# Department of Computing

**CS344: Web Engineering**

**Class: BESE – 12A**

# Lab 10: Attendance Recorder

**Date: 07 Dec 2023**

# Time: 02:00 AM - 04:50 PM

# Instructor: Dr. Fahad Satti

# Lab Engineer: Ms. Ayesha Asif

NAME Muhammad Momin Rauf 366523

Name Muhammad Waseem

# Lab 10: 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.

SCHEMA

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

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

-- Schema new\_schema1

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

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

-- Schema attendance

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

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

-- Schema attendance

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

CREATE SCHEMA IF NOT EXISTS attendance DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci ;

USE attendance ;

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

-- Table attendance.class

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

CREATE TABLE IF NOT EXISTS attendance.class (

id INT NOT NULL,

teacherid INT NOT NULL,

starttime TIME NOT NULL,

endtime TIME NOT NULL,

credit\_hours INT NOT NULL,

PRIMARY KEY (id))

ENGINE = InnoDB

DEFAULT CHARACTER SET = latin1;

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

-- Table attendance.attendance

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

CREATE TABLE IF NOT EXISTS attendance.attendance (

studentid INT NOT NULL,

isPresent BINARY(1) NOT NULL DEFAULT TRUE,

comments VARCHAR(200) NOT NULL,

class\_id INT NOT NULL,

date\_marked\_for DATE NULL,

CONSTRAINT fk\_attendance\_class

FOREIGN KEY (class\_id)

REFERENCES attendance.class (id)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = latin1;

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

-- Table attendance.user

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

CREATE TABLE IF NOT EXISTS attendance.user (

id INT 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 CHARACTER SET = latin1;

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

-- Table attendance.session

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

CREATE TABLE IF NOT EXISTS attendance.session (

studentid INT NOT NULL,

isPresent BINARY(1) NOT NULL DEFAULT TRUE,

comments VARCHAR(200) NOT NULL,

class\_id INT NOT NULL,

date\_marked\_for DATE NULL,

CONSTRAINT fk\_attendance\_class0

FOREIGN KEY (class\_id)

REFERENCES attendance.class (id)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = latin1;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

THIS IS HOW I ACHIEVED AUTHENTICATION   
TOOLS USED  
LARAVEL BREEZE

- Created a database and set up the environment.

- Installed Laravel Breeze.

- Created tables using migration.

- Ran `php artisan serve`.

- Installed npm packages.

- Generated a seeder: `php artisan make:seeder UsersTableSeeder`.

- Added `use DB;` in the seeder file.

- Used factories for fake data.

- Navigated to the main database seeder.

- Created controllers for agents and admins.

- Generated views and imported controllers into routes.

- Ran `php artisan make:Controller {Controller Name}`.

- Implemented role-based login in the auth controller.

- Fresh migration with seeding: `php artisan migrate:fresh --seed`.

- Created middleware for authentication: `php artisan make:middleware EnsureTokenIsValid`.

- Registered middleware in `kernel.php`.

- Added middleware in `web.php`.

- Modified `auth/login.blade.php` for name/phone login.

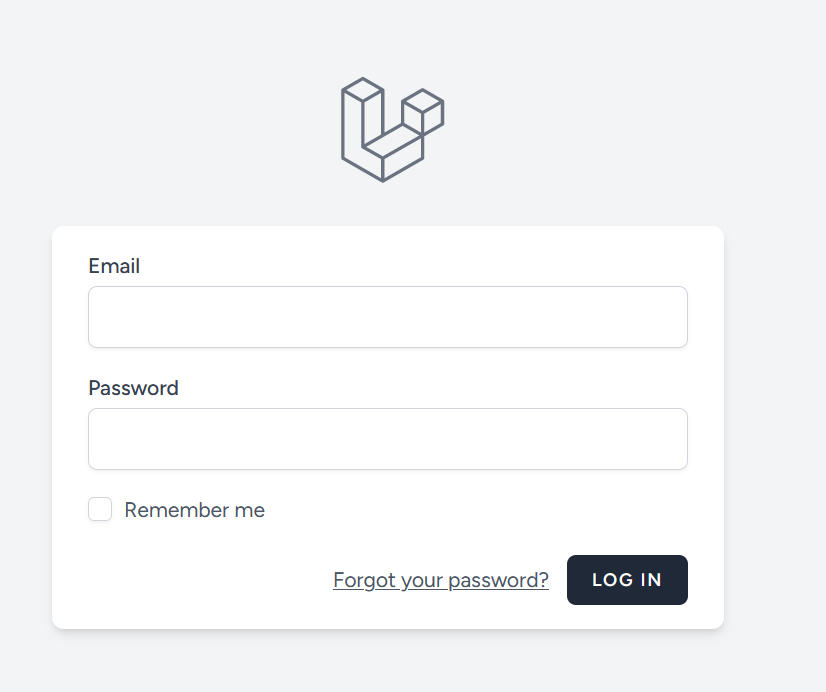
- Adjusted login request: changed from email to login.

- Consolidated authentication on the login page.

- Fetched data from the User model.

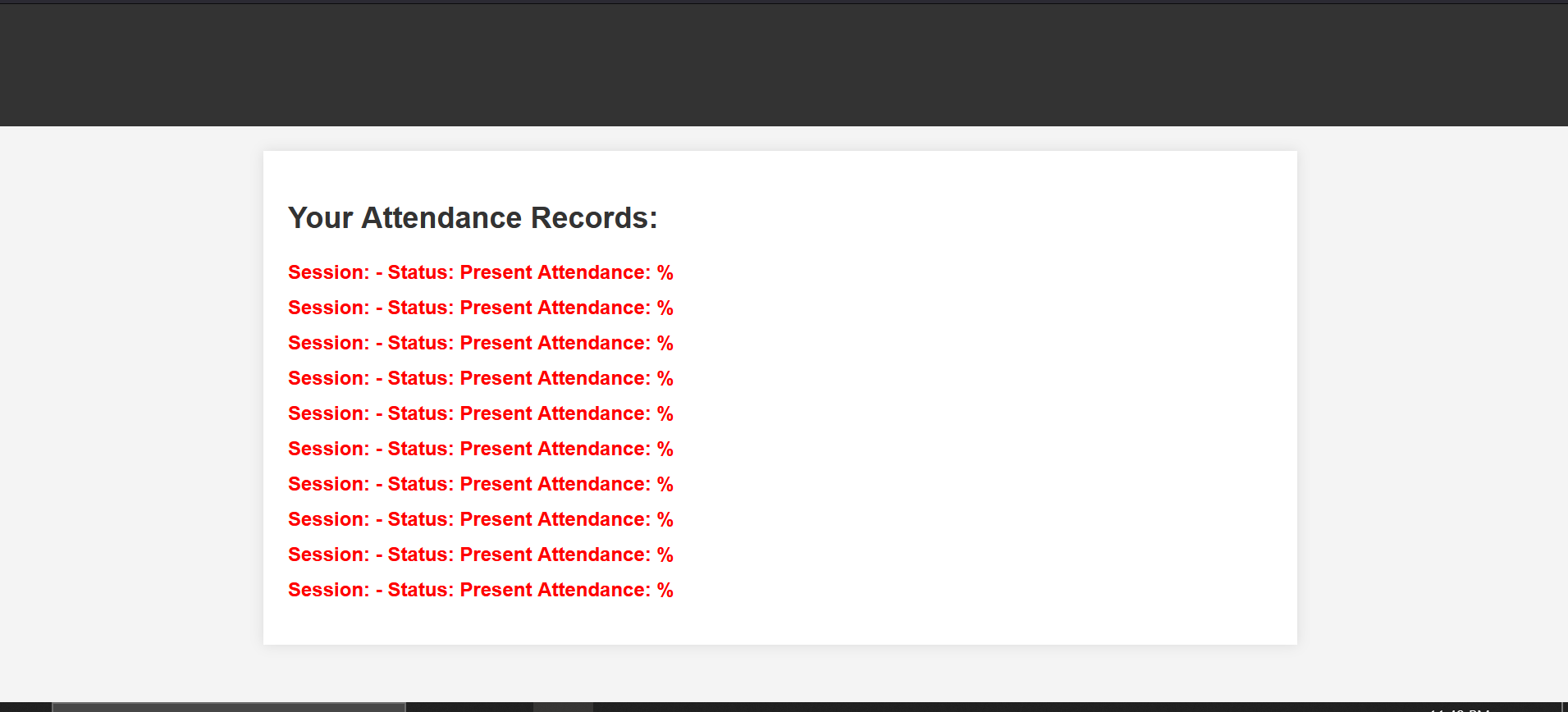
- Checked if the password, email, or name matched the user input.

- Proceeded if there was a match, changing from email to login where necessary.



USE [momin@gmail.com](mailto:momin@gmail.com) (email)

Password momin@123



Later on I added Models for other tables and also made Controllers for Student and Teacher

also added data in other tables using the phpMyAdmin

used Eloquent orm in the controller for fetching data from the database and adding data

**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. |
| R2 | Modern Solutions | CLO-3 | PLO-5 | The student is absent in the lab/ The submission is plagiarised. | The student is unable to build the website/webapplication using the specified modern tool or framework | Inbetween | The student is trying to use some of the modern tools but is unable to utilize all the appropriate tools or to build the correct solution. | Inbetween | The student is using all the appropriate modern tools to complete the tasks at hand, and provide a modern solution. |
| 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. |