TIME TRACKING SYSTEM WITH FLASK OR GUI

DESCRIPTION:

We are seeking a proficient programmer to create a comprehensive time tracking system using either Flask or a graphical user interface (GUI). This system will integrate MongoDB and MySQL databases, employing specific table structures detailed below:

TABLES:

Employees

Schedule

Clocking

TABLE STRUCTURES:

```
CREATE TABLE IF NOT EXISTS employees (
    id INTEGER PRIMARY KEY,
    uid TEXT UNIQUE,
    name TEXT,
    schedule_id TEXT
)

CREATE TABLE IF NOT EXISTS clocking (
    id INTEGER PRIMARY KEY,
    employee_id INTEGER,
    date TEXT,
    time TEXT,
    incident TEXT,
    FOREIGN KEY (employee_id) REFERENCES employees(id)
)


CREATE TABLE IF NOT EXISTS Schedule(
    id INTEGER PRIMARY KEY,
    schedule_id INTEGER,
    name TEXT,
    start_time TEXT,
    end_time TEXT,
    break_time TEXT
)
```

WORKER REQUIREMENTS:

- Intuitive interface for clocking in or out.
- Features for medical leave, work leave, and logging overtime hours.
- Mechanisms to prevent duplicate clock-ins/clock-outs.
- Weekly report generation.
- Audible and visual confirmation upon successful clocking.
- Each user will have an assigned schedule; varying schedules should be accommodated. The system should assess punctuality and calculate worked hours accordingly.

ADMINISTRATOR REQUIREMENTS:

- Ability to manage users by adding, modifying, or deleting them.
- Functionality to reassign schedules to employees.
- Manual clock-in and clock-out capabilities for administrators.
- Ability to edit clocking records.
- Report generation for individual employees or groups.
- Real-time visualization of employee attendance status.
- PDF report generation.

ADDITIONAL DETAILS:

- Experience with Flask or GUI development is essential.
- Familiarity with MongoDB and MySQL databases is required.
- Strong problem-solving skills and attention to detail are necessary.
- The ability to deliver clean and efficient code is expected.
- Applicants should submit proposals showcasing relevant experience and examples of previous work.

CURRENT WORKFLOW:

- The system utilizes an NFC reader connected to a Raspberry Pi to scan NFC cards or tags.
- When an NFC card or tag is detected, its unique identifier (UID) is captured by the NFC module.
- The system compares the UID with registered employees' UIDs stored in a CSV file.
- Upon finding a match, the employee's information, including name and unique ID, is retrieved.
- The system records the clock-in/out time by adding an entry to a CSV file for the current date.
- A sound notification confirms successful clocking.

IMPLEMENTATION DETAILS:

- The program utilizes the PN532 NFC module and the pn532 Python library for interfacing with the NFC reader.
- A screen will be integrated with the Raspberry Pi to enhance user interaction and provide visual feedback.
- If you're interested and capable of meeting these requirements, please submit your proposal. We look forward to reviewing your application!