

20284325.docx

by Divya Nethmi Amarasingha

Submission date: 31-Dec-2023 09:42AM (UTC+0000)

Submission ID: 220304494

File name: 126698_Divya_Nethmi_Amarasingha_20284325_2067441_1818393700.docx (1M)

Word count: 1981

Character count: 10908

Introduction

The Colombo Institute of Studies has made the decision to switch from manually handling staff records to an automated system in recognition of the requirement for accuracy and efficiency. By streamlining the personnel information management process, this shift seeks to empower HR managers and assistants. The proposed employee management system supports multiple user tiers, each with specialized features to improve the management process as a whole.

User Levels and Features:

Different user levels are defined by the system, with HR managers and administrators (admins) being the main categories. HR managers now have the power to easily onboard new hires, establish new departments and designations, and quickly search for employee information using a variety of parameters, including name, department, and EPF number.

Task 01.

Object Oriented Programming

1.1 what is OOP

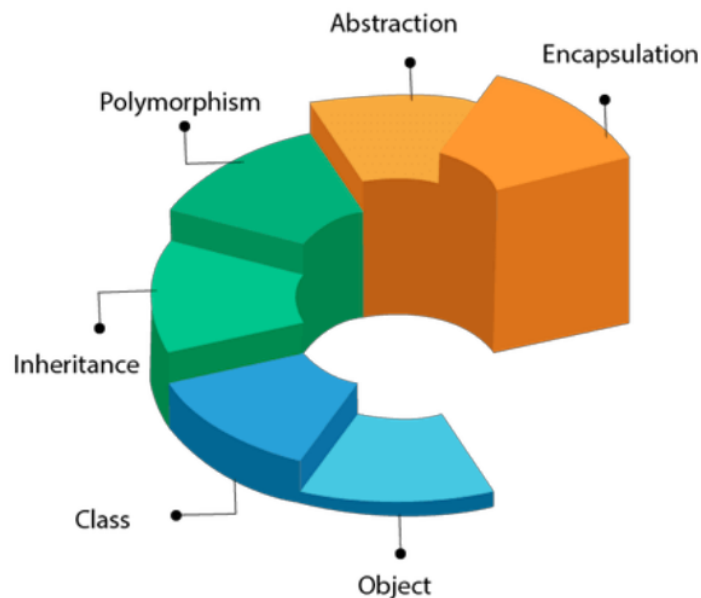
Classes and objects are used in the programming method known as object-oriented programming, or OOP. It breaks down software program into reusable code blueprints, or classes, that may be used to produce specific instances of objects later on. The programming languages PHP, Ruby, Perl, JavaScript, C++, Java, and Python are all object-oriented.

A class provides the building blocks for the production of more tangible, specialized items. Classes are big groupings of people with similar traits, like dog or bird. These classes specify the characteristics of a type instance, like color, but not the attribute values for an individual object.

Classes also have procedures, which are functions that are specific to that type of object. These functions, which are defined in the class, provide the intended purpose for the particular object type.

1.1.1 OOP Concepts.

OOPs (Object-Oriented Programming System)



- Object

In Java, classes and objects, together with their characteristics and functions, are connected to everything. For instance, an automobile is an object in the actual world. The car contains features like colour and weight, as well as functions like brake and drive.

A class is a "blueprint" for constructing things, similar to an object constructor..

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{
    HR_assistsave m3 = new HR_assistsave();
    m3.setVisible(b: true);
    this.hide(); // TODO add your handling code here:
}
```

Figure 1. example code for object

- Class

A class is an object collection. It makes sense as an entity.

Another way to think of a class is as a blueprint from which you can make a unique item. There is no space needed for class.

```
public class HR_assistsave extends javax.swing.JFrame {

    private static final String username="root";
    private static final String password="";
    private static final String dataConn="jdbc:mysql://localhost:3306/colombo_institute"

    Connection sqlConn =null;
    PreparedStatement pst =null;
    ResultSet rs =null;
    int n1,n2;
}
```

Figure 2 ex. for class

- Inheritance

It is feasible for classes to inherit properties and methods from one another in Java. The "inheritance concept" is divided into two groups:

- i. subclass (child) - the class that inherits from another class
- ii. superclass (parent) - the class being inherited from

To inherit from a class, use the extends keyword.

- Polymorphism

One activity carried out in multiple ways is referred to as polymorphism. To persuade the buyer in a different way, for instance, sketch a shape—a triangle, a rectangle, etc.

Polymorphism in Java is accomplished through the usage of overloading and overriding methods.

- Abstraction

Data abstraction is the process of presenting the user only the information that is absolutely necessary and hiding all other details.

Interfaces or abstract classes can be used to accomplish abstraction. An abstract class must be inherited from another class in order to access it; it is a limited class that cannot be used to construct objects.

Abstract methods don't have bodies and can only be used in abstract classes. The subclass (inherited from) provides the body.

- Encapsulation

Encapsulation is the binding, or wrapping, of data and code into a single unit. A Java class serves as an illustration of encapsulation. Since every member of the data is private in this case, the Java bean is the fully encapsulated class.

Advantages and disadvantages in OOP

Advantages	Disadvantages
The coding is easier to maintain.	It can cause duplication.
The modular classes are often reusable.	It can be too scalable.
It allows for parallel development.	It can be inefficient.

Figure 3 advantages and disadvantages

Task 02.

2.1 UML diagrams

Software system artefacts can be specified, visualised, built, and documented using the UML standard language.

⁷ The Object Management Group (OMG) developed UML, and in January 1997, the OMG received a draft of the UML 1.0 definition.

⁷ OMG is working tirelessly to establish a true industry standard.

³ Unified Modelling Language is known as UML.

Compared to other popular programming languages like C++, Java, COBOL, etc., UML is distinct.

A visual language called UML is used to create software blueprints.

A general-purpose visual modelling language for software system specification, construction, and documentation is called UML.

UML is not constrained by the fact that software systems are typically modelled using it. It is also employed in the modelling of non-software systems. Consider the manufacturing unit's process flow, etc.

⁸ It is possible to think of UML as the evolution of object-oriented (OO) analysis and design.

Data and methods for controlling the data are both contained in an object. The object's state is reflected in the data. In addition to describing an object, classes create a hierarchy that mimics the structure of real-world systems. Inheritance is used to illustrate the hierarchy, and ⁵ classes can also be connected in many ways depending on the situation.

Real-world items are called objects, and UML may be used to represent fundamental ideas like inheritance, polymorphism, abstraction, and encapsulation.

⁵ All of the concepts used in object-oriented analysis and design can be represented using UML thanks to its strength. ⁵ UML diagrams alone depict concepts that are object-oriented. Thus, it's crucial to understand this before studying UML.

2.1.1 types of UML diagrams

UML diagrams are divided into two subcategories: behavioral and structural diagrams.

Structural diagrams The elements that comprise a system and their interrelationships are shown in structural diagrams. These schematics display a system's static components.

Behavioral diagrams show the actions that take place inside a system. They display the interactions between all the parts and other users or systems.

Structural Diagram

A essential component of any object-oriented solution is a UML class diagram. It shows an object-oriented, static system with projects defined by classes, attributes, and functions.

Stated differently, it illustrates the various classes present in a system and how they function.

This interaction diagram is used by business managers and software engineers to represent the various relationships that are present in a process.

A rectangle in the diagram represents the class. Every rectangle has three vertically divided pieces. The class name appears in the top area, and information on the class's behaviors, operations, and characteristics may be found in the second and third sections.

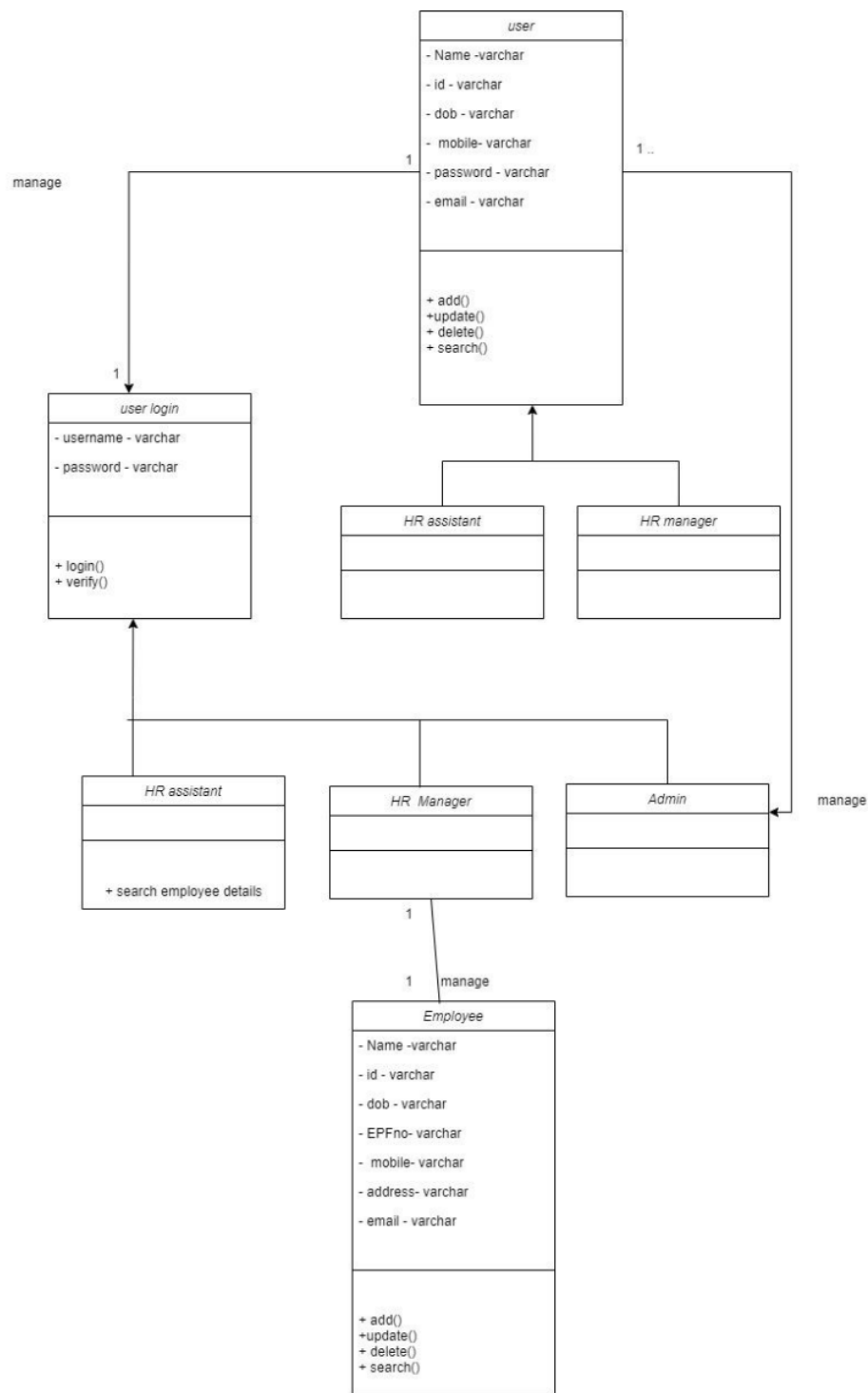


Figure 4 . class diagram

- Explanation

User:

Represents a generic user of the system.

Subclasses include HR Manager, HR Assistant, and potentially other user types.

Admin:

Use case: createAccount() - Permits the administrator to establish an account for the HR Manager and HR Assistant.

HR Manager:

Use cases:

addDepartments() - Enables the HR Manager to add new departments.

addDesignations() - Allows the HR Manager to add new designations.

addEmployees() - Lets the HR Manager add new employees and allocate them to available departments and designations.

searchEmployees() - gives the HR Manager the ability to look for employee information using a variety of parameters, including name, department, EPF number, and designation.

HR Assistant:

Use cases ;

addEmployees() - Allows the HR Assistant to add new employees and allocate them to available departments and designations.

searchEmployees() - Enables the HR Assistant to search for employee details based on various criteria such as department, designation, name, EPF number, etc.

Employee:

Represents the details of an employee.

- Associations:

The User class has associations with Admin and HR Manager, representing the relationships between different user types.

HR Manager is associated with Employee, representing the management of employee-related functionalities.

Behavioral diagrams

i. Use case diagram

Use case diagrams give a visual summary of the key players in a software system. They assist developers in analyzing the connections between use cases and personas by providing an overview of a system's intended behavior and displaying its capabilities.

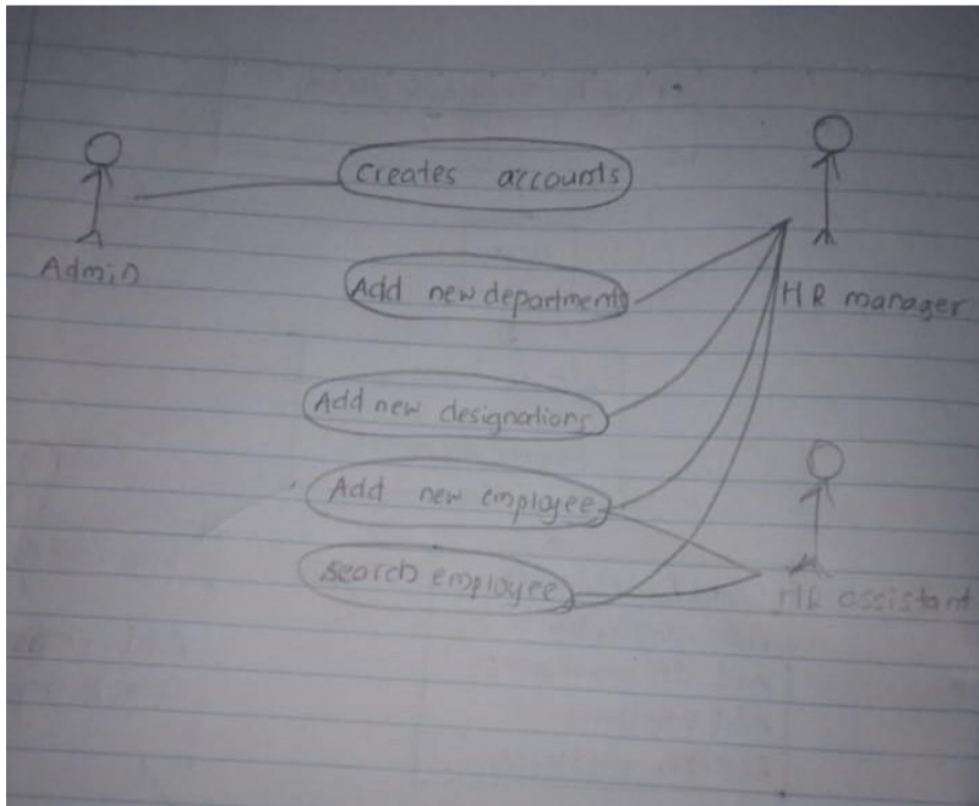


Figure 5. use case diagram

Classes:

user: Represents common attributes like name, employee ID, and contact details.

Employee: Inherits from Person, includes attributes like department, designation, and methods for searching.

HRManager: Inherits from Employee, includes methods for managing departments and designations.

Admin: Manages user accounts, inherits from Person.

System: Manages the overall system and file operations.

Use Case 1: Add New Department

Actors:

HR Manager

Description:

6 HRManager logs into the system.

HRManager navigates to the "Manage Departments" section.

1 HR Manager selects the option to "Add New Department."

The system prompts HR Manager to enter the department details (e.g., name, description).

HR Manager submits the information.

The system validates and adds the new department to the database.

The system confirms the successful addition of the department.

9
Alternate Flow:

If the entered department information is incomplete or invalid, the system prompts HR Manager to provide valid details.

Use Case 3: Add New Employee

Actors:

HR Manager

Description:

6 HR Manager logs into the system.

HR Manager navigates to the "Add New Employee" section.

HR Manager fills in the employee details (e.g., name, contact information, department, designation).

12 HR Manager submits the information.

The system validates the entered details.

If valid, the system adds the new employee to the database.

The system generates an employee ID and provides it to HR Manager.

The system confirms the successful addition of the employee.

9
Alternate Flow:

If the entered employee information is incomplete or invalid, the system prompts HR Manager to provide valid details.

Use Case 4: Search Employee Details

Actors:

HR Manager

Description:

6
HR Manager logs into the system.

HR Manager navigates to the "Search Employee" section.

HR Manager selects the search criteria (e.g., department, designation, name, EPF number).

1
HR Manager enters the search parameters.

The system retrieves and displays the matching employee details.

HR Manager can view, edit, or take further actions based on the displayed information.

Alternate Flow:

If there are no matching records, the system notifies HR Manager accordingly.

ii. Sequence diagram

Sequence diagrams (sometimes called event diagrams) in UML illustrate the sequence in which your items interact. This comprises the messages that are sent back and forth between your objects to carry out certain tasks, as well as the lifelines of your objects and the processes that engage with them.

These diagrams are frequently used by developers and business experts to comprehend how to organize a new system or enhance an existing procedure.

Employee

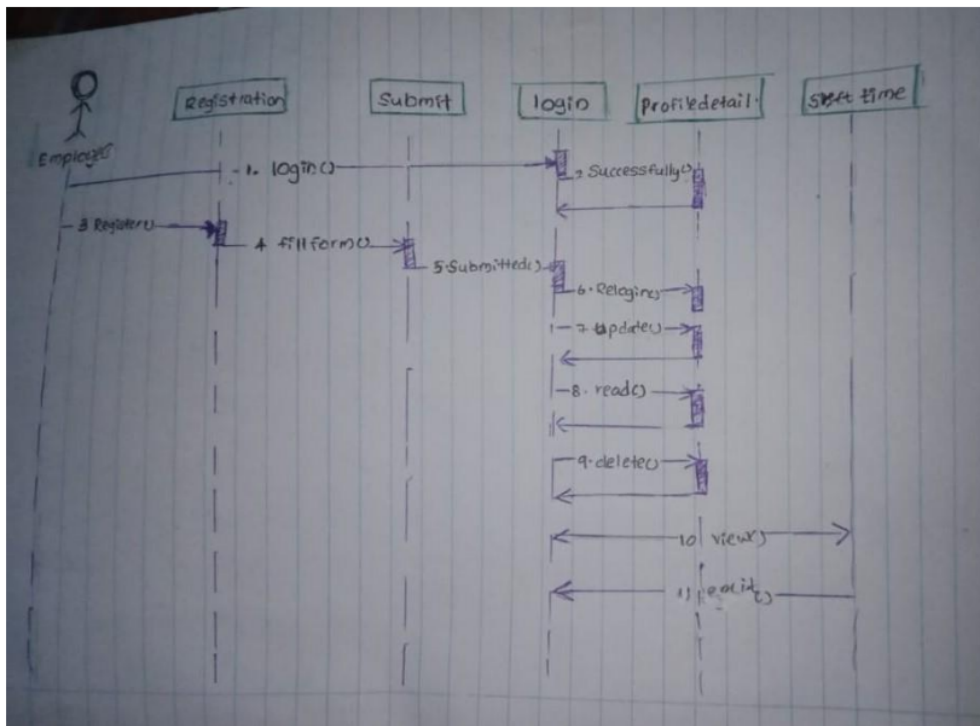


Figure 6 sequence diagram employee

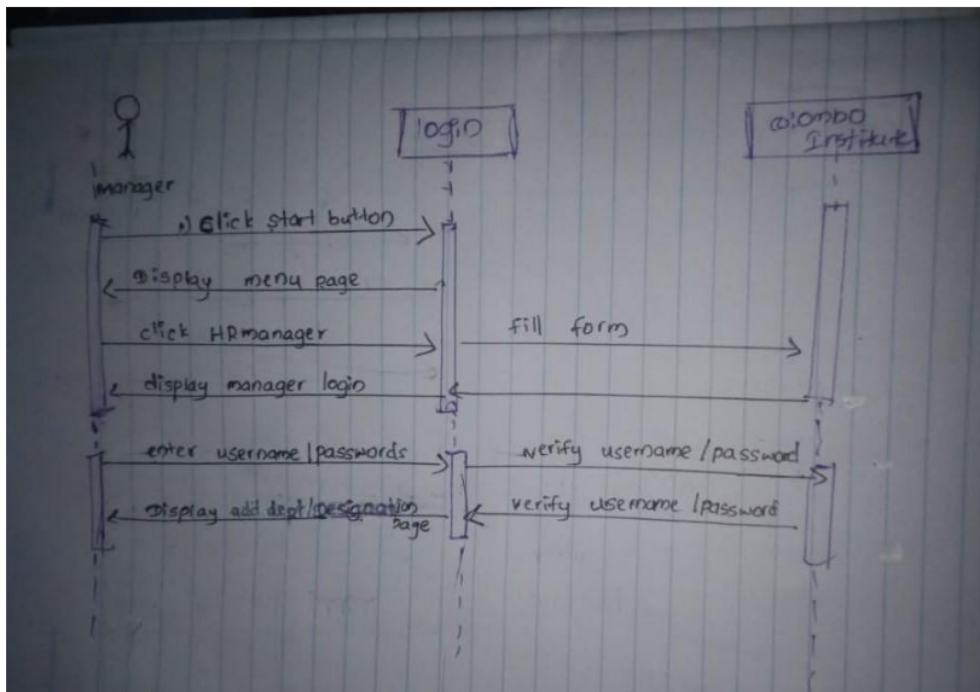


Figure 7. sequence diagram manager

HR Assistant

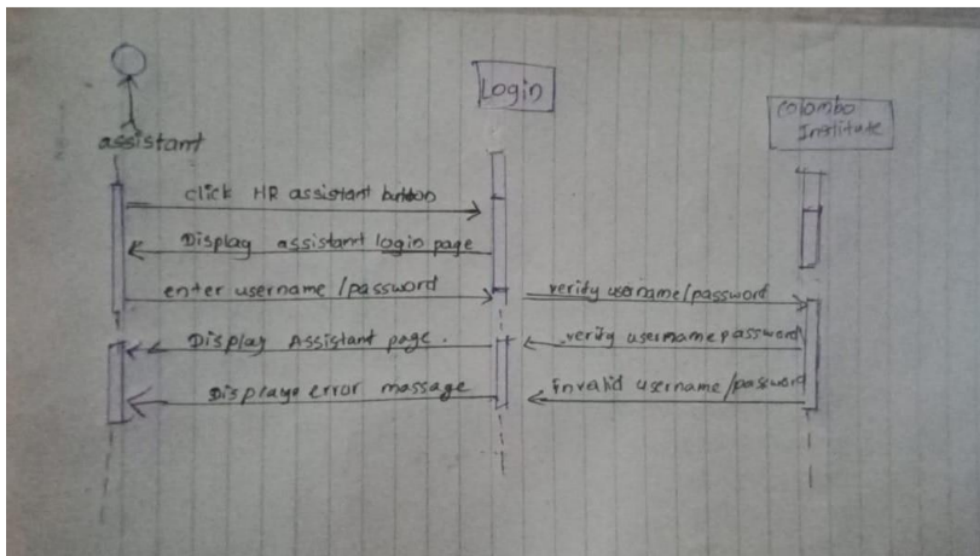


Figure 8 sequence diagram assistant

Task 03.

User guide of Colombo Institute of Studies

© Colombo Institute Of Studies 2023

Introduction

Welcome to the "Colombo Institute of Studies Employee Management System" User Guide. The purpose of this tutorial is to help HR managers and administrators use the newly built system's capabilities and navigate it. The system's goal is to automate and expedite the process of maintaining employee data, allowing HR staff to work more effectively on a variety of duties.

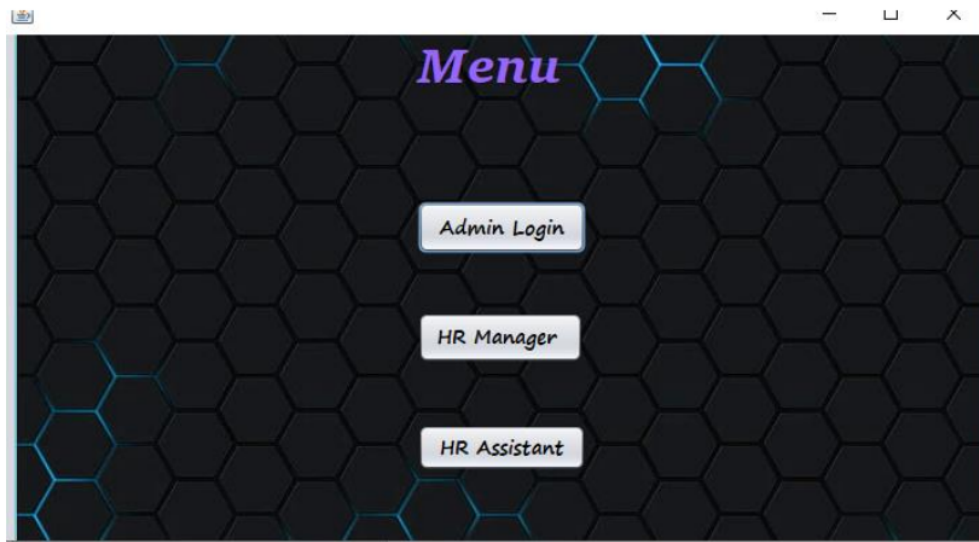
About the System

The "Colombo Institute of Studies Employee Management System" is a user-friendly and easy-to-use program designed to make the duties of the HR department easier. HR managers may effortlessly add and manage departments, designations, workers, and user accounts with the system. To provide safe and restricted access to the system, administrators can make new accounts for HR Managers and HR Assistants.

UI of start page



UI of 2nd page



1. Adding a New Department:

Log in as HR Manager.

Select the "Add Department" option.

Enter the department name and submit.

2. Adding a New Designation:

Log in as HR Manager.

Select the "Add Designation" option.

Enter the designation name and submit.

The screenshot displays a light gray background with two distinct form sections. The top section, titled 'Add New Department' in purple italicized text, features a label 'Department Name' in bold black font next to a white text input field. A 'SUBMIT' button is positioned to the right of the input field. The bottom section, titled 'Add New Designation' in pink italicized text, features a label 'Designation Name' in bold black font next to a white text input field. A 'SUBMIT' button is positioned to the right of the input field.

3. Adding a New Employee:

Log in as HR Manager.

Select the "Add Employee" option.

Enter employee details (name, ID, contact, department, designation) and submit.



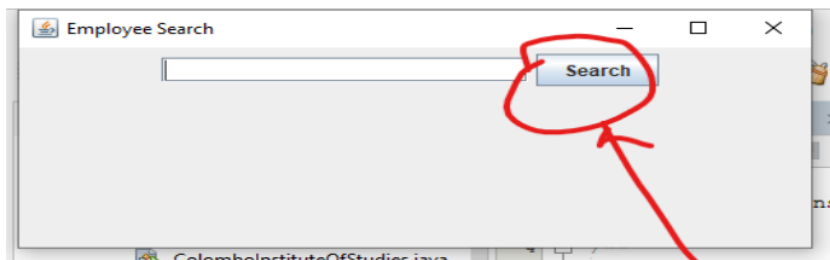
A screenshot of a Java Swing window titled "New Employee Registration". The window has a light gray background. At the top left, the title "New Employee Registration" is written in a purple, italicized font. To the right of the title is a small icon of a person with a plus sign. Further right are two buttons: "EXIT" and "NEXT". Below the title, there are seven text input fields arranged vertically, each preceded by a label: "Full Name", "NIC/ Passport No", "Date Of Birth", "Gender", "Mobile Number", "Address", and "Email". At the bottom right of the window is a "SAVE" button. The window is shown on a desktop with other Java Swing windows visible in the background, including one titled "m_menu.java".

4. Searching for Employee Details:

Log in as HR Manager.

Select the "Search Employee" option.

Enter search criteria (department, designation, name, employee ID, etc.) and submit.



A screenshot of a Java Swing window titled "Employee Search". The window has a light gray background. At the top left, the title "Employee Search" is written in a standard font. Below the title is a text input field. To the right of the input field is a button labeled "Search". A red circle is drawn around the "Search" button, and a red arrow points to it from the bottom right. The window is shown on a desktop with other Java Swing windows visible in the background, including one titled "ColomboInstituteOfStudies.java".

5. Creating a New Account:

Log in as Admin.

Select the "Create Account" option.

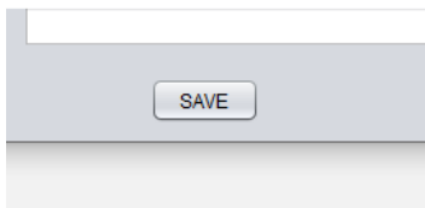
Choose the user type (HR Manager or HR Assistant).

Enter user details and submit.



A screenshot of a Java Swing window titled "New HR Manager". The window has a light gray background and a title bar with standard Windows controls (minimize, maximize, close). In the top right corner, there is an "EXIT" button. The form contains five labeled text input fields: "Username", "ID", "Password", "Email", and "Phone". At the bottom right of the form, there are two buttons: "SAVE" and "NEXT". The window is shown on a desktop with a taskbar at the bottom containing icons for "JAVA OOP" and "Email".

Same UI for HR assistant



A close-up view of a "SAVE" button. The button is rectangular with rounded corners, a light gray background, and a thin gray border. It is positioned on a dark gray horizontal bar.

Note: Ensure to save data regularly using the "Save" option to prevent data loss. Data save in a file .

Thank you .

ORIGINALITY REPORT

25%

SIMILARITY INDEX

3%

INTERNET SOURCES

0%

PUBLICATIONS

24%

STUDENT PAPERS

PRIMARY SOURCES

1

Submitted to University of Wales Institute,
Cardiff

Student Paper

8%

2

Submitted to Asia Pacific University College of
Technology and Innovation (UCTI)

Student Paper

3%

3

Submitted to St. Ignatius High School

Student Paper

2%

4

www.coursehero.com

Internet Source

2%

5

Submitted to Florida Institute of Technology

Student Paper

2%

6

Submitted to Texas A&M University -
Commerce

Student Paper

2%

7

Submitted to Unicaf University

Student Paper

1%

8

Submitted to CTI Education Group

Student Paper

1%

9

Submitted to American University in Cairo

Student Paper

1 %

10

Submitted to Info Myanmar College

Student Paper

1 %

11

www.freecodecamp.org

Internet Source

1 %

12

Submitted to Northern Caribbean University

Student Paper

<1 %

Exclude quotes On

Exclude matches Off

Exclude bibliography On