

Entity

Interface	Userable	User
<p>Responsibilities</p> <ul style="list-style-type: none">• Defines abstract methods for classes that implement this interface.• To break the dependency of AccountManager on User<ul style="list-style-type: none">○ Designed with Dependency Inversion Principle	<p>Collaborators</p> <ul style="list-style-type: none">• AccountManager	

User

Responsibilities

- Stores the name, id, account information, password, address, phone number of the employee
- Getters for everything above
- Setters for everything above
- Equals based on the id of the user
- Prints the user’s identity and name

Collaborators

- AccountManager

Abstract

Employee

**FullTimeEmployee,
PartTimeEmployee**

Responsibilities

- Stores the wage, attendance, department, and level of the employee
- Getters for everything above
- Setters for everything above
- Prints the department of the employee

Collaborators

- AccountManager

FullTimeEmployee

Responsibilities

- Stores the total vacation with salaries and the vacation used that the full-time employee has
- Getters for everything above
- Setters for everything above
- Prints the information of the full time employees' department, position, and level in strings.

Collaborators

Employee

PartTimeEmployee

Responsibilities

- Stores the schedule of the part-time employee
- Getter for schedule above
- Setter for schedule above
- Prints the information of the part time employees' department, position, and level in strings.

Collaborators

Interface

Work

Responsibilities

- Stores the work information, work requirements
- Getter method for the work information and work requirements
- Setter method for the work information and work requirements

Collaborators

- WorkManager
- WorkDistributor

Group

Responsibilities

- Stores a group's information, including group members, group leader, and project.
- Getter method for the above information.
- Setter method for the above information.

Collaborators

- Userable
- Work
- WorkDistributor
- WorkManager

Uses Case

AccountManager

- Stores a Hashmap called **employee map** that include the userable as the key and employee as the value.
- Includes a Counter for recording the IDs that have been issued
- Stores and count the total number of employees
- Stores and count the total number of part-time employees
- Stores and count the total number of full-time employees
- Overrides the toString method (i.e Accountmanager)
- Creates new user
- Deletes user
- Gets total number of employees
- Gets total number of part-time employees
- Gets total number of full-time employees

- Userable
- Employee
- Verifier

Verifier

- Stores a Hashmap called **employee map** that include the Account number (which is String type) as the key and Userable as the value.
- Verifies whether the account exists
- Verifies whether the account has the level (i.e: authority)
- Prints “this is the verifier”
- Tells the Admin_System that the login is successful or not

- AccountManager
- Admin_System
- AuthorityManager

AccountChanger

Responsibilities

- Modifies the information in user and employee account.

Collaborators

- Userable
- Employee
- Admin_System

AuthorityManager

Responsibilities

- Performs authority (level) -related tasks.

Collaborators

- Admin_System
- Verifier

PayManager

Responsibilities

- Calculates the wage of the employees.
- Gets the particular employee its salary in specific months.

Collaborators

- Admin_System
- AccountManager

VacationManager

Responsibilities

- Manages employees' vacation.
- Modifies the status of the employee
- Gets the particular employee its rest number of vacation.

Collaborators

- AccountManager
- Admin_System

WorkManager

Responsibilities

- Manages the work information.
- Stores the progress of the unfinished work.
- Records the group information of the finished project.

Collaborators

- Work
- Group
- Admin_System

WorkDistributor

Responsibilities

- Assigns new work to the group
- Creates a new group

Collaborators

- Work
- Group
- Admin_System

Interface Adapter

Admin_System

Responsibilities

- Stores all the manager classes (I.e: Account_Manager, Pay_Manager)
- Controller for the manager to operate on the use cases
- Receives input from Command_User_Interface
- Calls Verifier
- Tells the Command_User_Interface whether the login is successful or not
- Sends the information from the Command_User_Interface to AccountManager

Collaborators

- Command_User_Interface
- AccountManager
- Pay_Manager
- Verifier
- WorkDistributor
- WorkManager

Framework

Command_User_Interface

Responsibilities

- Provides a command line interface to user for accessing the systems.
- Sends information that user input to the Admin_System

Collaborators

Admin_System
Pay_Manager