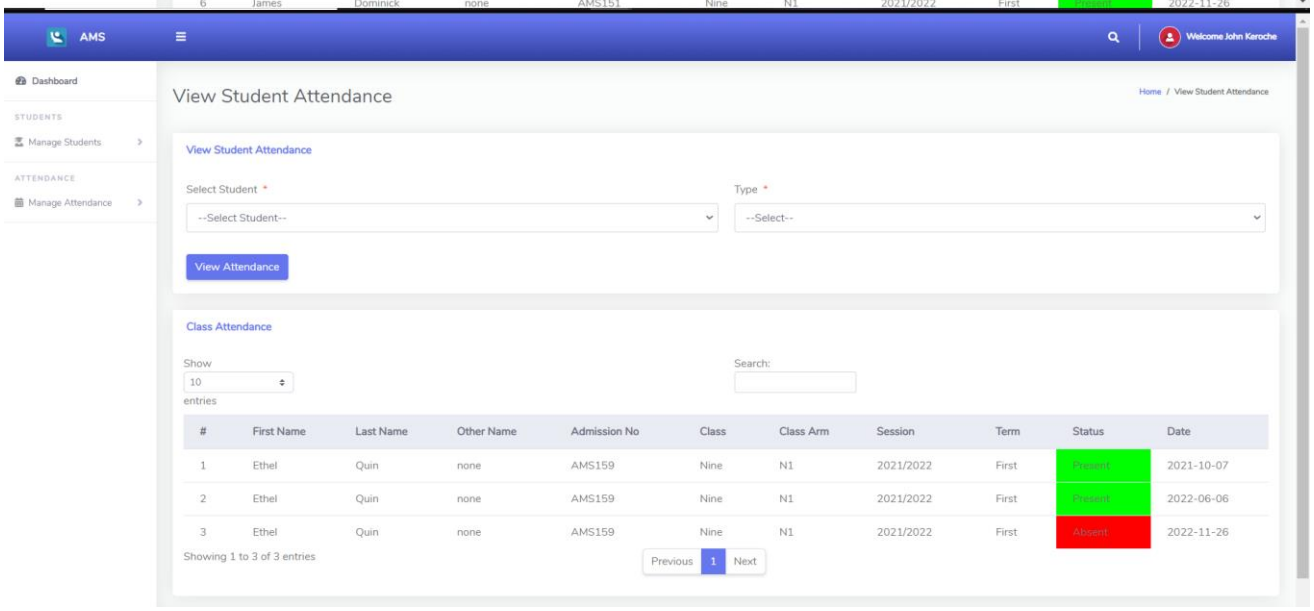

AMS
Welcome Admin
Dashboard
CLASS AND CLASS ARMS
Manage Classes
Manage Class Arms
TEACHERS
Manage Teachers
STUDENTS
Manage Students
SESSION & TERM
Manage Session & Term
Create Session and Term
Home / Create Session and Term
Create Session and Term
Session Name *
Session
Term *
--Select Tern--
Save
All Session and Term
Note: Click on the check symbol besides each to make session and term active!
Show
10
entries
Search:
Session Term Status Date Activate Edit Delete
1 2021/2022 First Active 2022-10-31
2 2021/2022 Second InActive 2022-10-31
Showing 1 to 2 of 2 entries
Previous 1 Next
© 2022


AMS
Welcome John Keroche
Dashboard
STUDENTS
Manage Students
ATTENDANCE
Manage Attendance
Class Teacher Dashboard (Nine - N1)
Home / Dashboard
STUDENTS
8
CLASSES
4
CLASS ARMS
4
TOTAL STUDENT ATTENDANCE
24
© 2022


AMS
Welcome John Keroche
Dashboard
STUDENTS
Manage Students
ATTENDANCE
Manage Attendance
All Student in (Nine - N1) Class
Home / All Student in Class
All Student In Class
Show
10
entries
Search:
First Name Last Name Other Name Admission No Class Class Arm
1 Jon Mbeeka none AMS110 Nine N1
2 Aida Moraa none AMS133 Nine N1
3 Miguel Bush none AMS135 Nine N1
4 Sergio Hammons none AMS144 Nine N1
5 Lyn Rogers none AMS148 Nine N1
6 James Dominick none AMS151 Nine N1
7 Ethel Quin none AMS159 Nine N1
8 Roland Estrada none AMS161 Nine N1
Showing 1 to 8 of 8 entries
Previous 1 Next


AMS
Welcome John Keroche
Dashboard
STUDENTS
Manage Students
ATTENDANCE
Manage Attendance
Take Attendance (Today's Date : 11-26-2022)
Home / All Student in Class
All Student in (Nine - N1) Class
Note: Click on the checkboxes besides each student to take attendance!
First Name Last Name Other Name Admission No Class Class Arm Check
1 Jon Mbeeka none AMS110 Nine N1
2 Aida Moraa none AMS133 Nine N1
3 Miguel Bush none AMS135 Nine N1
4 Sergio Hammons none AMS144 Nine N1
5 Lyn Rogers none AMS148 Nine N1
6 James Dominick none AMS151 Nine N1
7 Ethel Quin none AMS159 Nine N1
8 Roland Estrada none AMS161 Nine N1
Take Attendance


AMS
Welcome John Keroche
Dashboard
STUDENTS
Manage Students
ATTENDANCE
Manage Attendance
View Class Attendance
Home / View Class Attendance
View Class Attendance
Select Date *
mm/dd/yyyy
View Attendance
Class Attendance
Show
10
entries
Search:
First Name Last Name Other Name Admission No Class Class Arm Session Term Status Date
1 Jon Mbeeka none AMS110 Nine N1 2021/2022 First Present 2022-11-26
2 Aida Moraa none AMS133 Nine N1 2021/2022 First Absent 2022-11-26
3 Miguel Bush none AMS135 Nine N1 2021/2022 First Present 2022-11-26
4 Sergio Hammons none AMS144 Nine N1 2021/2022 First Present 2022-11-26
5 Lyn Rogers none AMS148 Nine N1 2021/2022 First Present 2022-11-26
6 James Dominick none AMS151 Nine N1 2021/2022 First Present 2022-11-26


AMS
Welcome John Keroche
Dashboard
STUDENTS
Manage Students
ATTENDANCE
Manage Attendance
View Student Attendance
Home / View Student Attendance
View Student Attendance
Select Student *
--Select Student--
Type *
--Select--
View Attendance
Class Attendance
Show
10
entries
Search:
First Name Last Name Other Name Admission No Class Class Arm Session Term Status Date
1 Ethel Quin none AMS159 Nine N1 2021/2022 First Present 2021-10-07
2 Ethel Quin none AMS159 Nine N1 2021/2022 First Present 2022-06-06
3 Ethel Quin none AMS159 Nine N1 2021/2022 First Absent 2022-11-26
Showing 1 to 3 of 3 entries
Previous 1 Next


PES
UNIVERSITY


PES
UNIVERSITY