**Attendance Management System**

A user friendly Attendance Management System that allows instructors to register their student’s record, record the attendance and generate attendance reports for specific date ranges. Meanwhile all the student’s data and their attendance record is storing in SQLite data base.

**Features**

* Login system with role-based access (Instructor).
* Instructors can add and manage student records.
* Mark student’s attendance for each day.
* Generate comprehensive attendance reports for specific date ranges.
* Uses SQLite for reliable data storage.
* Intuitive GUI built with Tkinter for seamless interaction.

**Installation**

**Prerequisites**

* **Python 3.7 or higher**: Ensure Python is installed on your system. You can download it from [Python's official website](https://www.python.org/downloads/).

**Steps**

1. **Clone the Repository**

git clone https://github.com/yourusername/attendance-management-system.git

cd attendance-management-system

1. **Install Required Dependencies**

pip install -r requirements.txt

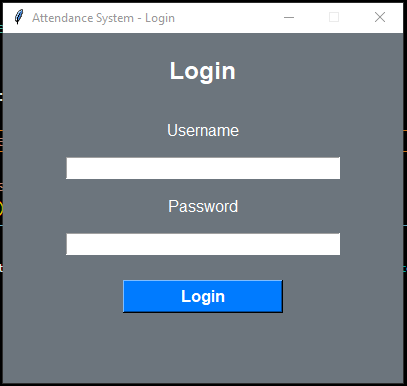
*Note: Tkinter usually comes pre-installed with Python. If not, refer to your operating system's instructions to install it.*

1. **Run the Application**

python attendance\_system.py

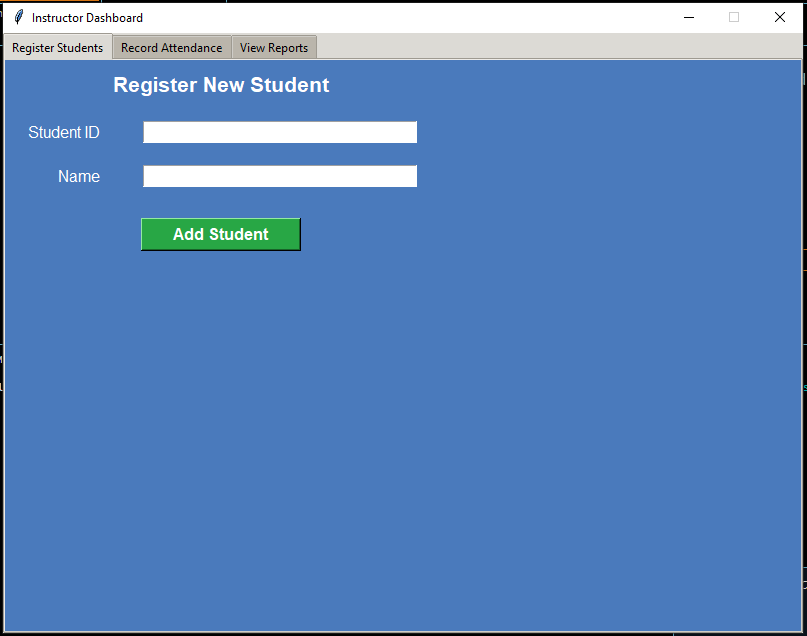
**Usage**

1. **Login Interface**

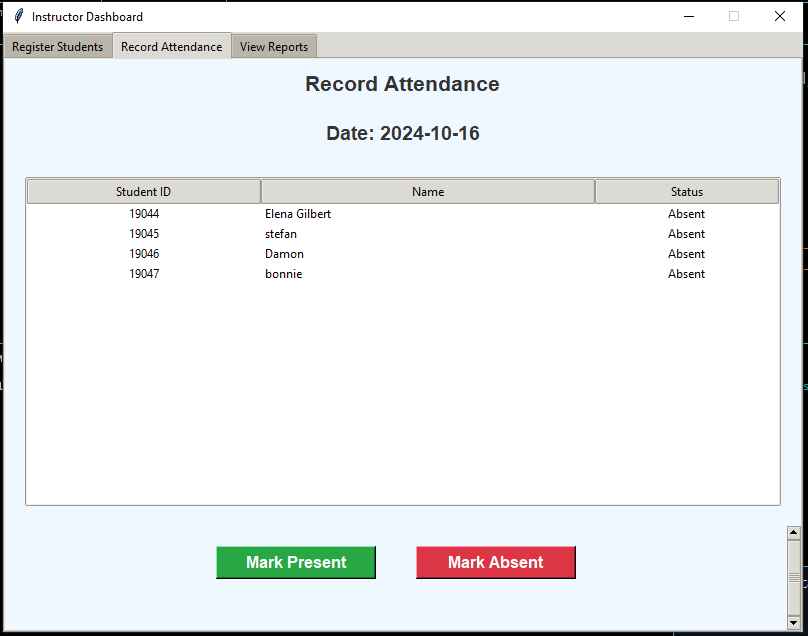


Here Instructors can enter their specific username and password

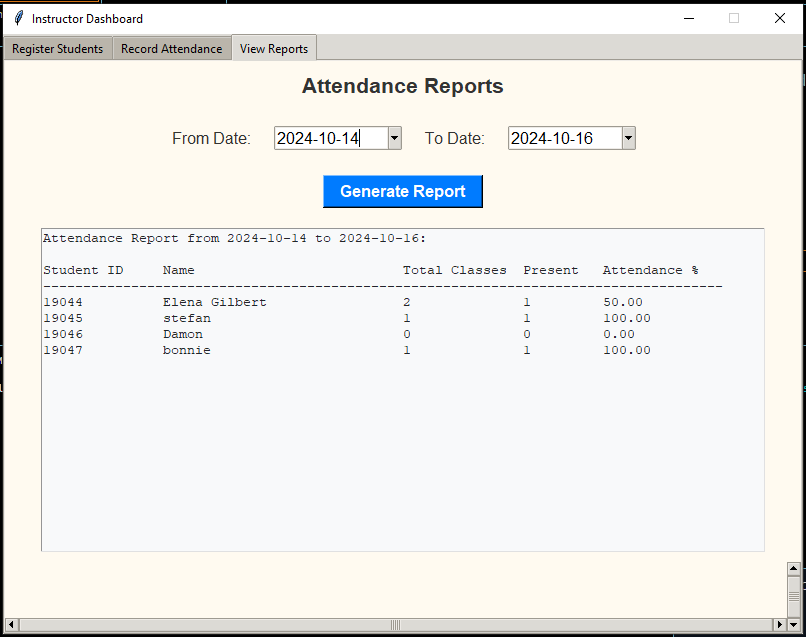
1. **Register Students**
   * Navigate to the **Register Students** tab.
   * Enter the student's ID and name.
   * Click **Add Student** to save the record.



1. **Record Attendance**
   * Go to the **Record Attendance** tab.
   * The current date is displayed.
   * Select one or multiple students from the list.
   * Click **Mark Present** or **Mark Absent** to update attendance status.



1. **View Reports**
   * Access the **View Reports** tab.
   * Select the desired date range using the date pickers.
   * Click **Generate Report** to view attendance statistics.
   * The report includes total classes, number of presents, and attendance percentage for each student.



**Functionalities**

**1. User Authentication**

* **Login System:** Secure login for instructors.
* **Password Hashing:** Utilizes SHA-256 for password security.
* **Role-Based Access:** Currently supports the Instructor role.

**2. Student Management**

* **Add Students:** Register new students with unique IDs.
* **View Students:** Display a list of all registered students.
* **Data Integrity:** Ensures no duplicate student IDs are entered.

**3. Attendance Management**

* **Record Attendance:** Mark students as present or absent for the current day.
* **Update Records:** Modify attendance status if needed.
* **Automatic Date Handling:** Records attendance based on the current date.

**4. Reporting**

* **Generate Reports:** Create attendance reports for all students.
* **Date Range Selection:** Customize reports based on specific time frames.
* **Comprehensive Data:** Includes total classes, present count, and attendance percentages.

**Database Structures and File Formats**

**Database**

* **SQLite Database (attendance\_system.db):** Stores all application data.

**Tables**

1. **Users**

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id | INTEGER | Primary key, auto-incremented |
| username | TEXT | Unique username for login |
| password | TEXT | Hashed password using SHA-256 |
| role | TEXT | User role (e.g., Instructor) |

1. **Students**

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id | INTEGER | Primary key, auto-incremented |
| student\_id | TEXT | Unique identifier for the student |
| name | TEXT | Full name of the student |

1. **Attendance**

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id | INTEGER | Primary key, auto-incremented |
| student\_id | TEXT | References student\_id in the Students table |
| date | TEXT | Date of attendance in YYYY-MM-DD format |
| status | TEXT | Attendance status (Present or Absent) |

**File Formats**

* **Python Scripts (.py):** Contains the application's source code.
* **Database File (.db):** SQLite database storing all persistent data.
* **README (README.md):** Documentation and usage instructions.