# Department of Computing

**CS344: Web Engineering**

**Class: BESE – 12A**

# Lab 9: Attendance Recorder

**Date: 30th Nov 2023**

# Time: 02:00 PM – 04:50 PM

# Instructor: Dr. Fahad Ahmed Satti

# Lab Engineer: Ms. Ayesha Asif

# Lab 9: Attendance Recorder

## Description

In this lab, you have to design and implement an application to maintain attendance for students at NUST. This application has two primary use cases:

1. Teachers can take attendance
2. Students can look up their attendance

In both cases there should be one master view for your application which all the pages will inherit. The application should authenticate the user first and based on their roles should open up their page only. Once authenticated, the teachers should be shown a view with the current attendance session and a list of all previous and upcoming sessions. The teacher should be able to mark attendance in any session for any student. The students should be able to view all of their attendance, only. In case any of their attendance is below 75%, it should be bolded and shown in red. In case the attendance is below 85%, it should be shown in yellow. Otherwise you can show the attendance in green.

You can use the following skeleton mysql db structure for recording and fetching records from the database.

--

-- Table structure for table `attendance`

--

CREATE TABLE IF NOT EXISTS `attendance` (

`classid` int(50) NOT NULL,

`studentid` int(50) NOT NULL,

`isPresent` tinyint(1) NOT NULL,

`comments` varchar(200) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `class`

--

CREATE TABLE IF NOT EXISTS `class` (

`id` int(50) NOT NULL,

// TODO: NAME OF SUBJECT

`teacherid` int(50) NOT NULL,

`starttime` time NOT NULL,

`endtime` time NOT NULL,

`credit\_hours` int(11) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `user`

--

CREATE TABLE IF NOT EXISTS `user` (

`id` int(50) NOT NULL AUTO\_INCREMENT,

`fullname` varchar(200) NOT NULL,

`email` varchar(200) NOT NULL,

`class` varchar(10) NOT NULL,

`role` enum('teacher','student','admin') NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO\_INCREMENT=1 ;

## Objectives

* Implement the attendance recorder application
* Authenticate the user
* Allow the teacher to view all previous, next and the current session.
* Allow the teacher to mark the attendance in any session.
* Allow the student to view all of his attendance, only
* Color code the attendance.
* Show a nice interface to the user, with a single master view.

## Tools/Software Requirement

* Solutions should be made using HTML5, CSS3, JavaScript, PHP and MySQL

## Pitfalls

* Any exceptions or errors leading to non-execution of submitted code.
* Failure to upload the solution on LMS.
* Failure to explain the submission, during viva.

## Deliverables

* Convert your files to a zip folder and name it as given below and upload the zip folder to LMS.
  + Name – Registration No. – Section
* This lab is graded. Min marks: 0. Max marks: 10.

**Lab Rubrics:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Item** | **Clo** | **Plo** | **Marks** | | | | | |
| **0** | **1** | **2** | **3** | **4** | **5** |
| R1 | Concepts Related to WWW | CLO-1 | PLO-1 | The student is absent in the lab/ The submission is plagiarised. | The student is unable to understand the given problem within the context of WWW and does not select the relevant method to solve it. | Inbetween | The student requires some guidance to  understand the problem, to select relevant method, and to develop appropriate web driven program flow. | Inbetween | The student fully understands the given problem in context of WWW, is able to select the relevant method to solve it. |
| R3 | Solution Accuracy | CLO-4 | PLO-3 | The student is absent in the lab/ The submission is plagiarised. | The student is unable to produce any part of the solution, with accurate results. | Inbetween | The student is able to produce accurate results for some parts of the solution. | Inbetween | The student is able to produce accurate results for the complete solution. |
| R5 | Team Response | CLO-5 | PLO-9 | The student is absent. | Student shows a lack of enthusiasm and willingness to contribute to the team's efforts, often disengaging or avoiding teamwork. | Student's contributions to the team are inconsistent, and they may hesitate to actively participate or initiate collaborative efforts. | Student generally participates willingly within the team, showing a proactive approach to collaborative work. | Student is an enthusiastic and active member of the web development team, actively participating, supporting teammates, and showing a positive attitude. | Student is a passionate and influential contributor to the web development team, actively inspiring and motivating teammates to achieve exceptional results. |

CREATE DATABASE attendancedb;

USE attendancedb;

CREATE TABLE student (

sid INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100),

email VARCHAR(100) UNIQUE,

password VARCHAR(100),

rollNo VARCHAR(20)

);

CREATE TABLE teacher (

tid INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100),

email VARCHAR(100) UNIQUE,

password VARCHAR(100)

);

CREATE TABLE course (

cid INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100),

tid INT,

FOREIGN KEY (tid) REFERENCES teacher(tid)

);

CREATE TABLE enrollments (

eid INT PRIMARY KEY AUTO\_INCREMENT,

sid INT,

cid INT,

FOREIGN KEY (sid) REFERENCES student(sid),

FOREIGN KEY (cid) REFERENCES course(cid)

);

CREATE TABLE classAttendance (

aid INT PRIMARY KEY AUTO\_INCREMENT,

sid INT,

tid INT,

cid INT,

startTime DATETIME,

endTime DATETIME,

attendance\_status ENUM('present', 'absent'),

FOREIGN KEY (sid) REFERENCES student(sid),

FOREIGN KEY (tid) REFERENCES teacher(tid),

FOREIGN KEY (cid) REFERENCES course(cid)

);

USE attendancedb;

-- Inserting data into the 'student' table

INSERT INTO student (name, email, password, rollNo) VALUES

('John Doe', 'john@example.com', 'password123', 'A001'),

('Alice Smith', 'alice@example.com', 'pass456', 'B002'),

('Emma Johnson', 'emma@example.com', 'secure789', 'C003'),

('Michael Brown', 'michael@example.com', 'pass123', 'D004'),

('Sophia Wilson', 'sophia@example.com', 'strongpass', 'E005');

-- Inserting data into the 'teacher' table

INSERT INTO teacher (name, email, password) VALUES

('Professor Smith', 'prof.smith@example.com', 'teacherpass'),

('Dr. Johnson', 'dr.johnson@example.com', 'secureteacher'),

('Ms. Davis', 'davis@example.com', 'teacher123'),

('Mr. Lee', 'lee@example.com', 'passforteacher'),

('Mrs. Adams', 'adams@example.com', 'teacherpass123');

-- Inserting data into the 'course' table

INSERT INTO course (name, tid) VALUES

('Mathematics', 1),

('Physics', 2),

('Literature', 3),

('History', 4),

('Biology', 5);

-- Inserting data into the 'enrollments' table

INSERT INTO enrollments (sid, cid) VALUES

(1, 1),

(2, 2),

(3, 3),

(4, 4),

(5, 5);

-- Inserting data into the 'classAttendance' table

INSERT INTO classAttendance (sid, tid, cid, startTime, endTime, attendance\_status) VALUES

(1, 1, 1, '2023-11-30 09:00:00', '2023-11-30 11:00:00', 'present'),

(2, 2, 2, '2023-11-30 10:00:00', '2023-11-30 12:00:00', 'present'),

(3, 3, 3, '2023-11-30 11:00:00', '2023-11-30 13:00:00', 'absent'),

(4, 4, 4, '2023-11-30 12:00:00', '2023-11-30 14:00:00', 'present'),

(5, 5, 5, '2023-11-30 13:00:00', '2023-11-30 15:00:00', 'absent');

-- Inserting additional data into the 'enrollments' table

INSERT INTO enrollments (sid, cid) VALUES

(1, 2),

(2, 3),

(3, 4),

(4, 5),

(5, 1),

(1, 3),

(2, 4),

(3, 5),

(4, 1),

(5, 2);

-- Inserting additional data into the 'classAttendance' table

INSERT INTO classAttendance (sid, tid, cid, startTime, endTime, attendance\_status) VALUES

(1, 1, 2, '2023-11-30 09:00:00', '2023-11-30 11:00:00', 'present'),

(2, 2, 3, '2023-11-30 10:00:00', '2023-11-30 12:00:00', 'absent'),

(3, 3, 4, '2023-11-30 11:00:00', '2023-11-30 13:00:00', 'present'),

(4, 4, 5, '2023-11-30 12:00:00', '2023-11-30 14:00:00', 'present'),

(5, 5, 1, '2023-11-30 13:00:00', '2023-11-30 15:00:00', 'absent'),

(1, 2, 3, '2023-11-30 14:00:00', '2023-11-30 16:00:00', 'present'),

(2, 3, 4, '2023-11-30 15:00:00', '2023-11-30 17:00:00', 'present'),

(3, 4, 5, '2023-11-30 16:00:00', '2023-11-30 18:00:00', 'absent'),

(4, 5, 1, '2023-11-30 17:00:00', '2023-11-30 19:00:00', 'present'),

(5, 1, 2, '2023-11-30 18:00:00', '2023-11-30 20:00:00', 'present');