

**Group member name**

<b>1.Bereket Sahlemariam Nega</b>	<b>1601439</b>
<b>2.Dawit Mulugeta Melaku</b>	<b>1601093</b>
<b>3.Hermela Girma</b>	<b>1601254</b>

## Project progress

### Human Resource Management System(HRMS)

#### Project Progress

#### 1. Project Title: Human Resource Management System (HRMS) Using Python

- **Completion Status:**
  - The core functionalities for employee, department, position, and attendance management have been implemented.
  - CRUD operations for all entities are working, and the data is being successfully stored in file.txt.
  - The system allows for viewing, updating, and deleting records as required.
- **Milestones Completed:**
  - a. **Design and System Architecture:** Completed.
  - b. **System Implementation:** The CRUD functionality for all entities (Employee, Department, Position, etc.) is complete.
  - c. **Testing:** Basic testing for data insertion, deletion, and update has been conducted.
- **Next Steps:**
  - **Refining the System:** Further testing to ensure robust handling of all edge cases.
  - **User Documentation:** Prepare the user manual for easy interaction with the system.
  - **Presentation:** Prepare the final demonstration for the stakeholders or project review

#### 2. Project progress Overview:

The **Human Resource Management System (HRMS)** is a Python-based application designed to handle and store information related to the organization's human resources. This system focuses on managing employees, departments, clients, positions, allowances, salary scales, vehicles, attendance, and their relationships. The data will be stored in a simple text file

(file.txt) in a comma-separated format, allowing easy access and management of employee-related data. The HRMS will offer a basic yet efficient platform to manage various HR-related

functions such as employee record management, payroll, attendance tracking, and department-client relationships.

### 3. Project Objectives:

The **HRMS** project has the following objectives:

- **Data Management:** To store and manage employee-related data including personal information, department affiliation, job position, salary, and attendance.
- **HR Process Automation:** To allow HR staff to easily update, insert, view, and remove employee details and payroll records.
- **Attendance Management:** To track employee attendance and provide status updates for HR reporting purposes.
- **Salary Management:** To facilitate salary management by defining salary scales and allowances.
- **Simple File Storage:** Data will be stored in a text file format (file.txt) to keep the system lightweight and easily portable.

### 4. Project Scope:

The system will focus on key HR entities and their relationships:

- **Department:** Information about various departments within the organization.
- **Client:** Store data related to clients associated with different departments.
- **Employee:** Employee details including personal information, department assignments, positions, and salary-related data.
- **Position:** Job positions within the organization and their related data.
- **Allowance:** Benefits and allowances associated with employee positions.
- **Salary Scale:** Salary ranges for different positions.
- **Vehicle:** Vehicles assigned to employees based on their positions.
- **Attendance:** Attendance data for employees, including time-in and time-out.
- **Department \_ Client:** Mapping of departments to clients.
- **Employee\_ Position:** Mapping of employees to job positions.

### 5. Functional Requirements:

- **Add Data:** The system will allow users to add new records for each entity (Employee, Department, Client, etc.).
- **Update Data:** Users can modify existing records (e.g., updating an employee's department or position).

- **Delete Data:** Users can remove records from the system (e.g., deleting an employee or department).
- **View Data:** The system will provide functionality to view the details of employees, departments, and other entities.
- **Menu-Based Interface:** A user-friendly menu will guide users to perform the required operations.
- **Text File Storage:** All records will be stored in a simple text file (`file.txt`) for persistence.

## 6. Technology Stack:

- **Programming Language:** Python 3.13.3
- **Data Storage:** Text file (`file.txt`)
- **Libraries:**
  - `input()` To take user inputs
  - `open()`, `readlines()`, `writelines()`: For reading from and writing to the text file
- **Text File Format:** Comma-separated values (CSV) for each record stored in the text file.

## 7. Expected Deliverables:

- **Python Code:** A working Python script implementing the HRMS with all CRUD functionalities.
- **Text File:** The `file.txt` where all data is stored.
- **Documentation:** User and technical documentation describing the operations, the design, and implementation of the system.
- **Demo/Presentation:** A live demonstration of the system, showcasing its features.

## 8. Conclusion:

This **Human Resource Management System (HRMS)** is designed to be a simple, yet powerful tool for managing employee information, attendance, positions, and payroll in small to medium-sized organizations. Using Python and a text file for storage makes the system lightweight and easy to set up, without the need for complex database setups. The system can be further expanded to include additional HR-related functionalities such as performance management, leave tracking, and more.

## case study( Human resource management System)

### use in this project

A small accounting firm wants a simple HR application that will help it to keep track of its employees, their positions, allowances, salary scales, and which company vehicles their employees drive. The application must keep track of all the positions at the firm, the employees filling these positions, the allowances for these positions, the salary scales for these positions, and the company vehicles assigned to these positions.

**Employees** will be characterized by Employee Id, SS Number, First Name, Last Name, Middle Name, Gender, DOB, Email, Mobile, Address Line, City, State and Post Code.

**Positions** will be characterized by Position Id, Position Name, Position Description and Details. **Allowances** will be characterized by Allowance\_ Id, Allowance\_ Name, Allowance\_ Description and Amount.

**Salary Scales** will be characterized by Salary Scale Code, Salary Scale Name, Salary Scale Description, Minimum Salary and Maximum Salary.

**Vehicles** characterized by Vehicle Id, VIN, Registration No, Year, Make, Model and Color.

**Client** characterized by Client Id (Primary Key), Client Name , Contact Person , Email , Phone , Address, City , State and Post Code.

**Department** will be characterized by Department Id (PK), Department Name, Department Description.

**Attendance** will be characterized by Attendance Id (PK), Employee Id (FK), Date, Time In, Time Out, Status.

A vehicle is assigned to a position; Allowances may be allocated for Positions; Employees may fill Positions and a position may have no or many Salary scales