ADDIS ABABA
SCIENCE AND
TECHNOLOGY
UNIVERSITY
UNIVERSITY FOR INDUSTRY

COLLEGE: Electrical and Mechanical Engineering

Department of Software Engineering

TITLE: Payroll managements system(Documentation)

SECTION: A

GROUP MEMBERS	ID
1 AHLAM AHMED ALI	ETS0115/16
2 ALEHEGNE GETA ENYEWU	ETS0130/16
3 AMANAWIT BEHAILU GEREMEW	ETS0135/16
4 AMANUEL AYELE GETACHEW	ETS0140/16
5 AMANUEL GETACHEW KASSA	ETS0147/16
6 AMANUEL HABTAMU ABERA	ETS0148/16

Submitted to Mr Kassahun Admek
Submission date: Jan 6 2025

Catalog

1. Abstract

The *Payroll Management System* streamlines employee management and automates payroll calculations. Built with C++ for high speed and efficiency, it handles large requests seamlessly and simplifies tasks such as adding, updating, searching, and deleting employee records. Designed for scalability and easy maintenance, the system reduces manual workload and enhances organizational productivity.

2. Introduction

Efficient employee management and payroll calculation are crucial for any organization. Manual processes are often time-consuming, prone to errors, and challenging to scale. To address these challenges, the *Payroll Management System* was developed as a software solution to automate and simplify these tasks.

The system is designed to handle key functions such as adding, viewing, updating, searching, and deleting employee records while calculating payroll accurately based on employment types. By leveraging C++ for its high performance and scalability, the system ensures reliability, handles large datasets, and is easy to maintain and expand.

This document provides a comprehensive overview of the project, including the use of GitHub for collaboration, a detailed explanation of the code, and the steps of the Software Development Life Cycle (SDLC) followed during development. It also includes user and technical documentation, along with challenges faced and lessons learned.

3. Use of GitHub

The project repository is hosted on GitHub, providing a centralized platform for version control and collaboration. Team members used GitHub to track changes, manage contributions, and resolve issues efficiently. Features like branches and pull requests facilitated collaborative development and ensured code quality. The repository also includes all project files, documentation, and commit history for transparency and ease of access.

Repository Link:

https://github.com/Alehegne/Payroll-Management-System/tree/main

4. Code Description and Explanation

Overall Structure

The *main.cpp* file forms the core of the Payroll Management System. It is modularly designed with functions handling different functionalities and a well-organized structure to manage employee data. The code begins with global constants and variables to define the maximum number of employees and store their data. Key function prototypes and utility functions, such as clearing input buffers and converting strings to lowercase, ensure smooth operation.

The main program includes a menu-driven interface that allows users to interact with various features such as adding, viewing, searching, updating, deleting employees, and generating summary reports.

Main Functionalities

Add Employee:
Adds a new employee's details, including name, gender, ID, employment type, and salary. Input validation ensures accurate data entry and prevents duplicate IDs.

View Employee:
Displays all employee records in a formatted tabular layout, including name, ID, gender, type, and salary.

Search Employee:
Allows users to search for employees by name, gender, or salary. It provides results in a structured format for clarity.

Update Employee:
Updates an existing employee's details, including their ID, name, gender, employment type, or salary. It validates new entries for accuracy and consistency.

Delete Employee:
Deletes an employee record after confirmation, adjusting the data arrays to maintain integrity.

Summary Report:
Generates a report summarizing the number of employees by gender and type, total salaries paid, and statistics like maximum, minimum, and average salaries.

Key Global Variables and Constants

Constants:

max_employees defines the maximum number of employees (100).

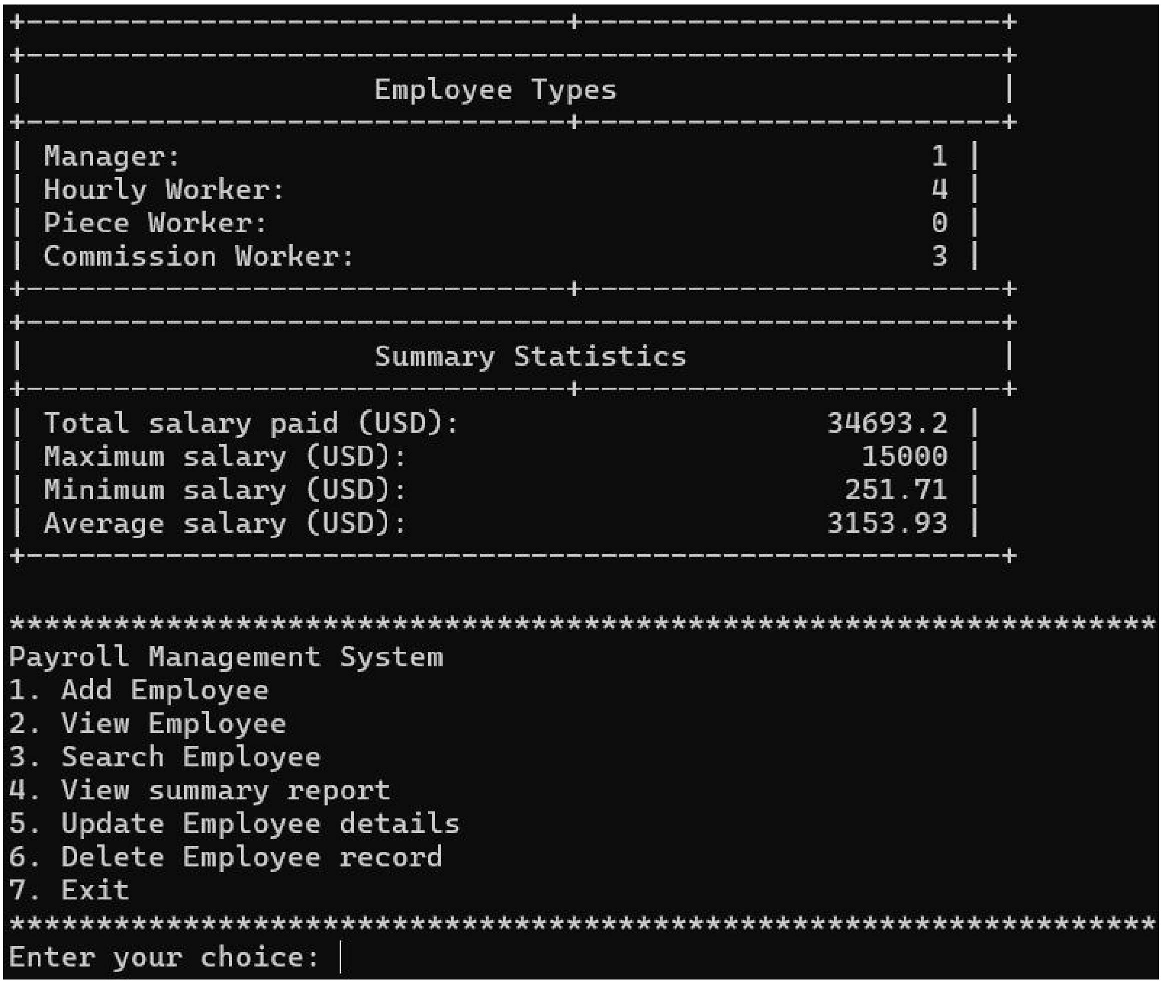
Global Variables:

- names[], gender[], employee_type[]: Arrays to store employee details.
- hoursWorked[], salary[]: Arrays to track hours worked and salaries.
- worker_count: Tracks the current number of employees.
- worker_id[]: Stores unique IDs for each employee.

This structured approach ensures scalability, ease of maintenance, and efficient handling of employee data.

- **Example**

  Execution:



```
+--------------------------------+-------------------------+
+----------------------------------------------------------+
|                    Employee Types                        |
+--------------------------------+-------------------------+
| Manager:                                             1 |
| Hourly Worker:                                       4 |
| Piece Worker:                                        0 |
| Commission Worker:                                   3 |
+--------------------------------+-------------------------+
+----------------------------------------------------------+
|                    Summary Statistics                    |
+--------------------------------+-------------------------+
| Total salary paid (USD):                       34693.2 |
| Maximum salary (USD):                            15000 |
| Minimum salary (USD):                           251.71 |
| Average salary (USD):                          3153.93 |
+----------------------------------------------------------+

*******************************************************************
Payroll Management System
1. Add Employee
2. View Employee
3. Search Employee
4. View summary report
5. Update Employee details
6. Delete Employee record
7. Exit
*******************************************************************
Enter your choice: |
```

5. Software Development Life Cycle (SDLC) Phases

- **Phase 1: Planning**

Problem Identification and Scope

Manual employee management and payroll processing are time-consuming, error-prone, and difficult to scale. This project aims to develop a system that automates these tasks, improving accuracy, efficiency, and scalability for organizations of varying sizes.

Initial Requirements Gathering

The system must support adding, viewing, updating, searching, and deleting employee records, calculate payroll based on employment type, and generate summary reports. It should handle up to 100 employees and ensure easy maintenance and scalability.

- **Phase 2: Analysis**

Analysis of Employee Management Requirements

The system must manage employee data efficiently, calculate payroll accurately for various employment types, and provide functionalities like adding, searching, updating, and generating reports. It should handle up to 100 employees and ensure data integrity with validation.

Technologies Chosen

The project is implemented in C++ for its high performance and efficiency. Standard libraries such as <iostream>, <iomanip>, <string>, <cctype>, and <limits> are used for input/output handling, data manipulation, and formatting.

- **Phase 3: Design**

Explanation of System Design

The system uses arrays to store employee data (e.g., names, IDs, salaries) for efficient retrieval and updates. Functions are modularly designed to handle specific tasks like adding, viewing, searching, updating, and deleting records, ensuring clarity and maintainability.

Illustrations

The system flow begins with a user menu leading to specific functionalities. Each function processes data and interacts with global arrays, ensuring logical flow.

Notice: Flowchart is attached to the last page.

- **Phase 4: Implementation**

Coding and Testing Approaches

The system was implemented using C++ with a modular approach, dividing functionalities into separate functions for clarity and maintainability. Testing was conducted iteratively to validate each function and ensure accuracy in data handling and payroll calculations.

Division of Responsibilities

Team members were assigned specific tasks:

- Adding, updating, deleting employees.
- Implementing search functionality and summary report.
- Managing the menu and overall program integration.

- **Phase 5: Testing**

Testing Strategies

- **Unit Testing**: Individual functions like adding, updating, and searching were tested with valid and invalid inputs to ensure correctness.
- **Integration Testing**: Verified interactions between functions, such as how employee updates reflect in search and summary reports.

Example Test Cases and Results

- **Case 1**: Add an employee with valid details → Employee added successfully.
- **Case 2**: Search for a non-existent employee → Correct "not found" message displayed.
- **Case 3**: Update an employee ID to a duplicate → Error message prevents duplication.

- **Phase 6: Deployment and Maintenance**

The Payroll Management System is deployed by compiling the C++ program and making it available for use on the target machines. After successful deployment, the program enters the maintenance phase, where it will be monitored for any issues, bugs, or performance improvements. Regular updates will be provided to ensure compatibility with newer operating systems, and any errors reported by users will be promptly addressed to maintain the system's reliability.

In terms of future enhancements, one of the key improvements would be transitioning from a Command-Line Interface (CUI) to a Graphical User Interface (GUI). This change would make the system more user-friendly, allowing for easier navigation, better interaction, and the ability to visualize payroll data through charts or tables. The introduction of a GUI would also enable more intuitive management of employee records, making the program accessible to a broader range of users with varying levels of technical proficiency.

6. User Guide

User Guide for Payroll Management System

This guide provides instructions on using the Payroll Management System to manage employee payroll, including adding, updating, deleting, and searching employee records.

1. Launching the Program

- Open the executable to start the program and access the main menu.

2. Entering Employee Details

- Enter the employee’s **name**, **gender**, **employment type** (Manager, Hourly Worker, Commission Worker, or Pieceworker), and relevant details (weekly sales or items produced).
- The system will automatically calculate the weekly pay.

3. Viewing Weekly Pay

- The weekly pay will be displayed in a tabular format after entering the details.

4. Searching Employee Records

- Select **Search** from the menu, enter the employee's **name**, and view their details.

5. Updating Employee Details

- Choose **Update**, input the employee’s **name**, and modify details like name, gender, employment type, or weekly sales/items produced. The pay will be recalculated.

6. Deleting Employee Record

- Select **Delete**, enter the employee’s **name**, and confirm the deletion.

7. Generating a Summary Report

- Select **Generate Report** to see:
    - Total number of employees
    - Total payments
    - Minimum and maximum payments by category

8. Exiting the Program

- Select **Exit** to close the program.

Tips:

- Ensure all fields are filled correctly to avoid errors.
- Follow input format prompts to prevent invalid data.

7. Conclusion and Future Scope

In conclusion, the Payroll Management System provides an efficient solution for managing employee payroll and records, automating tasks such as adding, updating, searching, and deleting employee details. Developed using C++ for its speed and scalability, the system simplifies payroll calculations and enhances productivity within organizations. The project's successful implementation, along with future plans for a more user-friendly GUI, ensures that the system remains adaptable and accessible to a broader range of users, promoting continued growth and efficiency in employee management.

8. References

- Savitch, W. (2018). *Problem solving with C++* (10th ed.). Pearson.
- Bronson, G. J. (2012). *C++ for engineers and scientists* (3rd ed.). Cengage Learning.
- Zak, D. (2022). *An introduction to programming with C++* (8th ed.). Cengage Learning.