

**St. Mary’s University**

**ASSIGNMENT OF PROGRAMMING FUNDAMENTALS ll**

**DEPARTMENT OF COMPUTER SCIENCE**

**COURSE CODE: CoSc2011**

**SECTION: C**

**GROUP ASSIGNMENT**

**Group members ID No**

1. **Esrael Bekele……………………………………………... RCD/0207/2015**
2. **Samuel Tezera……………………………………………. RCD/0539/2015**
3. **Saymen Wendwosen………………………………………RCD/0566/2015**
4. **Yohannes Gebre……………………………………………RCD/0571/2015**

**Submitted to: Instructor Dawit Yetmgeta**

**Submission Date: FRI, 17 May, 2024**

*Employee Registration system*

***An employee registration system***, also known as an employee onboarding system or a human resource information system (HRIS), is a software application used by organizations to manage the process of adding new employees to the workforce. It streamlines and simplifies the onboarding process for both the employer and the new hire.

Here are some of the key uses of an employee registration system:

**For Employers:**

* **Improved Efficiency:** Automates tasks like collecting new hire information, sending welcome emails, and assigning training materials. This saves time and frees up HR personnel to focus on other strategic tasks.
* **Reduced Errors:** Ensures accurate data collection and reduces the risk of errors during data entry.
* **Streamlined Onboarding Process:** Provides a clear and consistent onboarding experience for all new hires, ensuring they have the information and resources they need to be successful.
* **Improved Compliance:** Helps organizations comply with legal and regulatory requirements related to employee onboarding.
* **Data-Driven Decision Making:** Provides valuable data and analytics on the onboarding process, allowing employers to identify areas for improvement.

***Diagram Description and Functionality***

**Functions:**

* Each class has member functions for accessing and modifying data and displaying information.
* ***Add Employee*** prompts for user input and creates a new Department object (representing an employee) adding it to the Employee Manager.
* ***Search Employee*** finds an employee by ID and displays details if found.
* ***Display Employees*** iterates through all employees in the manager and displays details.
* ***Delete Employee*** removes an employee by ID from the manager.
* ***Update Employee*** prompts the user to update an employee's details based on ID.
* ***Main function*** handles user interaction with a menu for adding, searching, updating, deleting, and displaying employees. It also incorporates a login functionality.

**Number of Functions:**

There are approximately 30-35 functions in the diagram, including:

* Class member functions (getters, setters, display, update details)
* Standalone functions for adding, searching, deleting, updating employees, and displaying all employees
* Login functions for authentication and displaying messages

**Components:**

* **Classes:**
  + **Employee**: Base class for employees with basic information (id, name, email, phone). It provides functions for getting and setting details and a virtual displayDetails function for displaying information.
  + **Address:** Inherits from Employee and adds address information (street, city, state, zipCode). It overrides displayDetails to include address details.
  + **Department:** Inherits from Address and adds department information (deptName, description). It overrides displayDetails to include department details.
  + **EmployeeManager:** Manages employee data. It stores employees in a vector and provides functionalities for adding, removing, searching, updating, and displaying all employees.
  + **Login:** Handles user login with basic dummy authentication logic (replace with a more secure approach).

**General Description:**

This code demonstrates inheritance for building an employee hierarchy and polymorphism for displaying details based on employee type. It offers a more interactive user experience with a menu-driven system for managing employees. However, consider these improvements:

* Implement more secure password storage (e.g., hashing).
* Error handling for invalid user input.

**Overall, this diagram provides a solid foundation for an employee registration system with functionalities for managing and manipulating employee data.**

***Class Diagram of Structure and function Design for Employee Registration System***

