Report

1. Introduction

An automated attendance system that consists of a web system for entire organization to record attendance. The web application is to be used by students, faculty, and admin personnel.

Role of Admin

- The role of the admin is to add new student and faculty by entering his personal details and provides the faculty/student with userID and password so that he/she can access the application.
- Admin can check the attendance of the particular student by entering the roll number and generate attendance reports.
- The admin may check all attendance data, press a button to see defaulter list, search for particular student attendance by name, search class attendance and generate excel reports.

Role of Faculty

- Each Faculty has his/her own login. On login, the class faculty can see a list of students or members registered under it.
- The faculty may take the attendance and mark the present students using checkbox provided in front of every student name. This attendance sheet is stored and sent to the central administrator of the organization and stored there.

Role of Student

- The student logs in and checks his/her attendance statistics for different courses/faculties.
- The student also has the option to register for course available and once he registers for the course, his name will appear for attendance whenever a faculty logs in to record the attendance.

2. Software Architecture

First of all,we have *sql_login* info from where database will access,and create a database named as *records*;and then as we will run *automate/automate.php* it will generate all basic require tables.

We have some following tables in *records*: - ***admin_info, student_info, faculty_info***: where we are storing *Username* , *Password* is there. - ***students***: where we are storing *Username* , *Name* , *DOB*, *No. of courses* , *Branch* and *Batch* is there. - ***courses***: where we are storing *Course Code* , *Course Name* , *No. of Students* , *Branch* and *Batch* is there. - ***(username)m, (username)present, (username)absent***: where for a particular students' details like *present, absent* and dates of *present, absent* will be stored when faculty will mark their Attendance. - ***present, absent***: will store username of students which are *present* or *absent* on a particular date. - ***course_attendance***: will store *date* and on that day *no. of students present and absent* - ***course***: will store *username* of that registered course.

Added Logout successfully in proper way using *session_destroy()*.While converting data table to *Excel Sheet* we have use some php libraries which will basically copy data from data table and form a new excel sheet where ie will copy this data.

3. Solution/Implementation

As admin will add new student or faculty, it will store their data in their particular table. To check attendace it will take data from their *present,absent* tables and will display that data and there is also an option to check attendance in a particular range of date, and to convert that table data into *Excel* we have used some php libraries and implemented code with help of *stackoverflow*.

For *login*, *SQL Injection* check is properly done and will use data from their particular table where information is stored by *admin*.

When faculty will add some course it will create*course* table and courses with their details will display to student who is qualify to register, similarily when faculty will mark attendance it will update their require tables. When faculty wnats to veiw attendance, it will use all attendance related table and will dislay in a formated way.

Student will be shown their related courses of their particular*Batch* and *Branch* with their *course code, course name and faculty name* when student has to veiw their attendance it will access*(username)m* table form which data is extracted, and will display dates from *(username)present,(username)absent* within a formated way.

4. Results

We learned many important php libraries, HTML CSS cascading and many more.. Preferred Technologies: ***PHP, MySQL, HTML, Javascript, Git.***