Attendance Management System

1. Project

- **Project Goal:** To build a simple web-based application to record, store, and view student attendance efficiently·

GitHub Link:
Adarsh K G Kasaudhan :https://github.com/AdarshKasaudhan1/Agentic-AI-project-Attendance-management-system-
Sharvani Reddy: https://github.com/sharvani801/Agentic_AI_Project
Keerthana R:https://github.com/Keerthana2562006/Agenetic_AI_Project

2. Abstraction / Summary

The Attendance Management System is a simple web-based application designed to automate the manual process of recording student attendance. Built using basic web technologies (HTML, CSS, JavaScript) , the system minimizes manual errors and provides an efficient solution for attendance tracking[55]. Data is stored locally in the browser's LocalStorage. The system is structured with distinct modules for entry, display, and calculation, demonstrating core Software Engineering concepts·

3. Introduction

Traditional attendance tracking methods are often manual, time-consuming, and prone to error[8]. This project addresses these challenges by using basic web technologies to create a lightweight, easy-to-use system that automates the attendance marking process. The system's architecture is minimal, focusing on core functionality to demonstrate fundamental Software Development Life Cycle (SDLC) phases and software engineering principles.

4. Aim / Main Goal of the Project

The main goal of this project is to develop a simple, functional Attendance Management System that helps users:

- Automate attendance marking process.
- Reduce manual errors[12].
- Generate daily attendance reports.
- Provide an efficient solution for attendance tracking.

5. Proposed Solution (MVP Idea)

The Minimum Viable Product (MVP) for the Attendance Management System proposes a simple web interface that:

- Accepts basic student details (ID, Name) and a date, along with the attendance status (Present/Absent).
- Allows the user to mark attendance via a button click.
- Displays all recorded attendance records in a tabular format.
- Stores data locally using browser **LocalStorage**.
- Includes modules for Attendance Entry, Attendance Display, and Attendance Percentage Calculation.

6. Design / Architecture

The system follows a simple client-side architecture where the frontend handles both the user interface and data persistence.

System Design

The system consists of three main parts:

1. **Input Form:** To enter student details, date, and select the attendance status.
2. **Display Table:** To show the current list of attendance records.
3. **Data Storage:** Data is persisted using the browser's **LocalStorage**.

Modules Description

- **Attendance Entry Module:** Captures student ID, name, date, and status, and saves the record to attendanceList in LocalStorage.
- **Attendance Display Module:** Reads the attendanceList from LocalStorage and dynamically updates the HTML table to show all records.
- **Attendance Percentage Calculation Module:** Responsible for calculating attendance statistics (e.g., total days present, total days absent, and overall percentage).

7. Technology Used

Component	**Technology**	**Description**
Frontend/UI	HTML, CSS, JavaScript	Defines structure, styling, and client-side logic.
Logic/Backend	JavaScript	Handles data collection, validation (e.g., checking for empty fields), and data storage management.
Data Storage	LocalStorage	Used for persistent storage of the attendanceList array within the user's browser.
Development Tools	Web Browser	Required for running the application.

8. Advantages and Limitations

Advantages

- **Easy to Use:** Simple web interface.
- **Saves Time:** Automates the recording process.
- **Reduces Paperwork:** Digital record keeping.
- **Reduces Manual Errors:** Input validation helps ensure data integrity.

Limitations (Implied & Future Enhancements)

- **No Persistence/Scalability:** Data is stored in LocalStorage, meaning it's tied to the specific browser and user, and is not scalable for multiple users or large datasets.
- **Lack of Security:** There is no authentication system.
- **Limited Reporting:** The current system primarily displays raw data; it lacks full report generation and export features.

9. Future Enhancements

The project identifies key areas for future development to improve functionality and robustness:

- **Database Integration:** Implement a persistent backend database to replace LocalStorage (e.g., using MySQL or similar).
- **Authentication System:** Develop a secure login system with authentication for authorized user access.
- **Export Reports:** Add functionality to export attendance reports to formats like CSV or PDF