

Net Centric Programming Project

Abstract on

Student Attendance Management System

Under the guidance of

Mrs. Nalini S

**Assistant Professor(SG), Computer Science and Engineering, Amrita School
of Engineering, Bengaluru**

Submitted by

Lakshmi R Nair

(BL.EN.U4CSE16055)



AMRITA
VISHWA VIDYAPEETHAM
U N I V E R S I T Y
Established u/s 3 of UGC Act 1956
BENGALURU CAMPUS

CONTENTS

<i>SL. No</i>	<i>Topic</i>	<i>Page. No</i>
1	Introduction	3
2	Application	3
3	Tools Used	4
4	Navigating through the implementation	6
5	Database and queries used in the implementation	11
6	Schema	15
7	Screenshots	16

INTRODUCTION

The purpose of this project is to create a constructive database management system to access and manage the student attendance management system in more meaningful way. It generates the attendance of a student on the basis of his/her presence in class. This system deals with the maintenance of student attendance details on a daily basis where the faculty will be provided with a separate username and password to update the student's attendance status. The faculty handling each particular course is responsible for taking the attendance for all the students in that class. Only if the student is present for that period, attendance will be updated else will be marked absent. In case of any ambiguity in attendance there is also a favorable option will allow the faculty the liberty to get back to the student attendance in case of absence due to an on-duty leave. After the attendance has been taken for a particular class, a consolidated sheet showing the absentees is displayed.

APPLICATIONS

- 1) Scope for a digitized attendance where the database model can provide a more effective and efficient way than the conventional file register system.
- 2) Late entry student permissions can be granted based on the time of arrival at campus and arrival to class.
- 3) An app-based implementation of student records which helps in easy access and maintenance of the same.

TOOLS USED

1) PHP

- The PHP Hypertext Preprocessor (PHP) is a programming language that allows web developers to create dynamic content that interacts with databases. PHP is basically used for developing web-based software applications.
- PHP is a server-side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.

2) CodeIgniter Framework

- CodeIgniter is a powerful PHP framework with a very small footprint, built for developers who need a simple and elegant toolkit to create full-featured web applications.
- It is an Open Source framework. It has a very rich set of functionalities, which will increase the speed of website development work.
- A website built in CodeIgniter is secure too, as it has the ability to prevent various attacks that take place through websites.

3) CSS (Cascading Style Sheet)

- Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as MathML or XHTML).
- CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.
- CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

4) HTML

- Hypertext Markup Language is the standard markup language for documents designed to be displayed in a web browser.
- It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

5) MySQL

- MySQL is a fast, easy to use relational database. It is currently the most popular open-source database. It is very commonly used in conjunction with PHP scripts to create powerful and dynamic server-side applications.
- MySQL is used for many small and big businesses. It is developed, marketed and supported by MySQL AB, a Swedish company. It is written in C and C++.

6) Ajax

- AJAX stands for Asynchronous JavaScript and XML. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS, and Java Script.
- Ajax uses XHTML for content, CSS for presentation, along with Document Object Model and JavaScript for dynamic content display.
- With AJAX, when you hit submit, JavaScript will make a request to the server, interpret the results, and update the current screen. In the purest sense, the user would never know that anything was even transmitted to the server.

7) JavaScript

- JavaScript is an open source & most popular client-side scripting language supported by all browsers. JavaScript is used mainly for enhancing the interaction of a user with the webpage.
- The main advantage of JavaScript is that all modern web browsers support JavaScript. So, you do not have to worry about whether your site visitor uses Internet Explorer, Google Chrome, Firefox or any other browser. JavaScript will be supported.

NAVIGATING THROUGH THE IMPLEMENTATION

LOGIN PAGE

- This page gives access to 2 users namely the admin and the faculty
- The users can login by validating the credentials of username and password
- Based on user validation, the appropriate user can login into the respective home page
- pages
- Username should be an Email ID and password of the admin is a secure key known secretly only to the admin. Here we have the admin password as '**admin123**' (assuming we have only one admin because if we have more than one admin the current admin can assign email id and password to the new admin by adding username and password in the database which can lead to a security breach issue). Passwords of the faculty will be in the form digits provided by admin (using md5 encryption in the database).
- Admin has the only right to create, update and delete the details of the department, students, faculty, subjects, assigned subjects etc. Faculty have only the right to take attendance of the students.

DASHBOARD (HOMEPAGE)

- We consider two cases here and they are as follows: -
 1. *Dashboard -when admin is the user*
 - ❖ A dashboard will be displayed in which total no of students, total no of faculty, total no department and total number of subjects are displayed are icons.
 - ❖ When admin clicks the "**more info**" in each icon, it directs to the respective page. For instance: -When admin clicks "**more info**" in the department icon, department management page will pop out to the left side of the dashboard which contains the menu bar.
 - ❖ Menu contains the following:

- a)* Dashboard: navigate the page to the home page or dashboard
- b)* Department management: contains details of each branch
- c)* Student management: contains details of each student
- d)* Faculty management: contains details of each teaching faculty
- e)* Subject management: contains details of each course subject taught by the faculty.
- f)* Assigned subjects: contains details of each the courses assigned to each faculty
- g)* Logout

2. Dashboard -when faculty is the user

- ❖ A dashboard will be displayed where the total number of assigned subjects is displayed as an icon.
- ❖ When the admin clicks “more info” in each icon ,it direct to the respective page on the left side of the dashboard which contains the menu bar.
- ❖ Menu contains the following:
 - a)* Home: Navigates the page to the home page or dashboard
 - b)* Reports
 - c)* Logout
- ❖ Dash board contains a table called “Assigned subjects” contain id, Department id, Department name, subject id, subject Name, subject code, overall report and adds attendance of each subjects of the teacher who logged into the home page.
- ❖ Add attendance contain a button called “add attendance” in each row which navigates to the page to add attendance form.

- ❖ Overall report contains a button called “overall report” in each row which navigate the page to the page called “**overall report**”.

DEPARTMENT MANAGEMENT

- Details are in the form of table which consist of id, Department Name, Department code, Action.
- Action allows the user (admin) to delete the particular department.
- There is a search bar option on right side top of the page and a button called “Add department” which navigate the page into another page called Add department.

ADD DEPARTMENT

- Page consist of two text fields called Department name and Department code.
- When admin clicks the “**submit**” button whatever the details admin gives will be reflected in the “**Department management**” table. Hence this page will navigate to the page called **Department management**.

STUDENT MANAGEMENT

- There is a button called “**Add student**” in the right side of page navigate the page into Add student page where we can enter the details of the students.
- There is another form called “**Search student**” which helps to retrieve the students in the specified department and semester, following which it will be displayed in the student list table.
- Form consists of two lists called “**Department Name**” and “**Semester**”. After entering the details when admin clicks the submit button, details of the students will be shown in the students list
- Student list is in the form of table consist of id (row id, which should not be confused with id of the student), Student name, Roll number, created date and Action (allows the admin to delete the details of the particular student) .

ADD STUDENT

- A form which consist of 2 lists (Department name and semester) and 2 text fields (student name and roll number).
- When the admin clicks “submit” button, after entering the details, it will navigate to the page called “**Student management**”.

FACULTY MANAGEMENT

- Page consists of three text fields (Faculty email, Faculty name and Faculty code). Faculty code is in the form of an integer which will be created using md5 encryption. So, we need not give a password explicitly.
- When admin click the “**submit**” button whatever the details admin give will be reflected in the Faculty management table.so this page will navigate to the page called Faculty management.

SUBJECT MANAGEMENT

- Details are in the form of table which consist of id, Department, Semester, subject name, subject code and Action.
- There is a search bar option is there in right side top of the page and a button called “**Add Subject**” which navigates from one page to another page.

ADD SUBJECT

- This page consists of 2 text fields (Department name and Department code).
- When admin click the “**submit**” button whatever the details admin give will be reflected in the subject management table and this page will navigate to the page called “**Subject Management**”.

ASSIGNED SUBJECT

- Details are in the form of table which consist of id, department id, department name, subject id, subject name, subject code, faculty id, faculty email, faculty name and action (allows the user(admin) to delete the details of particular row).
- There is a search bar option on the top right-hand corner of the page. “**Assign subject**” navigates the page to another page called “**Add subjects**” (different from previous one).

ADD SUBJECT (TO ASSIGN SUBJECTS)

- This page consists of 3 lists (select Department, Teacher name and select subject).
- When admin clicks the “**submit**” button, all the details that the admin gives will be reflected in the “**Assigned subjects table**” and it will navigate to the page called “**Assigned subjects**”.

ATTENDANCE AND REPORTS

- It is a form contain 4 fields (Department, semester, subject and Teacher).
- Date is designed in the form of a calendar where Year, month and date can be set into mm/dd/yyyy format. Hour list contain 6 hours from 1 to 6.
- When Teacher presses “**submit**” button a table called students list will pop out. Students list displays the students details in the class, we specified before. Student list Table consist of (represent number of id not roll number), Roll number, student.
- Label contains a list that makes the users to mark attendance details (present/absent/on duty). Database is written in such a way where we consider all students marked as “**On duty**” is considered to be “**present**”.

- After marking attendance, if the user clicks the button called “**submit**” which navigate the current page to another page called “**reports**” which is in the form of a table.

DATABASE AND QUERIES USED IN IMPLEMENTATION

- This project contains the main database called “**Attendance**”.
- The “**Attendance**” database contains 9 tables in total. We have used foreign keys in almost every table and this makes the programmer’s task of writing queries even simpler and we have not used join queries.
- Each table in turn contains all the attributes and the queries used to populate each table is given below.
- For each table we have mentioned attribute names, variable data type and representation.

1. login_user

- *id*: int, unique value, primary key of the table.
- *username*: varchar, username of the user.
- *password*: varchar, password of the user (encrypted version, encrypted using md5 encryption).
- *status*: int, whether that person is currently active (1) or not (0).
- *user_status*: int, represent what rights are given to the user Admin who has the right to create, update, insert and delete the data from the database.
- *created_data*: timestamp, when that details created (here details about the user).

2. Semester

- *id*: int, primary key (unique value which identifies each row separately).
- *Semester code*: int, represent the semester 1-8.
- *Semester name*: varchar, represent which semester in string format Semester 1, Semester 2 etc.
- *created _date*: timestamp, when the details are created.

3. Branch

- *id*: int, primary key (unique value which identifies each row separately).
- *branch_name*: varchar, name of the branch.
- *branch_code*: varchar, code of the branches.
- *status*: int, to find if the given branch is currently active in the front end or not. If active 1 else it is 0 (note: if user deletes the details of the particular branch it will remain in the database but status will be different).
- *created _date*: timestamp, when the details are created.

4. Subjects

- *id*: int, primary key (unique value which identifies each row separately).
- *branch_id*: int, foreign key (primary key from the table branch).
- *sem_id*: int, foreign key (primary key from the table semester).
- *sub_name*: varchar, name of the subject .
- *sub_code*: varchar, code of the subject.
- *status*: int, the given details of subject is currently active in the front end or not (if active 1 else 0).
- *created _date*: timestamp, when the details are created.

5. Student

- *id*: int, primary key (unique value which identifies each row separately).
- *branch_id*: int, foreign key (primary key from the table branch)
- *sem_id*: int, foreign key (primary key from the table semester).
- *student_name*: varchar, name of the student..
- *roll_number*: varchar, registration number of the student.
- *status*: int, the given details of student is currently active in the front end or not (if active 1 else 0).
- *created_date*: timestamp ,when the details are created.

6. Teachers

- *id*: int, primary key (unique value which identifies each row separately).
- *teachers_email*: varchar email of the teacher (serves as the username in the login_user table).
- *teacher_name*: varchar, name of the teacher.
- *teacher_code*: varchar, code given to each teacher (serves as the password to login...).
- *status*: int, the given details of teacher is currently active in the front end or not (if active 1 else 0).
- *created_date*: timestamp, when the details are created.

7. Assigned_subjects

- *id*: int, primary key (unique value which identifies each row separately).
- *department_id*: int, foreign key (primary key from the table branch).
- *subject_id*: int, foreign key (primary key from the table subject).

- *teacher_id*: int, foreign key (primary key from the teacher table).
- *status*: int, the given details of student is currently active in the front end or not (if active 1 else 0).
- *created _date*: timestamp, when the details are created.

8. Attendance_posts

- *id*: int, primary key (unique value which identifies each row separately).
- *department_id*: int, foreign key (primary key from the table branch).
- *semester_id*: int, foreign key (primary key from the table semester).
- *subject_id*: int, foreign key (primary key from the table subject).
- *teacher_id*: int, foreign key (primary key from the teacher table).
- *student_id*: int, foreign key (primary key from the student table).
- *attendance_value*: int, 1 for present and on duty, 2 for absent.
- *created _date*: timestamp, when the details are created.

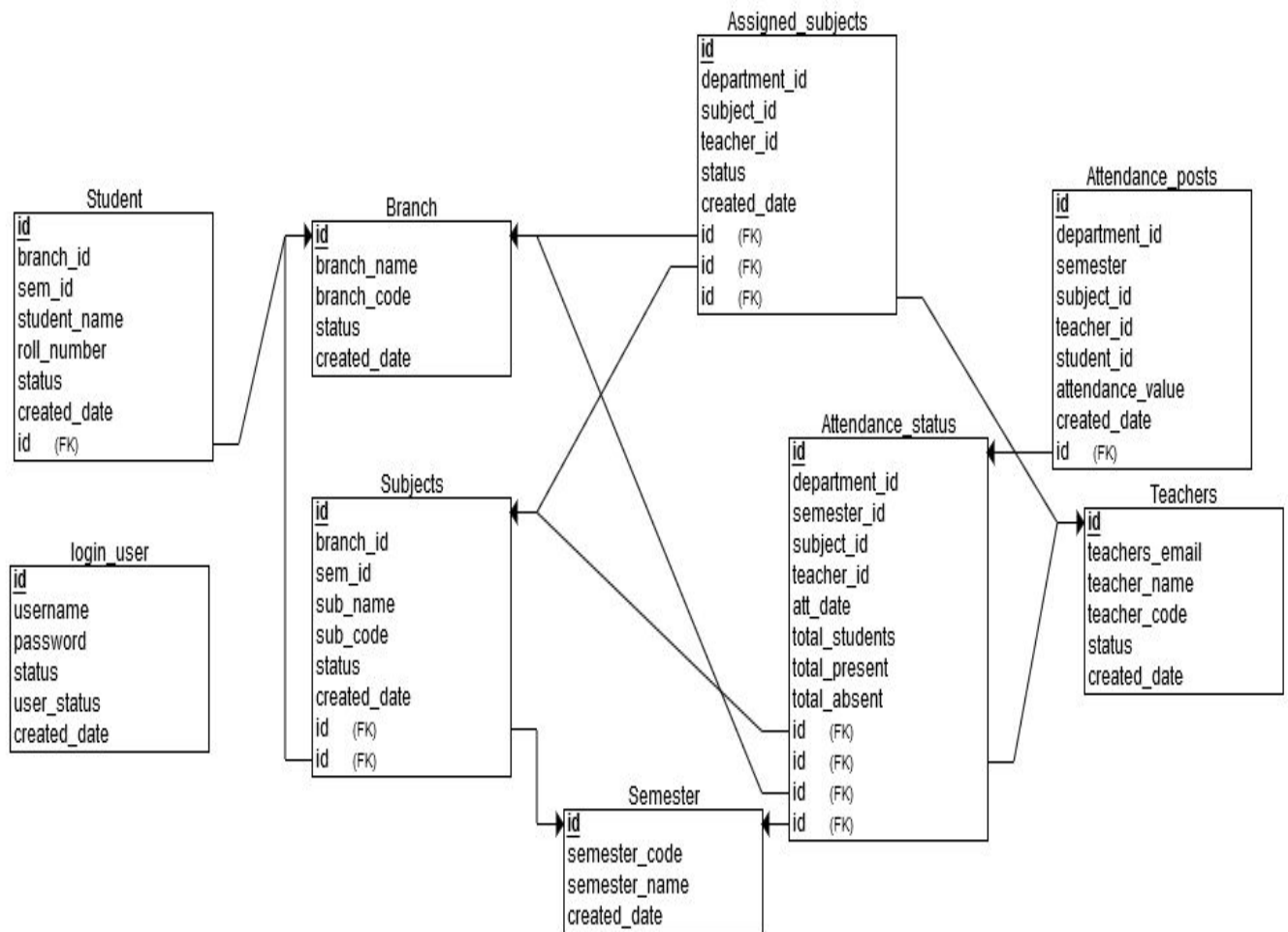
9. Attendance_status

- *id*: int, primary key (unique value which identifies each row separately).
- *department_id*: int, foreign key (primary key from the table branch).

- *semester_id*: int, foreign key (primary key from the table semester).
- *subject_id*: int, foreign key (primary key from the table subject).
- *teacher_id*: int, foreign key (primary key from the teacher table).
- *att_date*: varchar, on which date the teacher took the attendance.
- *total_students*: int, total students of the class.
- *total_present*: int, total no. of students present.
- *Total_absent*: int, total no. of students absent.

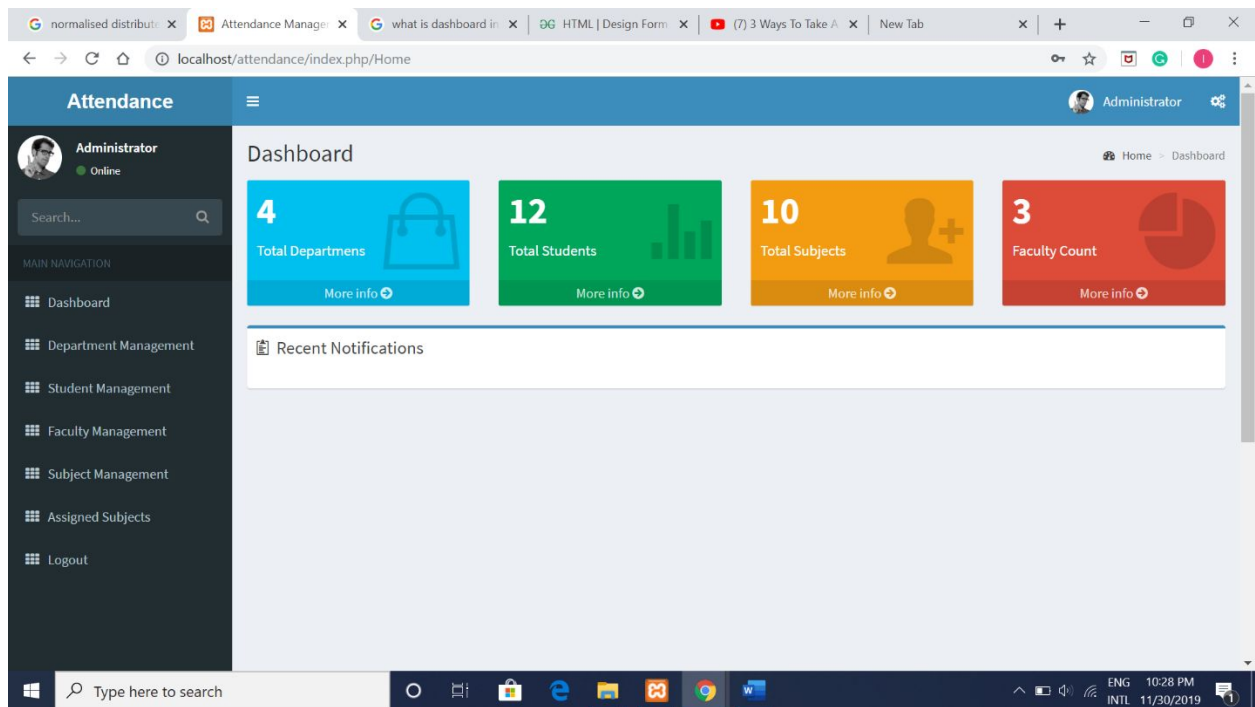
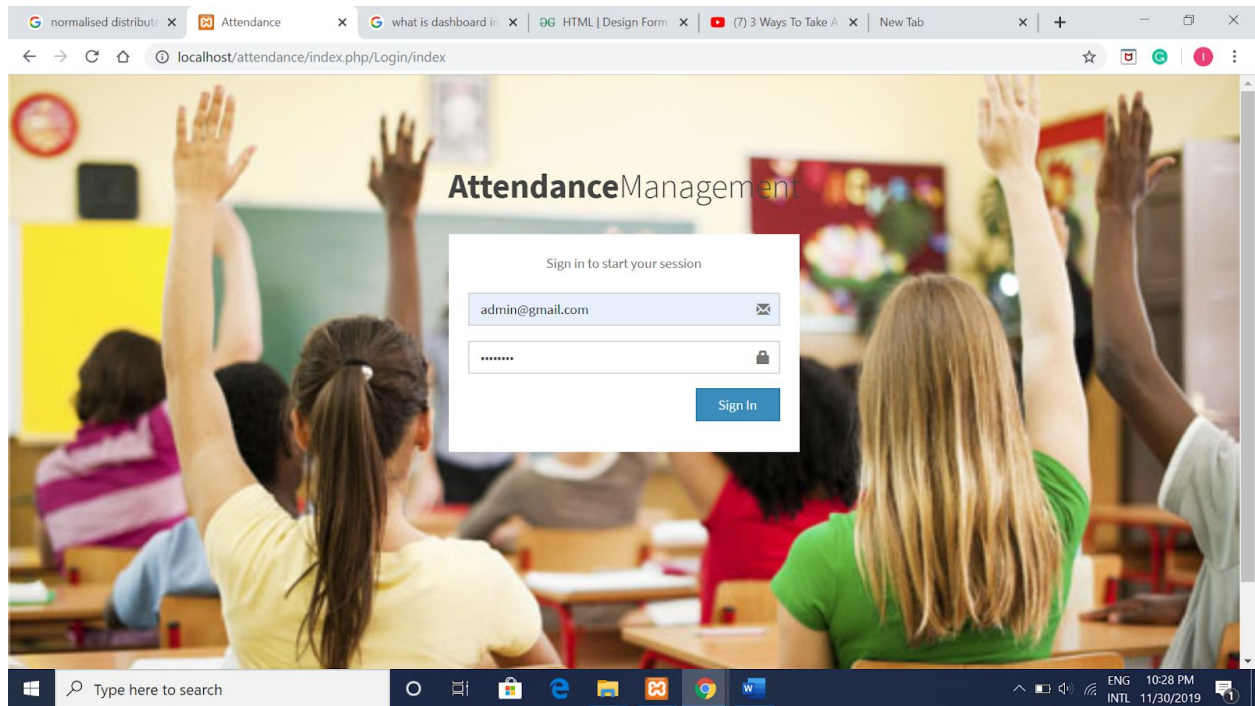
SCHEMA

Attendance Management



- Here we have a schema representation of the student attendance system. The schema contains all the 9 tables that has been used in the “**Attendance**” database.
- Each box represents a table with its corresponding attributes. An attribute which is either a primary key or a foreign key is represented differently.
- FK- It represents the foreign keys
- Id – An attribute which is underlined represented the primary key of that table.

SCREENSHOTS (User-Admin)



normalised distribute x Attendance Manager x what is dashboard in x OG HTML | Design Form x (7) 3 Ways To Take A x New Tab

localhost/attendance/index.php/Departments/list

Attendance

Administrator Online

Search...

MAIN NAVIGATION

- Dashboard
- Department Management
- Student Management
- Faculty Management
- Subject Management
- Assigned Subjects
- Logout

Department Management

Home > Department Management > list

Department List

Show 10 entries Search:

Id	Department Name	Department Code	Action
1	Mechanical Engineering	me	Delete
2	Computer Science And Engineering	CSE	Delete
3	Electronics and communication Engineering	ECE	Delete
4	Aerospace Engineering	ASE	Delete

Showing 1 to 4 of 4 entries

Previous 1 Next

Type here to search

ENG 10:28 PM
INTL 11/30/2019

normalised distribute x Attendance Manager x what is dashboard in x OG HTML | Design Form x (7) 3 Ways To Take A x New Tab

localhost/attendance/index.php/StudentManagement/list

Attendance

Administrator Online

Search...

MAIN NAVIGATION

- Dashboard
- Department Management
- Student Management
- Faculty Management
- Subject Management
- Assigned Subjects
- Logout

Student Management

Home > Student Management > list

Search Students

Department name
Computer Science And Engineering

Semester
Semester 7

Submit

Student List

Show 10 entries Search:

Id	Student Name	Roll Number	Created Date	Action
No data available in table				

Type here to search

ENG 10:30 PM
INTL 11/30/2019

normalised distribut... Attendance Manager x what is dashboard in x OG HTML | Design Form x (7) 3 Ways To Take A x New Tab

localhost/attendance/index.php/FacultyManagement/list

Attendance

Administrator Online

Search...

MAIN NAVIGATION

- Dashboard
- Department Management
- Student Management
- Faculty Management
- Subject Management
- Assigned Subjects
- Logout

Faculty Management

Home > Faculty Management > list

Faculty List

Show 10 entries Search:

Id	Faculty Email	Faculty Name	Faculty Code	Action
1	lakshmirajna1@gmail.com	Lakshmi R	1570025459	Delete
2	nalinis@gmail.com	Nalini S	1570025472	Delete
3	suja_p@gmail.com	suja p	1575124220	Delete
4	priyank_vivek@gmail.com	Priyanka Vivek	1575133285	Delete

Showing 1 to 4 of 4 entries

Previous 1 Next

Type here to search

ENG 10:32 PM 11/30/2019

normalised distributed vect... Attendance Management x OG HTML | Design Form - Geel... x (7) 3 Ways To Take A Screen... x New Tab

localhost/attendance/index.php/SubjectManagement/list

Subject Management

Home > Subject Management > list

Subject List

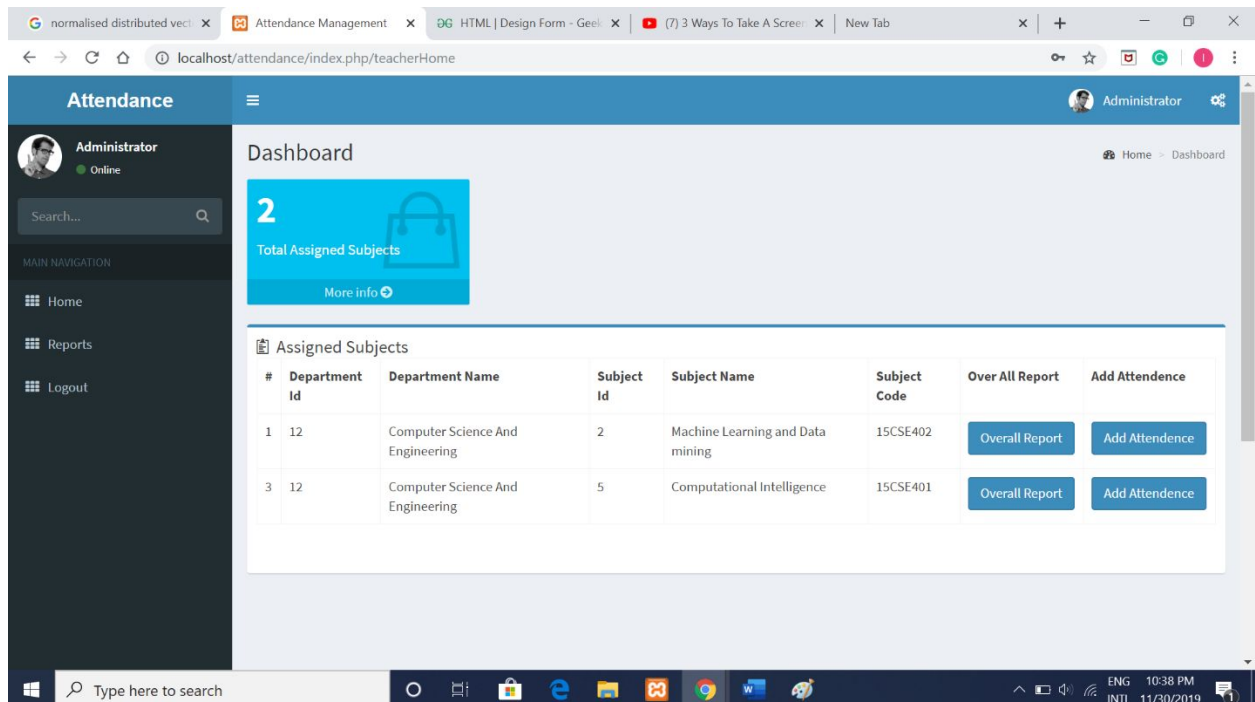
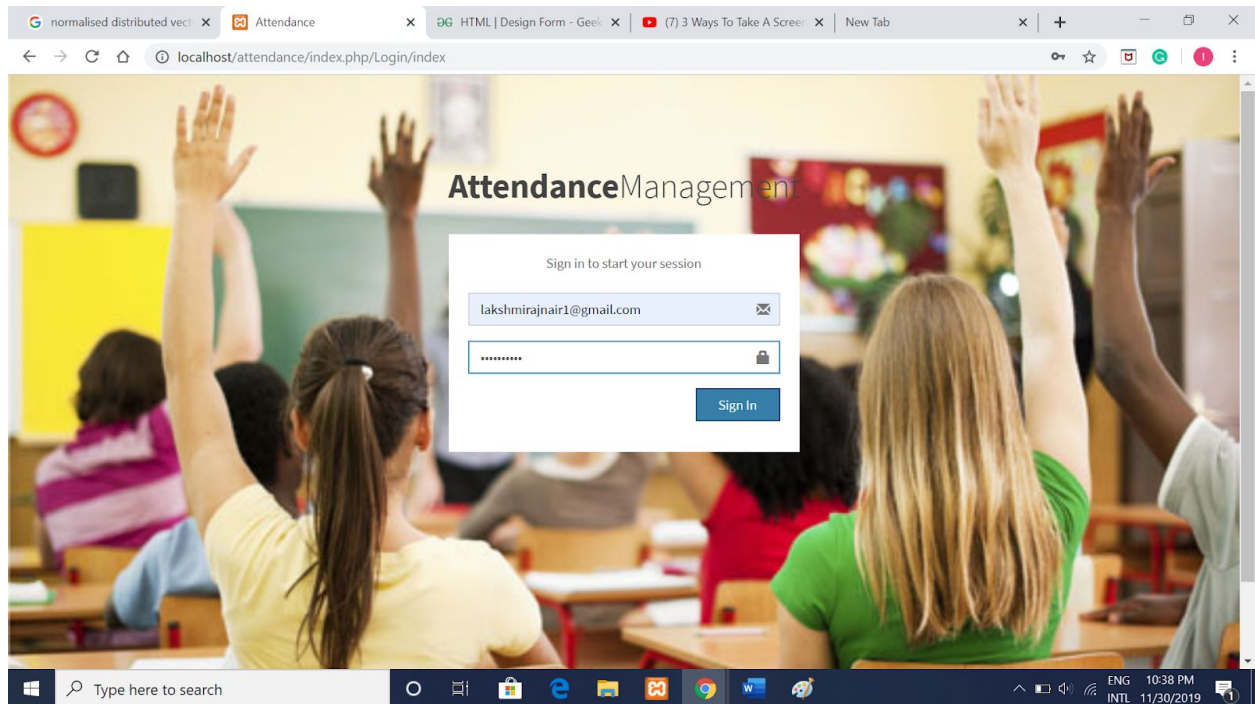
Show 10 entries Search:

Id	Department	Semester	Subject Name	Subject Code	Action
1	Computer Science And Engineering	Semester 7	Netcentric Programming	15CSE420	Delete
2	Computer Science And Engineering	Semester 7	Machine Learning and Data mining	15CSE402	Delete
3	Mechanical Engineering	Semester 1	Fluid dynamics	17MEE202	Delete
4	Computer Science And Engineering	Semester 7	Structures Interpretation of Computer Programs	15CSE400	Delete
5	Computer Science And Engineering	Semester 7	Computational Intelligence	15CSE401	Delete
6	Computer Science And Engineering	Semester 7	Computer Vision	15CSE403	Delete
7	Computer Science And Engineering	Semester 7	Natural Language Processing	15CSE404	Delete
8	Computer Science And Engineering	Semester 7	Information Security	15CSE405	Delete
9	Computer Science And Engineering	Semester 7	Machine Learning and Data Mining LAB	15CSE406	Delete

Type here to search

ENG 10:34 PM 11/30/2019

SCREENSHOTS (User-Faculty)



normalised distributed veci x Attendance Management x OG HTML | Design Form - Geel x (7) 3 Ways To Take A Screen x New Tab x

localhost/attendance/index.php/AttendanceManagement/addAttendance/1

Administrator Online

Search...

MAIN NAVIGATION

- Home
- Reports
- Logout

Add Attendance

Home > Add Attendance

Department

Computer Science And Engineering

Subject Name

Machine Learning and Data mining

Subject Code

15CSE402

Semester

7

Date

11/30/2019

Hour

2

Submit

Type here to search

ENG 10:39 PM
INTL 11/30/2019

normalised distributed veci x Attendance Management x OG HTML | Design Form - Geel x (7) 3 Ways To Take A Screen x New Tab x

localhost/attendance/index.php/AttendanceManagement/addAttendancePost

#	Roll Number	Student Name	Label
1	BL.EN.U4CSE16055	Lakshmi R Nair	Present
5	BL.EN.U4CSE16054	vivek	On Duty
6	BL.EN.U4CSE16058	Lavanya Karthikeyan	Present
7	BL.EN.U4CSE16001	Aakash R	On Duty
8	BL.EN.U4CSE16064	Malavika R Nair	Present
10	BL.EN.U4CSE16002	Abhik	Absent
11	BL.EN.U4CSE16003	Abhijna	Absent
12	BL.EN.U4MEE16004	Abhirami	Present
13	BL.EN.U4CSE16005	Harshith	Absent
14	BL.EN.U4CSE16014	Arvind Sudheer	Present

Submit

Type here to search

ENG 10:40 PM
INTL 11/30/2019

normalised distributed vec | Attendance Management | HTML | Design Form - Geel | (7) 3 Ways To Take A Screen | New Tab

localhost/attendance/index.php/Reports/subjectWiseReport/1

Subject Wise Report

Home > Reports

Add Department

#	Department Id	Department Name	Subject Name	Subject Code	Semester
# 12		Computer Science And Engineering	Machine Learning and Data mining	15CSE402	Semester 7

Show 10 entries

Search:

Id	Student Name	Roll Number	Total Hours Class Took	Hours Present	hours Absent	Percent Present
1	Lakshmi R Nair	BL.EN.U4CSE16055	8	7	1	87.5
2	vivek	BL.EN.U4CSE16054	8	6	1	75
3	Lavanya Karthikeyan	BL.EN.U4CSE16058	8	6	1	75
4	Aakash R	BL.EN.U4CSE16001	8	4	2	50
5	Malavika R Nair	BL.EN.U4CSE16064	8	4	1	50
6	Abhik	BL.EN.U4CSE16002	8	4	1	50
7	Abhijna	BL.EN.U4CSE16003	8	4	1	50
8	Abhirami	BL.EN.U4MEE16004	8	5	0	62.5

Type here to search

ENG 10:41 PM
INTL 11/30/2019

normalised distributed vec | Attendance Management | HTML | Design Form - Geel | (7) 3 Ways To Take A Screen | New Tab

localhost/attendance/index.php/Reports

Reports

Home > Reports

Show 10 entries

Search:

Id	Department Name	Semester	Subject Name	Total Students	Total Present	Total Absent	Attendance Date	Action
1	Computer Science And Engineering	Semester 7	Machine Learning and Data mining	2	1	1	2019-10-02	Report
2	Computer Science And Engineering	Semester 7	Machine Learning and Data mining	4	3	1	2019-10-16	Report
3	Computer Science And Engineering	Semester 7	Machine Learning and Data mining	5	4	1	2019-10-16	Report
4	Computer Science And Engineering	Semester 7	Machine Learning and Data mining	10	9	1	2019-11-26	Report
5	Computer Science And Engineering	Semester 7	Machine Learning and Data mining	10	7	3	2019-11-25	Report
6	Computer Science And Engineering	Semester 7	Machine Learning and Data mining	10	10	0	2019-11-25	Report
8	Computer Science And Engineering	Semester 7	Computational Intelligence	10	7	3	2019-11-30	Report
9	Computer Science And Engineering	Semester 7	Machine Learning and Data mining	10	10	0	2019-11-30	Report

Type here to search

ENG 10:40 PM
INTL 11/30/2019