

Project Report: Online Attendance Management System

Name: Rahul Prasad Vagu

Internship ID: TS-RISE-DBSM-2579

Date: 15-06-2025

1. Project Overview

This project is a web-based Attendance Management System built using Flask and MySQL. It allows marking, viewing, and summarizing student attendance across multiple subjects and dates. It features a clean UI, summary statistics, and scalable database design.

2. Technologies Used

- Frontend: HTML, CSS, Bootstrap
- Backend: Python (Flask)
- Database: MySQL
- Tools: VS Code, MySQL Workbench

3. Features Implemented

- Mark attendance for all students by subject and date
- View attendance records in tabular format
- Generate summary statistics (Present/Absent/Total)
- Stylish homepage with navigation cards
- Back buttons and smooth navigation

4. Application Screenshots

Attendance Management System

127.0.0.1:5000

Attendance Management System

Manage and track student attendance easily

Mark Attendance

Mark daily attendance for all students by subject and date.

Mark Attendance

View Attendance

View attendance records for a given date and subject.

View Attendance

Attendance Summary

See summary of total, present, and absent students.

View Summary

© 2025 Rahul Prasad Vagu | Internship ID: TS-RISE-DBSM-2579

127.0.0.1:5000/view-attendance

Mark Attendance Form

Mark Attendance

127.0.0.1:5000/mark-attendance

Mark Attendance

Date

14-06-2025

Subject

English

Students

☒ 1 - Student 1

☐ 4 - Student 4

☒ 7 - Student 7

☒ 10 - Student 10

☐ 13 - Student 13

☒ 16 - Student 16

☐ 19 - Student 19

☒ 22 - Student 22

☐ 25 - Student 25

☒ 28 - Student 28

☒ 31 - Student 31

☐ 34 - Student 34

☐ 37 - Student 37

☒ 40 - Student 40

☒ 43 - Student 43

☒ 46 - Student 46

☐ 49 - Student 49

☒ 2 - Student 2

☒ 5 - Student 5

☐ 8 - Student 8

☒ 11 - Student 11

☒ 14 - Student 14

☐ 17 - Student 17

☐ 20 - Student 20

☒ 23 - Student 23

☐ 26 - Student 26

☐ 29 - Student 29

☒ 32 - Student 32

☒ 35 - Student 35

☐ 38 - Student 38

☒ 41 - Student 41

☐ 44 - Student 44

☒ 47 - Student 47

☒ 50 - Student 50

☒ 3 - Student 3

☒ 6 - Student 6

☐ 9 - Student 9

☒ 12 - Student 12

☐ 15 - Student 15

☒ 18 - Student 18

☒ 21 - Student 21

☐ 24 - Student 24

☒ 27 - Student 27

☐ 30 - Student 30

☒ 33 - Student 33

☐ 36 - Student 36

☐ 39 - Student 39

☒ 42 - Student 42

☐ 45 - Student 45

☒ 48 - Student 48

☒ Submit Attendance

View Attendance Table

 Attendance on 2025-06-14 - English

| Student ID | Name | Status |
|------------|------------|---------|
| 1 | Student 1 | Absent |
| 1 | Student 1 | Present |
| 2 | Student 2 | Absent |
| 2 | Student 2 | Present |
| 3 | Student 3 | Absent |
| 3 | Student 3 | Present |
| 4 | Student 4 | Present |
| 4 | Student 4 | Absent |
| 5 | Student 5 | Absent |
| 5 | Student 5 | Present |
| 6 | Student 6 | Present |
| 6 | Student 6 | Present |
| 7 | Student 7 | Absent |
| 7 | Student 7 | Present |
| 8 | Student 8 | Absent |
| 8 | Student 8 | Absent |
| 9 | Student 9 | Absent |
| 9 | Student 9 | Absent |
| 10 | Student 10 | Absent |
| 10 | Student 10 | Present |
| 11 | Student 11 | Absent |
| 11 | Student 11 | Present |
| 12 | Student 12 | Absent |
| 12 | Student 12 | Present |
| 13 | Student 13 | Absent |
| 13 | Student 13 | Absent |

Summary Result

Summary Result

127.0.0.1:5000/attendance-summary

Summary for Mathematics on 2025-06-14

| |
|--------------------|
| Total Students: 50 |
| Present: 11 |
| Absent: 39 |

[Back to Summary Form](#)

Code and Flask Logs in VS Code

File Edit Selection View Go Run Terminal Help

AttendanceWebApp

EXPLORE app.py attendance_summary.html attendance_summary_result.html mark_attendance.html view_attendance.html view_attendance_result.html

ATTEN...> _pycache_> static> templates> attendance_summar...> attendance_summar...> index.html> mark_attendance.ht...> view_attendance_res...> view_attendance.html

app.pydb_config.pyrequirements.txt

OUTLINE

TIMELINE

1 from flask import Flask, render_template, request2 from db_config import get_connection34 app = Flask(__name__)56 # Home route7 @app.route('/')8 def index():9 | return render_template('index.html')1011 # Mark Attendance route12 @app.route('/mark-attendance', methods=['GET', 'POST'])13 def mark_attendance():14 | conn = get_connection()15 | cursor = conn.cursor()1617 | if request.method == 'POST':18 | | subject_id = request.form['subject']19 | | attendance_date = request.form['date']20 | | present_ids = request.form.getlist('present')21 |22 | | cursor.execute("SELECT student_id FROM Students")23 | | student_ids = cursor.fetchall()2425 | for (sid,) in student_ids:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

* Debugger PIN: 468-609-855127.0.0.1 - - [14/Jun/2025 14:24:30] "GET / HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:24:32] "GET /mark-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:24:40] "GET /view-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:24:53] "POST /view-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:24:56] "GET /view-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:25:01] "GET /mark-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:26:41] "GET /mark-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:27:37] "POST /mark-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:27:39] "GET /mark-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:27:42] "GET /view-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:27:52] "POST /view-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:28:20] "GET /view-attendance HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:28:23] "GET /attendance-summary HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:28:31] "POST /attendance-summary HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:28:51] "GET /attendance-summary HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:29:02] "POST /attendance-summary HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:29:06] "GET /attendance-summary HTTP/1.1" 200 -127.0.0.1 - - [14/Jun/2025 14:29:14] "POST /attendance-summary HTTP/1.1" 200 -

5. Developer Remarks

This project demonstrates a complete CRUD-based web app with clean design and MySQL integration. The use of Bootstrap ensures a responsive UI, while Flask keeps the backend logic simple and effective.