Human Resource Management System

Prepared by:

Omar Mamdouh Zaki-fathy Youssef

144255

Graduation Project Proposal

Submitted in partial satisfaction of the requirements for the degree of

Management Information System

In the

Faculty of Management Sciences

Of

October University for modern sciences & arts

Supervised by: **Dr. Adel Ghannam**

Fall 2017

Acknowledgment

This project will not be possible without the help of my professor. I would like to express my gratitude to my supervisor Prof. Mohamed Adel Ghannam for his invaluable guidance and assistance throughout the whole project and devoting much of his time to supervise us in our project, without which the accomplishment of the task would have never been possible. Thanks for being a good mentor and for guiding me on the right path. I will forever be grateful for your guidance and kindness.

I am also thankful to my teaching assistance Eman Osama for providing me with the relevant information and necessary clarifications required for the graduation project. Thanks for your valuable support!

Table of Contents

1.0 Executive summary

This project is done as a final year graduation project for Bachelors of Management Information System (MIS) offered by Modern Sciences and Arts (MSA) University, Egypt in which this project is undertaken to plan, design and develop a Human Resource Management system.

Behind production of every product or service there is a human mind, effort and man hours (working hours) in which no product or service can be produced without help of human being for which human being is the fundamental resource for making or construction of anything. Today many experts claim that machines and technology are replacing human resource and minimizing their role or effort. However, indeed, machines and technology are built by the humans; they need to be operated or at least monitored by humans. Maybe because of this reason, companies have continuously been searching for talented, skilled and qualified professionals to hire or improving their employees' skills and knowledge which is the role of **human resource management** for further development of machines and technology, which again have to be controlled or Monitored by humans to bring out products/services.

The **human resources department** within any organization is considered to be highly critical for the entire organization. Its many functions serve as a supportive background for the company by providing everything from skilled and talented labour to management training services, employee enrichment opportunities and more. Since **labour** is the single largest expense for most organizations, human resources helps companies derive the **greatest value** from this important **asset**.

In order to function optimally, however, human resources departments must have the right **tools** and **resources** in place. A **human resources information system**, or **HRIS**, is a type of software program that can be utilized within the department to help human resources employees and managers improve their productivity and the results of their efforts.

2.0 Scope of the project and comparison with two systems in the market

2.1 Project Scope Statement

Date: March 28, 2017

From: Omar Mamdouh Zaki

Subject: Human Resource Management System

Version	Date	Author	Description
1.0	February, 28, 2017	Omar Mamdouh	Draft initiated
2.0	March, 8, 2017	Omar Mamdouh	Editing the functional requirements and the project boundaries
3.0	March, 15, 2017	Omar Mamdouh	Editing the functional requirements and the project boundaries
4.0	March, 28, 2017	Omar Mamdouh	Editing the project constraints, project assumptions and adding removing some functional requirements

Project Overview

Human Resource Management System

A Human Resource Management System (HRMS) is a form of HR software that combines a number of systems and processes to ensure the easy management of a business's employees and data. These systems could deal with everything from payroll to performance evaluation, covering the whole business. Human Resources Management System is used by businesses to combine a number of necessary HR functions such as storing employee data, managing payrolls, recruitment processes, benefits administration and keeping track of attendance records. It ensures everyday Human Resources processes are manageable and easy to access.

Project Objectives

Objectives of this project are improving and easing the process of human resource management by providing advanced features such as request features, payroll management features, organizational management features, workforce management features, time management features, personnel management features, benefit administration feature, and other features to reduce time & cost, help to retain good employees, less cost instead of buying expensive external software for HR or paying monthly fees for it, and helping the business to collaborate and interact between HR department and other departments and employees through the HR system by providing enhanced sharing of information and collaboration.

Product Scope Description

The software will run on PCs running Windows 10 and will support all the company's employees specially the human resource management and financial management departments.

Feature (PB)	Description	MH	Min MW	Max MW	COM	COR	FEA	NEC	TRA	VER
1.0	Employee attendance management	■								
1.1	Clock in & clock out	■								
2.0	Recruitment	■								
2.1	Hiring new employees	■								
3.0	Employee appraisal entry	■								
4.0	Employee salary deduction entry	■								
5.0	Employee penalties entry	■								
6.0	Employee	■								

	incentives entry									
7.0	**Employee salary increase entry**	■								
8.0	**Employee achievements entry**	■								
9.0	**Payroll management**	■								
9.1	**Ability to view monthly salary by employee**	■								
10.0	**Generating management report**	■								
11.0	**Design training for employee**	■								
11.1	**Ability to view scheduled trainings**	■								
12.0	**Update employee job data**	■								

PB=Product Backlog Item, **MH**=Must Have (otherwise, Nice to Have)
COM=Complete,**COR**=Correct,**FEA**=Feasible,**NEC**=Necessary,**TRA**=Traceable,V**ER**=Verifiable.

Project Boundaries

Human Resources Management System is used by businesses to combine a number of necessary HR functions such as storing employee data, managing payrolls, recruitment processes, benefits administration and keeping track of attendance records. It ensures everyday Human Resources processes are manageable and easy to access.

The system didn't cover **creating career succession planning for employees** (Succession Planning is one of those areas which can make a big difference inside an organization; put the right employees into the appropriate career paths), **improving the process of recruitment**

(how the applicants apply for a job or a role and how the HR filter applicants in order to hire the best new employee for a specific role or a job in which online assessment such as leadership assessment, problem solving skills assessment, corporate communication skills assessment, technical assessment, and other types of assessments will be also provided in order to filter applicants), **requesting vacations** from the system & receive response, **shift planning**, **travel management** (booking flight tickets from the system & receive response and booking hotel reservation from the system & receive response), **storing data about competitors' employees**, **requesting school coverage** and **making a daily online newspaper** in the HR system to retain employees.

Project Deliverables

Project deliverables and functional requirements are Employee attendance management, Recruitment & hiring new employees, Employee appraisal entry, Employee salary deduction entry, Employee penalties entry, Employee incentives entry, Employee salary increase entry, Employee achievements entry, Payroll management, Generating management report, Design training for employee, and Update employee job data.

Nonfunctional requirements are performance requirements in which production of a simple report shall take less than 30 seconds for 90% of the cases, security in which the system shall identify all of its employees before allowing them to use its capabilities, installation of a new version shall leave all database contents and all personal settings unchanged, and the estimated loss of data in case of a disk crash shall be less than 0.01%.

Product Acceptance Criteria

Each new feature is expected to improve the business execution which **improve business efficiency and effectiveness** by providing time management, making requests from the system, automatic salary increase, and others features, **increase profit and decrease cost** by providing training and event management (scheduling seminars, courses and business events for each employee), monitor & track the performance of employees and creating career & succession planning for each employee to improve employees knowledge, skills and experience in order to increase company's profit and reducing company's cost, **support strategic goal and reduce paper waste** by many other features stated before.

Each functional requirement stated in the project deliverables does its work correctly without any mistakes.

Project Constraints

The main constraint faces the project is access control that help us restrict whom and what accesses our information resources, and they possess four general functions such as identifying verification, authentication, authorization, and accountability.

Time of implementing is another constraint that prevented applying some advanced features which is nice to have such as **creating career succession planning for employees** (Succession Planning is one of those areas which can make a big difference inside an organization; put the right employees into the appropriate career paths), **improving the process of recruitment** (how the applicants apply for a job or a role and how the HR filter applicants in order to hire the best new employee for a specific role or a job in which online assessment such as leadership assessment, problem solving skills assessment, corporate communication skills assessment, technical assessment, and other types of assessments will be also provided in order to filter applicants), **requesting vacations** from the system & receive response, **shift planning**, **travel management** (booking flight tickets from the system & receive response and booking hotel reservation from the system & receive response), **storing data about competitors' employees**, **requesting school coverage** and **making a daily online newspaper** in the HR system to retain employees.

Project Assumptions

Human Resource Management System should provide the capability to more effectively plan, control and manage HR costs, achieve improved efficiency and quality in HR decision making, and improve employee and managerial productivity and effectiveness.

Identified Risks

Key risks for this project are:

Difficulty of implementing some advanced features

Time of implementing some advanced features

Schedule Milestones

Basic schedule range is expected to be:

Technical Feasibility	Dec 15-Jan 1
Release	Jan 22

Cost Estimate

Costs estimate cannot be identified or known yet

Approval Requirements

The project should be approved by the upper management (Company's CEO) to approve the whole project, human resource management department (Chief HR Officer, HR Director) to approve the HR features needed, financial management department (Chief Financial Officer, Financial Director) to approve the cost of the project and the HR features needed by the financial management department, and management information system department (Chief MIS Officer, MIS Director) to approve the technicalities, time and other issues regarding the project plan, design & implementation.

The requirement approval is based in the final traceability matrix shown in section (…).

Acceptance

Submitter's signature	Sponsor's signature
Omar Mamdouh Zaki	[Sponsor's name]
Human Resource Management System	[Sponsor's title]
28/1/2018	Date accepted

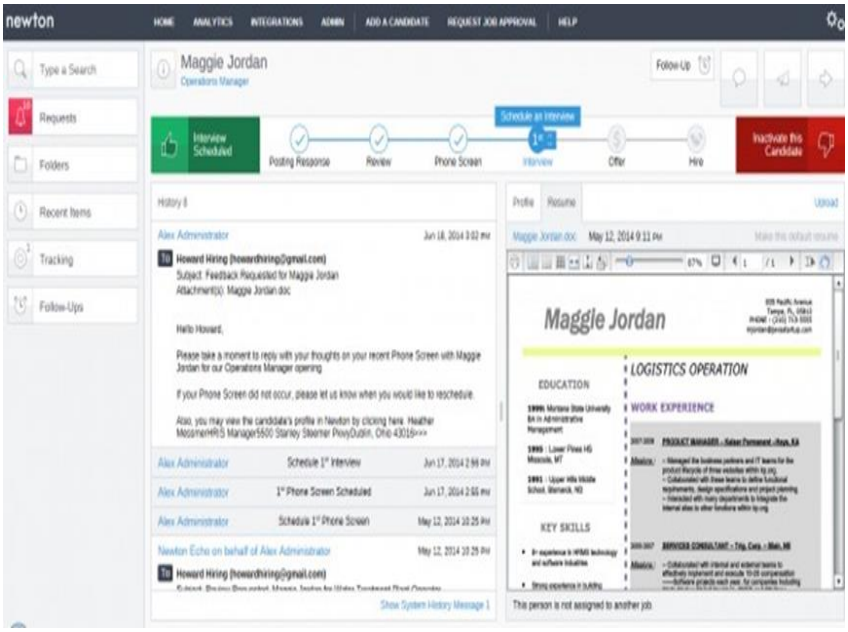
2.2 Features of two similar HR packages

2.2.1 Paycor HR package

First, **Paycor**'s Perform is an integrated cloud-based software suite that offers functionality to encompass the entire employee lifecycle - from recruiting and on boarding to time and attendance management. Collectively, the solution offers tools for managing recruitment, hiring, payroll, on boarding, tax compliance, time& attendance, benefits and compliance reporting activities.

With Paycor Applicant Tracking System, businesses can streamline their recruiting activities. Recruiters can track prominent candidates, track communications, schedule interview rounds, initiate on boarding and more. HR teams can manage payroll process, process tax deductions, file compliance requirement and distribute benefits and more.

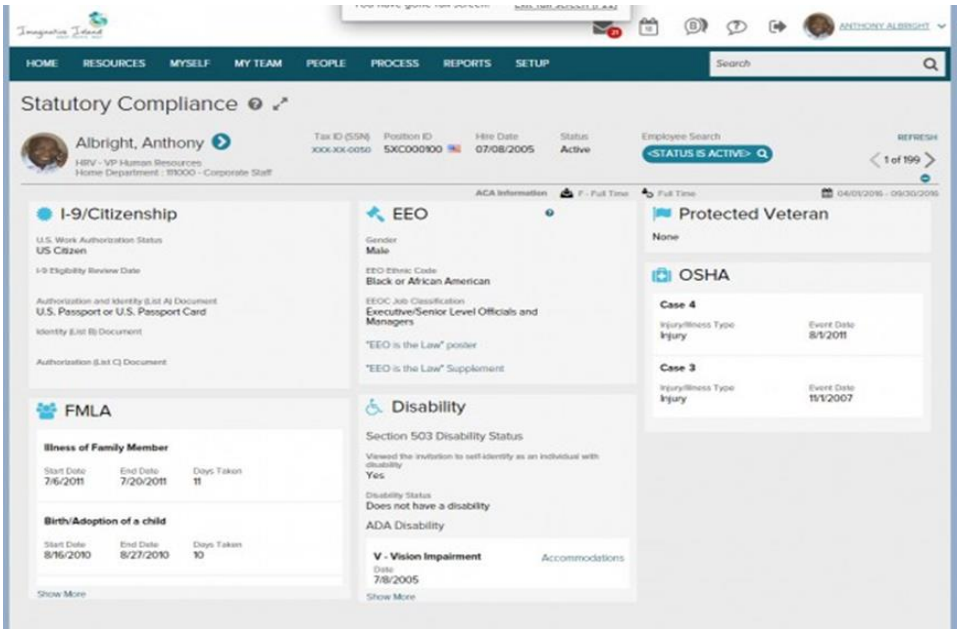
Paycor also offers time & attendance tracking feature, which helps businesses to automate tracking employee hours. HR teams can further integrate employee payroll information and their actual working hours, which help businesses in automating their payroll distribution process.



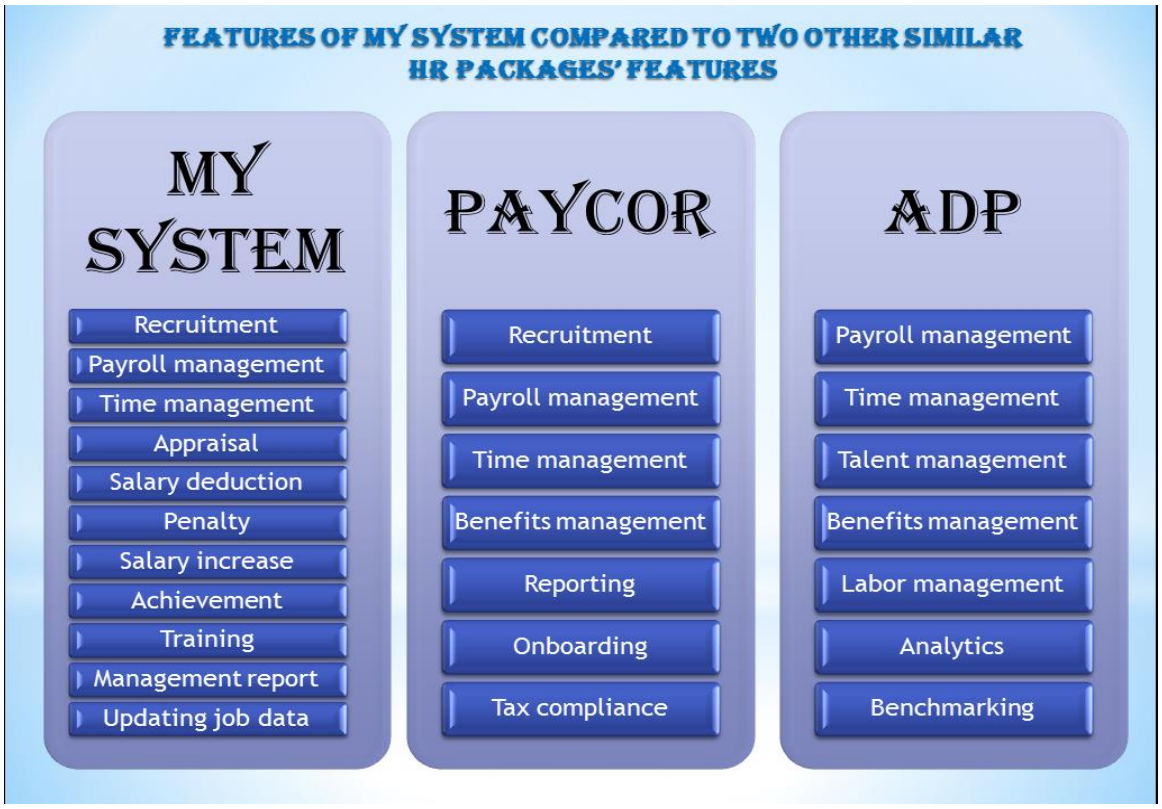
2.2.2 ADP HR package

Second, ADP Workforce Now is a human capital management solution for companies with more than 50 employees. The core capabilities of the system include human resources management, payroll, benefits, talent management, compliance, time and labour management and analytics and benchmarking.

The fully integrated suite is built on a single database, which automates data synchronization across different components and helps users to minimize manual administrative tasks. The platform also integrates with other HR and business applications via the ADP Marketplace and APIs.



2.2.3 My HR software's features compared to the other two HR packages



The above chart shows the comparison between my HR software features and the other two HR packages features. Therefore, the comparison shows that the features are similar but maybe in different names in which there are little differences between the packages' features.

3.0 Objectives of the Graduation project

The aim of this Graduation project is to give the MIS-major student, of the faculty of management science, hands on experience on how technology can best be used by businesses. Each student identifies a problem or an opportunity in a business domain. The student defines the scope and explains his proposed solution. The focus is then on practicing the SDLC through a detailed system design using DFD to model the business processes, the ERD, and the Use-case model is then developed to show the services expected from the system for the business actors. If the student uses OOAD he/she will show traceability down to coding. The student can target desktop application; however, he can develop web application. In this later case he should show how Web forms are designed and the programs

logic behind the forms are developed using the Visual Studio express 2012 for web applications , or any other web application development tool .

In all cases, he/she should design, run, and document Integration tests. These tests are described and demonstrated through running test scenarios under an Explorer environment. A traceability matrix is also developed to show the alignment of all the SDLC stages to the original defined scope. Finally, proposals are given on how to measure expected business benefits from a life system.

4.0 Development Methodology

In this section, we generally describe the methodological approach that we have followed to ensure correct building of a good quality-based business information system project that complies with the standards of the software engineering rules that are acknowledged all over the world. There are two main approaches of building information system software which is the traditional system analysis and design based on DFD" since the 70's of the last 20th century [1]". On the other hand, we have the new approach "Object Oriented Analysis and Design -OOAD "since 1990's by Booch [2]. The OOAD Leads to direct link between the transaction services expected from the system Use case model & (Reusable components (class diagram), as well the dynamic behaviour of the operations to execute (sequence diagram)

Considering the above, we have followed the object-oriented analysis and design approach which addresses the technical approach of analysing, and designing an information system software application throughout the system development life. Moreover, the object-oriented analysis and design approach encompasses the following stages: Requirements definition, logical design of the system, physical design of the system and implementation, and verification and maintenance. These stages are usually called the "Waterfall" model. The stages are created and accomplished by models starting with the **Use Case** which describes the main services/features of the domain that the system must accomplish to serve the users, **Class diagram** which describes the main structure of the system through objects that contains attributes that identifies each object and the operations needed to fulfil the system output results (we should consider that the class diagram is considered as the main building block of any information system application that simulates real-life events and how these

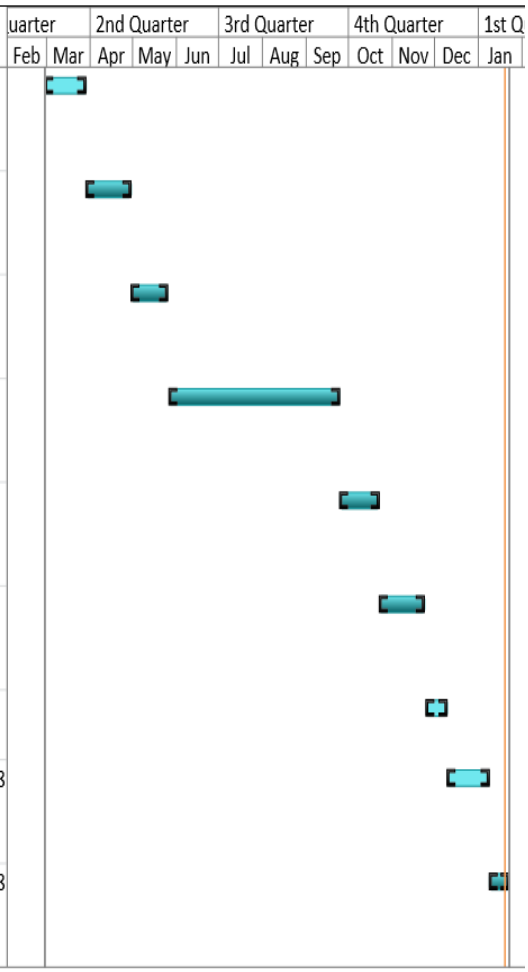
events can be achieved), **Sequence diagram** which describes the dynamic behaviour of logical sequencing of achieving the events of the system through the interaction of objects with each other through sent and received messages containing the operations needed to fulfil an event, and **Activity Diagram** which describes the main activities needed for achieving the computational processes of the system ,presented in **screen-formats** and **programs logic** of the system using the **classes operations** according to the **sequence diagram design**.

The Object Management Group (www.omg.org), a non-profit technology standards consortium, has overseen the definition and maintenance of UML specifications for OOAD. This allows engineers and programmers the ability to use one language for many purposes, during all phases of the software lifecycle, and for all system sizes

In this project we have also used the traditional Context and 0-level Data Flow Diagram (DFD), as leverage to better understand the transactions **of the Use Case model** as well as the data stores to better specify **the** Entity **relationship diagram (ERD)** which describes the main information structure that the system needs in order to achieve the business processes. This gives the project reviewer a good perspective for the business case operation processes.

5.0 Project Plan (Gantt Chartt)

ID	Task Mode	Task Name	Duration	Start	Finish
1		scope definition	21 days	Tue 28-02-17	Tue 28-03-17
2		scope review&domainapplications in market	24 days	Wed 29-03-17	Sat 29-04-17
3		use case model	20 days	Sun 30-04-17	Thu 25-05-17
4		semester exam & summer&presentationfor review	87 days	Sat 27-05-17	Sun 24-09-17
5		class model	21 days	Mon 25-09-17	Sun 22-10-17
6		sequence model	24 days	Mon 23-10-17	Thu 23-11-17
7		verification and validation testing	12 days	Sat 25-11-17	Sat 09-12-17
8		activity diagram &prototype	22 days	Sun 10-12-17	Mon 08-01-18
9		documentation	10 days	Tue 09-01-18	Mon 22-01-18



Project: plan
Date: Sat 20-01-18

Task		External Milestone		Manual Summary Rollup	
Split		Inactive Task		Manual Summary	
Milestone		Inactive Milestone		Start-only	
Summary		Inactive Summary		Finish-only	
Project Summary		Manual Task		Deadline	
External Tasks		Duration-only		Progress	

6.0 Theoretical background

6.1 Introduction

Generally speaking, a **Human Resource Management System** (HRIS) is a form of HR software that combines a number of systems and processes to ensure the easy management of a business' employees and data. Human Resources Software is used by businesses to combine a number of necessary HR functions, such as storing employee data, managing payrolls, recruitment processes, benefits administration and keeping track of attendance records for which it ensures everyday Human Resources processes are manageable and easy to access.

While the **Human Resource Management System** features benefit the organization in many ways, one of the most important of all HRIS benefits relates to the ability of the software program to improve **productivity** of human resources employees in which these systems are highly detailed, and they are designed to enhance and speed up the efforts of HR employees in a number of ways. For example, they can assist with ***recruitment*** by simplifying the process of collecting resumes, reviewing candidate information and more.

HRIS systems can also be used to improve **productivity** related to financial management and other things through ***payroll*** , ***attendance***, ***appraisal***, ***salary deductions***, ***penalties***, ***salary increases***, ***trainings*** and ***management reports*** processing tasks in which these and other related tasks may require numerous hours of manpower each week. However, the time and effort required to complete them can be drastically reduced when some of the tasks are automated through a HRIS system for which the tasks that may have required many hours of labour may possibly reach completion very quickly and easily or sometimes even done automatically with the software program.

Finally, a **Human Resources Management System** (HRMS) is a type of information system (IS) that is designed to manage an organization's computerized and automated human resource (HR) processes in which it is a combination of hardware and software resources that hosts and provides most, if not all, of a HR department's business logic.

6.2 Brief History

Generally speaking, **Human resource management** (HRM) is the management of human resources. Commonly referred to as the HR Department in which it is designed to maximize employee performance in service of an employer's strategic objectives for which HR is primarily concerned with the management of people within organizations, focusing on policies and on systems. HR departments are responsible for overseeing employee-benefits design, employee recruitment, training and development, performance appraisal, and rewarding (e.g., managing pay and benefit systems), etc.

According to Almutawa, Z., Muenjohn, N., & Jiaying, Z. (2016, human resource is the fundamental resource for making or construction of anything that help the company to accomplish a desired competitive advantage. Today many experts claim that machines and technology are replacing human resource and minimizing their role or effort. However, indeed, machines and technology are built by the humans; they need to be operated or at least monitored by humans.

Peretz H. (2013) stated that Human Resource Information Systems, or HRIS, have drastically evolved since they were first introduced more than 50 years ago. When HRIS solutions were originally developed, they served as a computerized database for storing records. These systems are now capable of handling most of the complex yet repetitive tasks that fall to the human resources department. HRIS solutions improve efficiency, and save both space and valuable human resource hours for more important and humanized tasks.

Ferguson, KL & Reio Jr, TG (2010) stated that Human Resource Information Systems **(HRIS)** solutions are more than just employee ***data houses***. They also allow employees and managers to take on some of the HR tasks that directly affect them. These solutions are ***helpful tools*** for performance and attendance tracking, and reporting systems that help with compliance and decision making.

Ferguson, KL & Reio Jr, TG (2010) also stated that the efficiency of Human Resource Information Systems (HRIS) solutions has enabled companies to produce more effective and efficient results than what can be accomplished on paper. By automating tasks and analyzing reports, companies use systems to their greatest potential in which Human Resource Information Systems (HRIS) features commonly include recruitment and selection, performance appraisals, training, compensation, etc.

To put it another way, Peretz H. (2013) stated that a human resource information system is basically an intersection of human resources and information technology through HR software in which a human resource information system (HRIS) will also lead to increases in efficiency when it comes to making decisions in HR that the decisions made should also increase in quality and as a result the productivity of both employees and managers should increase and become more effective.

According to Ferguson, KL & Reio Jr, TG (2010), An increased interest in human resource systems by the companies to achieve a desired competitive advantage in which these systems not only encompass single HR practice, but a set or sets of HR practices designed to increase individual and organizational performance for which key indicator of the effectiveness of the human resource system is the job performance that may also affect the overall organization performance.

According to Peretz H. (2013), many organizations have come under the services of an HRIS to support HR department in performing basic HR functions, enhancing administrative efficiency, and improving decision making, as well as speeding up information sharing. Usage of Information Technology has increased the effectiveness and efficiency of all tenets like planning, recruitment process, human resource accounting, training and development and other process of human resource management (HRM).

Ferguson, KL & Reio Jr, TG (2010) addressed that a human resource information system (HRIS) has been designed for human resource management to realize the strategic objectives of the organizations, to improve its productivity and proficiency by increasing the efficiency and effectiveness of daily functions. A human resource information system (HRIS) has given an opportunity to the organizations to solve and manage their different issues and processes related to the management of employees. Therefore, a human resource information system (HRIS) also used for different purposes, particularly in human resource processes as recruitment and selection, performance evaluation, compensation and benefits, training and development, etc.

6.3 Conclusion

In conclusion, I have followed the **object-oriented analysis and design approach** to design the **Human Resource Management system** by **defining** the ***functional requirements*** while making the project scope then **designing** the ***use case***, ***class***, ***sequence*** and ***activity*** diagrams in addition to the ***entity relationship diagram*** and the ***data flow diagram*** (traditional approach diagram to help determine the data stores), **making** the ***validation*** and ***verification*** (***V&V***) in order to test the logical design as early as possible and check that the logical design is moving in the right way, and finally **developing** the ***screens*** and making the project ***prototype*** to put the design into action. Therefore, I realized that there are many types of **human resources management** systems and these systems have contributed success to many organizations in the world. Information systems can help the organization to save time, cost, and energy in their operations and managements. These systems have provided a large database inventory for the organizations to store and maintain all the information.

I can conclude that human resources management system can help both **employer** and **employee** in order to do their job in which it can help organization going smoothly using technology for which a **Human Resource Management System** (HRMS) is a form of HR software that combines a number of systems and processes to ensure the easy management of a business's employees and data that the **objectives** of this project are improving and easing the process of human resource management by providing advanced features such as request features, payroll management features, organizational management features, workforce management features, time management features, personnel management features, benefit administration feature, and other features to reduce time & cost, and help to retain good employees.

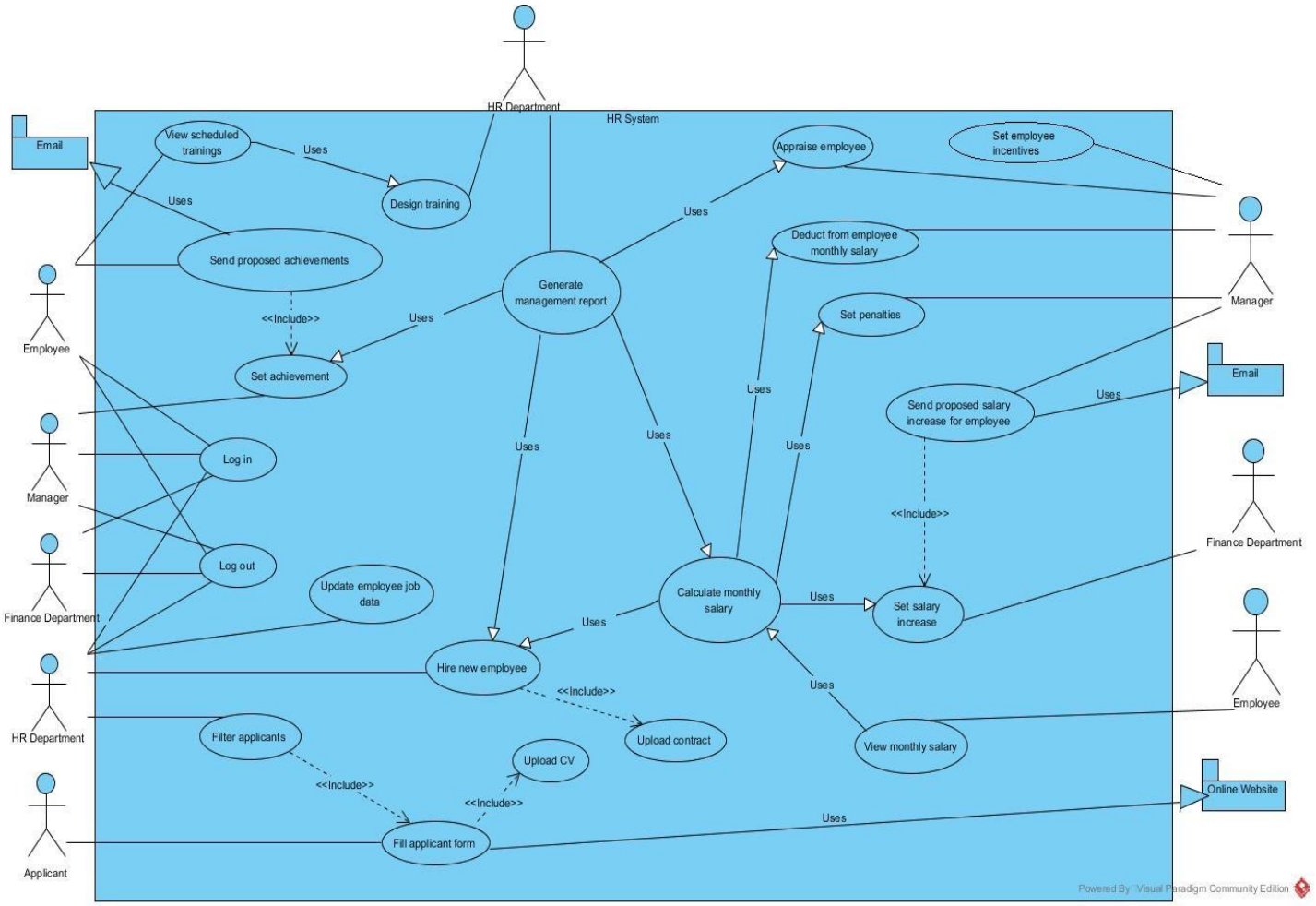
Project **deliverables** are attendance management, recruitment & hiring new employees, appraisal entry, salary deduction entry, salary increase entry, achievements entry, payroll management, generating management report, ability to view management report, design training for employee, and ability to view scheduled trainings in which the main **constraint** faces the project is ***access control*** that help us restrict whom and what accesses our information resources, and ***time of implementing*** is another constraint that prevented applying some **advanced features** which is *nice to have* such as tracking of employee performance, requesting vacations from the system & receive response, shift planning, travel

management, storing data about competitors' employees, requesting school coverage and making a daily online newspaper in the HR system to retain employees.

Finally, behind production of every product or service there is a human mind, effort and man hours (working hours) in which no product or service can be produced without help of human being for which human being is the fundamental resource for making or construction of anything. Today many experts claim that machines and technology are replacing human resource and minimizing their role or effort. However, indeed, machines and technology are built by the humans; they need to be operated or at least monitored by humans so that introduces the ***importance*** of **human resource management** and its **information system** in any company.

7.0 Use case model

7.1 Use Case diagram



7.2 Use Case Documentation:

Actors

Employee is a normal employee who is not managers, not in HR department or finance department. They have limited access to the system features in which they only permitted to **view** his/her **scheduled training** & **monthly salary**, and **send** his/her **proposed achievements**.

Manager is a person maybe marketing manager, operation manager, HR manager, finance manager, etc. who has the right to **appraise** his/her direct employees who report to him/her, **penalize** & **deduct** from salary his/her direct employees who report to him/her, **set achievements** for his/her employee, and **send salary increase request** for his/her employee.

HR Department is the department which has access to most of the system features such as **hiring new employee**, **setting new employee job data**, updating employee job data if change happened, **viewing monthly salary**, **scheduling trainings**, and **generating management report**.

Finance Department is the department which has access to approving finance features such as **setting salary increase**.

Applicant is a person who applies for role or a job in the company by **filling the application form** and **uploading the CV**.

Note

There are some actors and email package repeated or duplicated to ease the way management can view the use case diagram in which if the actors are not duplicated it will be very hard to view or understand the use case diagram for which it would be so complex to deal with the use case diagram if not duplicating the actors.

❖ **Use Case: Log in**

Actors: Employee, HR Department, Finance Department, Manager

Type: Primary

Description: All normal employees, managers, financial employees and HR employees must log in to calculate the employee attendance in which when employee logs in, it will be automatically calculated, and if he/she didn't log in this means he/she is absence & if he/she logged in late the system will calculate his/her lateness.

Pre-condition: The person who logs into the system must be authorized first before allowing him/her to use the system features.

Post-condition: The system should prepare when he/she attend every day, calculate absence and calculate lateness.

Cross reference: ----------

❖ **Use Case: Log out**

Actors: Employee, HR Department, Finance Department, Manager

Type: Primary

Description: All normal employees, managers, financial employees and HR employees must log out to calculate the employee attendance in which when employee log out, overtime or early leave will be automatically calculated if logged out before/after work time stated.

Pre-condition: The person who logs out of the system must log in first.

Post-condition: The system should prepare when he/she left every day, calculate overtime and calculate early leave.

Cross reference: ----------

❖ **Use Case: Fill applicant form**

Actor: Applicant

Type: Primary

Description: Applicant fill application form containing the applicant name, date of birth, address, phone number, objectives, skills and experiences.

Pre-condition: ---------

Post-condition: --------

Cross reference: **Filter applicants** in which the HR department must filter all applicants' experiences and skills in order to find the best applicant to hire for a specific job or role and **Upload CV** in which the applicant upload his/her contract after setting the his/her info.

❖ **Use Case: Upload contract**

Actor: Applicant

Type: Secondary

Description: Applicant uploads his/her CV in order if they meet the HR filtration criteria for the job the HR department views his/her CV.

Pre-condition: ---------

Post-condition: --------

Cross reference: **Fill applicant form** in which the applicant uploads his/her contract after setting the/her info.

❖ **Use Case: Filter applicants**

Actor: HR Department

Type: Primary

Description: The HR department must filter all applicants' experiences and skills in order to find the best applicant to hire for a specific job or role.

Pre-condition: The applicants' info must be set first

Post-condition: ----------

Cross reference: **Fill applicant form** in which the HR department must filter all applicants' experiences and skills in order to find the best applicant to hire for a specific job or role.

❖ **Use Case: Hire employee**

Actor: HR Department

Type: Primary

Description: After filtering the applicants to hire the best one, HR must set his/her data to hire the new employee such as employee ID, name, job tittle, job grade, salary, job description, department, bonus, housing allowance, gender, date of birth, address, phone number, marital status, courses needed & qualifications needed.

Pre-condition: HR should receive the applicants' data first and chose the best applicant to hire.

Post-condition: --------

Cross reference: **Upload contract** in which the contract must be uploaded after hiring the new employee.

❖ **Use Case: Upload contract**

Actor: HR Department

Type: Secondary

Description: After hiring the new employee, HR must upload the contract in order to have a soft copy form his/her contract on the system.

Pre-condition: A new employee must be hired first

Post-condition: ----------

Cross reference: **Hire new employee** in which a new employee must be hired first by the HR department before uploading his/her contract.

❖ **Use Case: Set penalties**

Actor: Manager

Type: Primary

Description: Managers can set penalty to be deducted from his/her monthly salary for example due to lack of complying with the manager or his/her team, and the deducted amount will be used when calculating monthly salary.

Pre-condition: ----------

Post-condition: ----------

Cross reference: **Calculate monthly salary** uses the penalty deducted amount to deduct it from his/her overall monthly salary.

❖ **Use Case: Deduct from employee salary**

Actor: Manager

Type: Primary

Description: Managers can deduct from employee salary for example due to medical insurance, social security insurance & other types of insurances payment and, and the deducted amount will be used when calculating monthly salary.

Pre-condition: ----------

Post-condition: ----------

Cross reference: **Calculate monthly salary** uses the salary deduction amount to deduct it from his/her overall monthly salary.

❖ **Use Case: Appraise employee**

Actor: Manager

Type: Primary

Description: Manager should appraise employee by evaluating and providing feedback on employee job performance that provides the basis for pay increase, bonus and promotions in which it is very important to help employees improve their performance and to help HR to tack employees' performance.

Pre-condition: ----------

Post-condition: ----------

Cross reference: **Generate management report** uses the employee appraisal in order to include the appraisal in the management report generated by the HR department to be used by

their direct manager, HR department & top management to track the employees' performance in the company.

❖ **Use Case: Send proposed achievements**

Actor: Employee

Type: Primary

Description: Employee should send proposed achievements made such as accomplishment made, courses taken, qualifications taken via **email** in which this use case (send proposed achievements) uses package called email to send the proposed achievement made to his/her direct manager in order to set & update his/her achievements made data.

Pre-condition: ----------

Post-condition: The manager must approve change made in achievements first before making any change in his/her achievements.

Cross reference: **Set achievements** in which the employee sends his/her proposed achievement made to his/her direct manager via email in order to set & update his/her achievements made data.

❖ **Use Case: Set achievements**

Actor: Manager

Type: Primary

Description: The manager is the one who set these achievements into the system to update his/her employee achievements after receiving the proposed achievements made via email from his/her employee first in order to set his/her achievements after approving and accepting them as an achievement made that will benefit the company.

Pre-condition: Employee should send first the proposed achievements made via email.

Post-condition: ----------

Cross reference: **Send proposed achievements** in which the employee sends his/her proposed achievement made to his/her direct manager via email in order to set his/her achievements made data by the manager, and then **Generate management report** uses the employee achievements (courses taken, qualifications taken, accomplishment made) in order to include them in the management report generated by the HR department to be used by their direct manager, HR department & top management to track the employees performance in the company.

❖ **Use Case: Send proposed salary increase for employee**

Actor: Manager

Type: Primary

Description: Manager request salary increase for his/her employee based on the company's policy regarding salary increase in which the manager send proposed salary increase for employee (amount) via **email** in which this use case (send proposed salary increase for employee) uses package called email to send the proposed salary increase for employee to finance department in order to set the salary increase after approving & accepting the salary increase due to having sufficient budget for salary increases.

Pre-condition: Employee must make achievements in the company such as making accomplishment, taking a course or qualification that will benefit the company in order for manager to request salary increase for him/her.

Post-condition: The finance department must approve the salary increase first before making any change in system.

Cross reference: **Set salary increase** in which the manager sends his/her proposed salary increase for employee to the finance department via email in order to set his/her salary increase.

❖ **Use Case: Set salary increase**

Actors: Finance department

Type: Primary

Description: The finance department set the salary increase into the system to update his/her employee basic salary after receiving the proposed salary increase for employee via email from his/her manager first in order to set his/her salary increase after approving and accepting the increase.

Pre-condition: Manager should send first the proposed salary increase for employee via email.

Post-condition: ----------

Cross reference: **Send proposed salary increase for employee** in which the manager sends his/her proposed salary increase for employee to the finance department via email in order to set his/her salary increase by the finance department, and then **Calculate monthly salary** uses the salary increase amount to add it to his/her overall monthly salary.

❖ **Use Case: Generate monthly salary calculation**

Actor: HR department

Type: Primary

Description: The HR department generate monthly salary calculation then the system calculates monthly salary by using the **employee attendance data** (absence, lateness, overtime, early leave), **employee salary data** (salary, bonus & housing insurance)**, employee salary increase** (salary increase amount if there is salary increase that have been set for this employee), **employee penalties** (penalty amount if the employee is penalized this month), **employee salary deduction** (salary deduction amount if the employee deducted from his salary this month).

Pre-condition: ----------

Post-condition: ----------

Cross reference: **Set employee job data** by using his/her salary data (salary, bonus, housing allowance) in order to calculate monthly salary by the system every month, **set penalties** by using the penalty deducted amount to deduct it from his/her overall monthly salary, **deduct from employee salary** by using the salary deduction amount to deduct it from his/her overall monthly salary, and **Set salary increase** by using the salary increase amount to add it to his/her overall monthly salary.

❖ **Use Case: View monthly salary**

Actors: Employee

Type: Primary

Description: Employee can view the employee monthly salary.

Pre-condition: Monthly salary must be prepared by the system first to be able to view it.

Post-condition: ----------

Cross reference: **Calculate monthly salary** in which monthly salary must be prepared by the system first to be able to view it.

❖ **Use Case: Generate management report**

Actor: HR Department

Type: Primary

Description: HR department generates management report containing the **employee attendance data** (absence, lateness, overtime, early leave), **employee achievements needed data** (courses needed & qualifications needed to be taken to benefit the company**), employee**

achievements made data (courses taken, qualifications taken, accomplishments made), **employee appraisal** (appraisal document), **employee monthly salary** (monthly salary amount paid to him/her) in order to be used by their direct manager, HR department & top management to track the employees performance in the company.

Pre-condition: ----------

Post-condition: ----------

Cross reference: **Set employee job data** by using his/her achievement needed data (courses needed, qualifications needed) in order to include them in the management report**, Appraise employee** by using the employee appraisal in order to include the appraisal in the management report **, Set achievements** by using the employee achievements made (courses taken, qualifications taken, accomplishment made) in order to include them in the management report, and **Calculate monthly salary** by using the employee monthly salary in order to include the monthly salary in the management report.

- **Use Case: Design training**

Actor: HR Department

Type: Primary

Description: HR department can design training such as courses, qualifications, seminars and business events to benefit his/her company in his/her field & to improve the business effectiveness and efficiency, and then set the training type, date and time.

Pre-condition: ---------

Post-condition: --------

Cross reference: **View scheduled trainings** in which employee can view the scheduled training for them after the HR department has design & set the training type, date and time for them.

- **Use Case: View scheduled trainings**

Actors: Employee

Type: Primary

Description: Employee can view the employee training (type, date, time) assigned to him/her in order to benefit the company.

Pre-condition: Training must be designed by the HR department first to be able to view it.

Post-condition: ----------

Cross reference: **Design training** in which training must be designed by the HR system first to be able to view it.

- **Use Case:** **Update employee job data**

Actor: HR department

Type: Primary

Description: HR department can update the employee personal information for example if he/she changed his home address or mobile number.

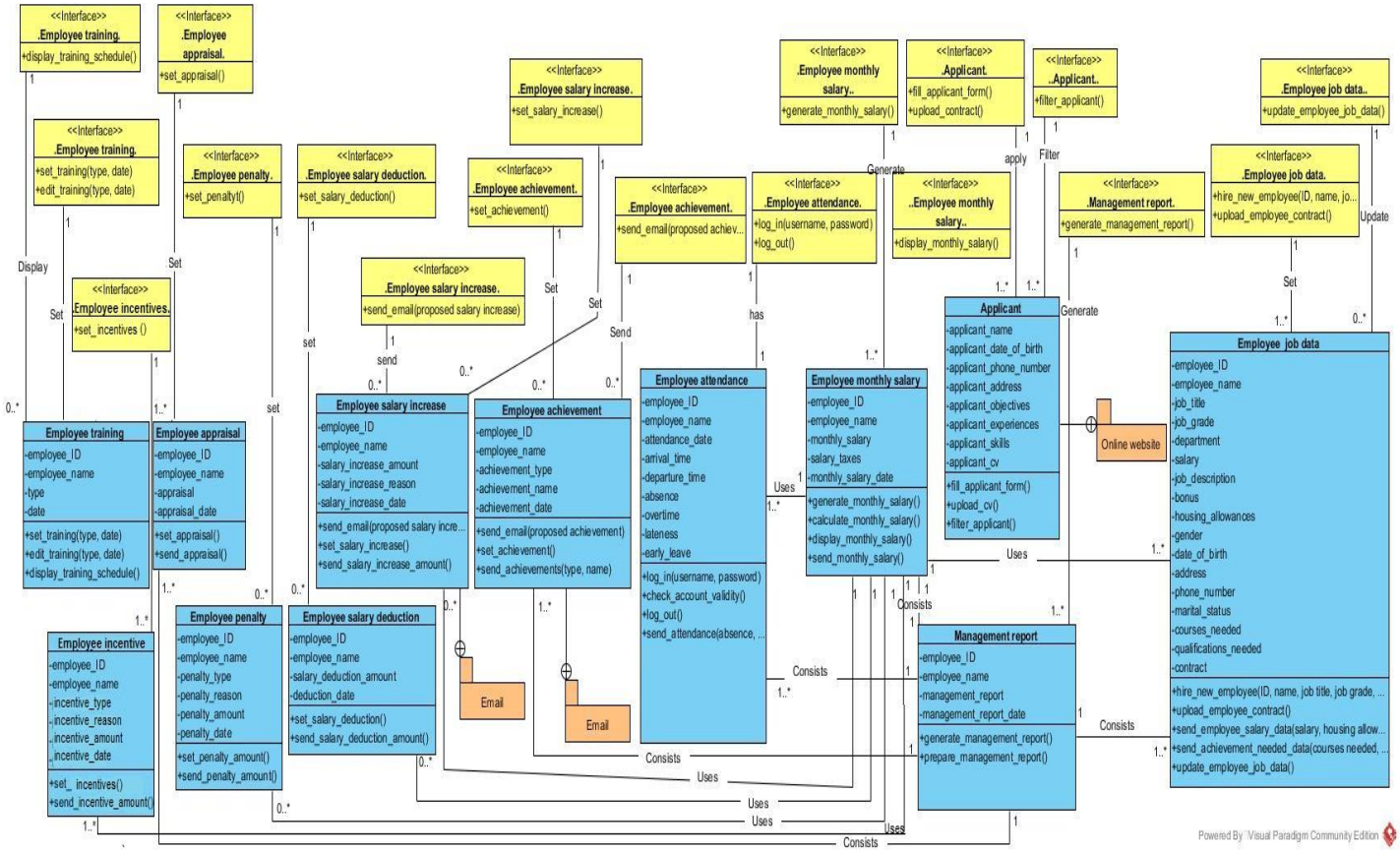
Pre-condition: ----------

Post-condition: ----------

Cross reference: ----------

8.0 Class model

8.1 Class diagram



8.2 Class diagram Documentation

❖ **Class: Interface "Employee training"**

Relations: HR department can set zero or more trainings.

Attributes: - - - - - - - - - -

Operations: Set training (type, date, time) in which the HR department set the training type (course, qualification or seminar), date & time to store it in the system for further use.

❖ **Class: Interface "Employee training"**

Relations: Every employee can display zero or more trainings.

Attributes: - - - - - - - - - -

Operations: Display training schedule in which the employee can view the training type (course, qualification or seminar), date & time from the system.

❖ **Class: Employee training**

Relations: HR department can design zero or more trainings, and every employee can display zero or more trainings.

Attributes: Employee ID & name, and training type, date &time.

Operations: Set training (type, date, time) in which the HR department set the training type (course, qualification or seminar), date & time to store it in the system, and Display training schedule in which the employee can view the training type (course, qualification or seminar), date & time from the system.

❖ **Class: Interface "Employee appraisal"**

Relations: Every manager can set one or more appraisal.

Attributes: - - - - - - - - - -

Operations: Set appraisal in which Manager set appraisal by setting the employee name & ID, appraisal & date in the system for further use.

❖ **Class: Employee appraisal**

Relations: Every manager can set one or more appraisal.

Attributes: Employee ID & name, appraisal and appraisal date.

Operations: Set appraisal in which Manager set appraisal by setting the employee name & ID, appraisal & date in the system for further use and send appraisal in order in include it in the management report.

❖ **Class: Interface "Employee penalty"**

Relations: Every manager can set zero or more penalty.

Attributes: - - - - - - - - - -

Operations: Set penalty in which Manager set penalty by setting the employee name & ID, type, reason, amount & date in the system for further use.

❖ **Class: Employee penalty**

Relations: Every manager can set zero or more penalty.

Attributes: Employee ID & name, penalty amount, penalty type, penalty reason and penalty date.

Operations: Set penalty in which Manager set penalty by setting the employee name & ID, type, reason, amount & date in the system for further use, and send penalty amount in order in include it in the monthly salary calculation.

❖ **Class: Interface "Employee salary deduction"**

Relations: Every manager can set zero or more salary deduction.

Attributes: - - - - - - - - - -

Operations: Set salary deduction in which Manager set salary deduction by setting the employee name & ID, reason, amount & date in the system for further use.

❖ **Class: Employee salary deduction**

Relations: Every manager can set zero or more salary deduction.

Attributes: Employee ID & name, salary deduction amount, salary deduction reason and salary deduction date.

Operations: Set salary deduction in which Manager set salary deduction by setting the employee name & ID, reason, amount & date in the system for further use, and send salary deduction amount in order in include it in the monthly salary calculation.

❖ **Class: Interface "Employee salary increase"**

Relations: Every manager can send zero or more email regarding salary increase request.

Attributes: - - - - - - - - - -

Operations: Send email (proposed salary increase) in which Manager send proposed salary increase to send it to finance department via email for setting the salary increase.

❖ **Class: Interface "Employee salary increase"**

Relations: Finance department set zero or more salary increase.

Attributes: - - - - - - - - - -

Operations: Set salary increase by setting employee name& ID, amount, reason & date in which finance department set salary increase to store it in the system for further use after receiving the email from manager regarding the proposed salary increase.

❖ **Class: Employee salary increase**

Relations: Every manager can send zero or more email regarding salary increase request and Finance department set zero or more salary increase.

Attributes: Employee ID, employee name, salary increase amount, salary increase reason and salary increase date.

Operations: Send email (proposed salary increase) in which Manager send proposed salary increase to send it to finance department via email for setting the salary increase, Set salary increase by setting employee name& ID, amount, reason & date in which finance department set salary increase to store it in the system for further use after receiving the email from manager regarding the proposed salary increase, and send salary increase amount in order in include it in the monthly salary calculation.

❖ **Class: Interface "Employee achievements"**

Relations: Every employee can send zero or more email regarding proposed achievements made.

Attributes: - - - - - - - - - -

Operations: Send email (proposed achievement) in which Employee sends proposed achievements made to send it to manager via email for setting the achievements.

❖ **Class: Interface "Employee achievements "**

Relations: Manager set zero or more achievement.

Attributes: - - - - - - - - - -

Operations: Set achievement by setting the employee name & ID, achievement type & date in which Manager set achievement & its date to store it in the system for further use after receiving the email from employee regarding the proposed achievements.

❖ **Class: Employee achievement**

Relations: Every employee can send zero or more email regarding proposed achievements made and Manager set zero or more achievement.

Attributes: Employee ID, employee name, achievement type, achievement name, and achievement date.

Operations: Send email (proposed achievement) in which Employee send proposed achievements made to send it to manager via email for setting the achievements, set achievement by setting the employee name & ID, achievement type, name & date in which

Manager set achievement & its date to store it in the system for further use after receiving the email from employee regarding the proposed achievements, and send achievement in order in include it in the management report.

❖ **Class: Interface "Employee attendance"**

Relations: Every employee, manager, HR employee and finance employee has one or more attendance in which for example every employee can have one or many arrival time in different days.

Attributes: - - - - - - - - - -

Operations: Log in (username, password) & Log out in which every employee, manager, HR employee and finance employee has only one attendance to store it in the system for further use.

❖ **Class: Employee attendance**

Relations: Every employee, manager, HR employee and finance employee has one or more attendance in which for example every employee can have one or many arrival time in different days.

Attributes: Employee ID & name, attendance date, arrival time, departure time, absence, lateness, overtime & early leave.

Operations: Log in (username, password) & Log out in which every employee, manager, HR employee and finance employee has only one attendance to store it in the system for further use, check account validity in which the system checks the account validity when he/she log in to the system in which the system check if the username and password are valid or not, and send attendance (absence, lateness, overtime, early leave) in order to include it in the monthly salary calculation and generating management report.

❖ **Class: Interface "Monthly salary"**

Relations: HR department can generate one monthly salary or more for one employee or many employees.

Attributes: - - - - - - - - - -

Operations: Generate monthly salary in which HR generate monthly salary in order to calculate monthly salary for the company's employees.

❖ **Class: Interface "Monthly salary"**

Relations: Every employee can display his/her monthly salary

Attributes: - - - - - - - - - -

Operations: Display monthly salary in which every employee can view his/her monthly salary.

❖ **Class: Monthly salary**

Relations: Monthly salary generated by the HR department for which every employee can display his/her monthly salary.

Attributes: Employee ID & name, monthly salary, salary taxes and monthly salary date.

Operations: Generate monthly salary in which HR generate monthly salary in order to calculate monthly salary for the company's employees, calculate monthly salary in which the system after receiving all the required data calculate automatically the monthly salary for each employee, display monthly salary in which every employee can view his/her monthly salary, and send monthly salary in order to include it in the management report.

❖ **Class: Interface "Management report"**

Relations: HR department can generate one or more management report.

Attributes: - - - - - - - - - -

Operations: Generate management report in which HR department can generate management report in order to track employee performance.

❖ **Class: Management report**

Relations: HR department can generate one or more management report.

Attributes: Employee ID & name, management report, and management report date.

Operations: Generate management report in which HR department can generate management report in order to track employee performance, prepare management report in which the system after receiving all the required data start preparing the management report automatically.

❖ **Class: Interface "Applicant"**

Relations: HR department can fill one or more applicant form for different jobs or roles.

Attributes: - - - - - - - - - -

Operations: fill applicants form containing applicant name, date of birth, address, phone number, objectives, skills and experiences & upload CV.

❖ **Class: Interface "Applicant"**

Relations: HR department can filter one or more applicant info to hire the best applicant.

Attributes: - - - - - - - - - -

Operations: Filter applicants based on applicants' skills & experiences to choose the best applicant to hire.

❖ **Class: Applicant**

Relations: HR department can set one or more applicant info for different new applicants.

Attributes: Applicant name, date of birth, address, phone number, objectives, skills and experiences

Operations: fill applicants form containing applicant name, date of birth, address, phone number, objectives, skills and experiences upload CV, and Filter applicants based on applicants' skills & experiences to choose the best applicant to hire.

❖ **Class: Interface "Employee job data"**

Relations: HR department can hire one or more employee

Attributes: - - - - - - - - - -

Operations: Hire new employee (employee ID, name, job tittle, job grade, salary, job description, department, address, gender, date of birth, phone number, marital status, courses needed, qualifications needed), and upload employee contract in which HR department those employee job data after hiring a new employee.

❖ **Class: Interface "Employee job data"**

Relations: HR department can update one or more employee job data for different current employees.

Attributes: - - - - - - - - - -

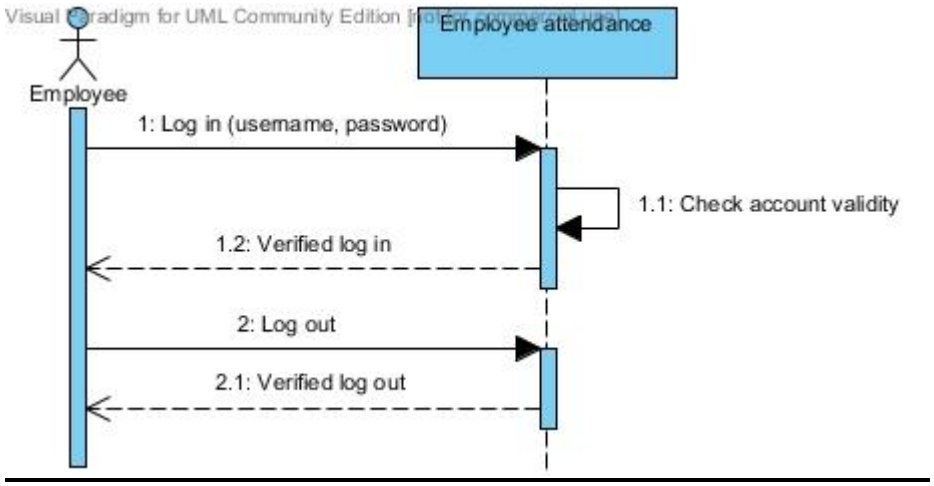
Operations: Update employee job data in which HR department can update employee job data in order to make sure the data is always correct about their employees.

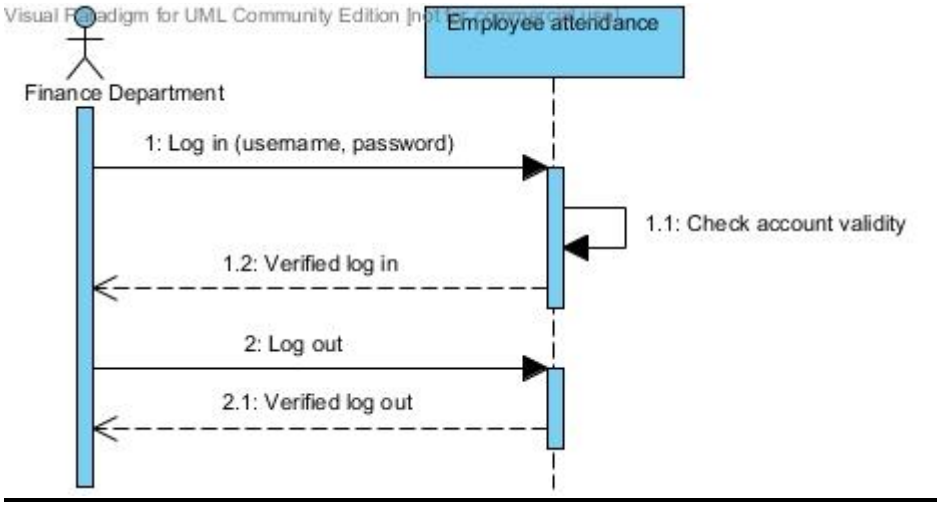
❖ **Class: Employee job data**

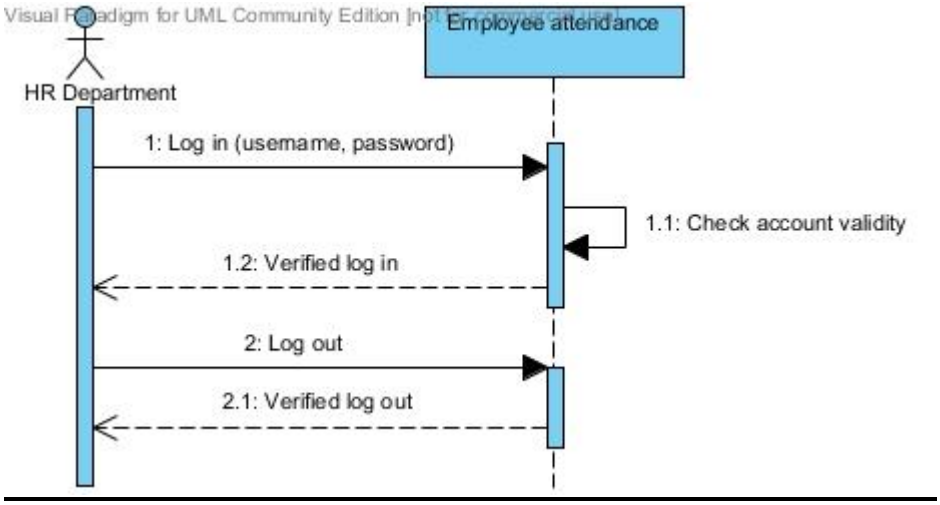
Relations: HR department can hire one or more employee and set one or more employee job data for different new employees and HR department can update one or more employee job data for different current employees.

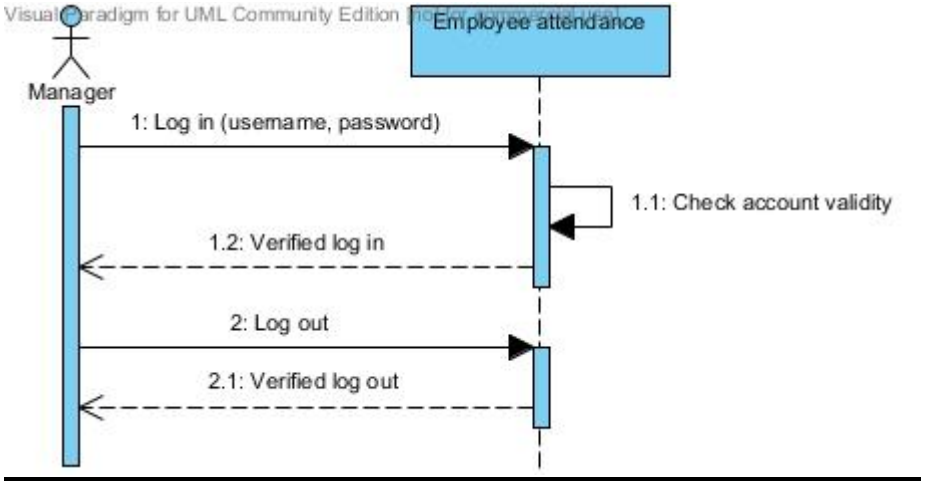
Attributes: Hire new employee (employee ID, name, job tittle, job grade, salary, job description, department, address, gender, date of birth, phone number, marital status, courses needed, qualifications needed), upload employee contract in which HR department those employee job data after hiring a new employee, send employee salary data (salary, bonus, housing allowance) in order to calculate monthly salary, send achievement needed data (courses needed, qualifications needed) in order to include them in the management report, and update employee job data in which HR department can update employee job data in order to make sure the data is always correct about their employees.

9.0 Sequence model - documented





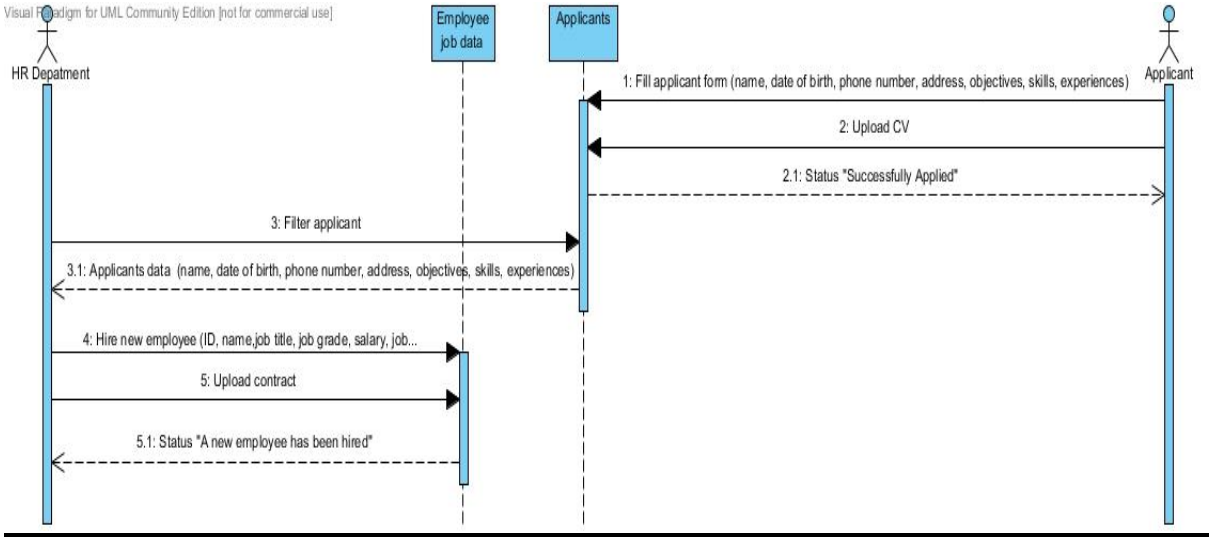




Log in & log out:

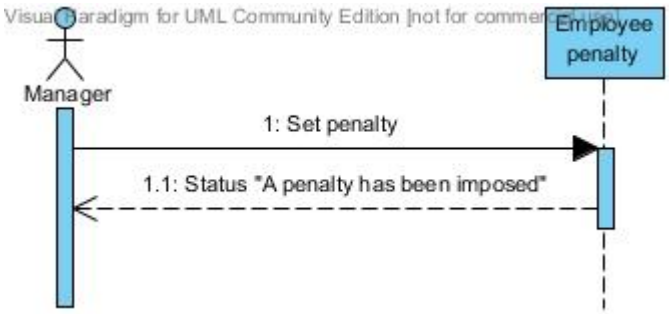
Employee, **manager**, **HR** or **finance** **log in** then the system **checks the account validity** in which it checks if username& password are valid & finally receiving response (**verified log in**). After sending the verifying log in, the system set Arrival time will be set automatically in the employee attendance class then after setting the arrival time the employee attendance class begins to calculate absence if he/she didn't log in and calculate lateness if he/she logged in late which is explained in the activity diagram.

Employee, **manager**, **HR** or **finance** **log out** & then receive response (**verified log out**). After sending the verifying log out, the system set Departure time will be set automatically in the employee attendance class then after setting the departure time the employee attendance class begins to calculate overtime if he/she log out late and calculate early leave if he/she logged out early which is explained in the activity diagram.



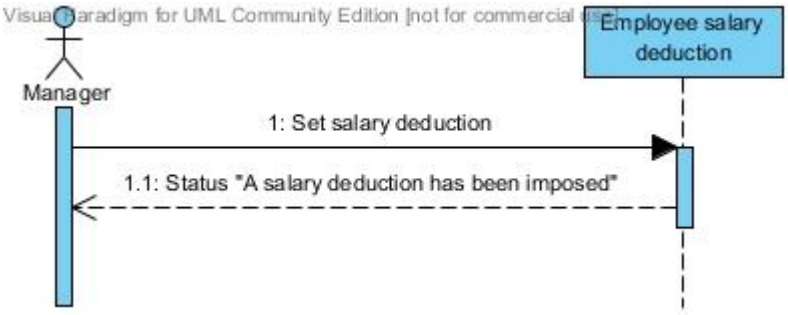
Filling applicant form, uploading cv, filter applicants, hiring new employee & uploading contract:

Applicant uses an interface in order to **fill applicant form** (name, date of birth, phone number, address, objectives, skills, experiences) then **uploading CV** and finally the system sends Status "Successfully applied" after finishing the applicants form & uploading the CV. Therefore, the **HR department filter applicants** based on skills & experiences needed for a specific role or job then receive from the system the applicants who have the needed skills & experiences for the specific role or job. Finally, **HR department** uses an interface in order to **hire the new employee** chosen after filtering applicants (New employee ID, name, job tittle, job grade, salary, job description, department, bonus, housing allowance, gender, address, date of birth, phone number, marital status, courses needed, qualifications needed), and **upload contract** & finally receive response (**status "A new employee has been hired"**) all of those operations are done by either applicant class or employee job data class.



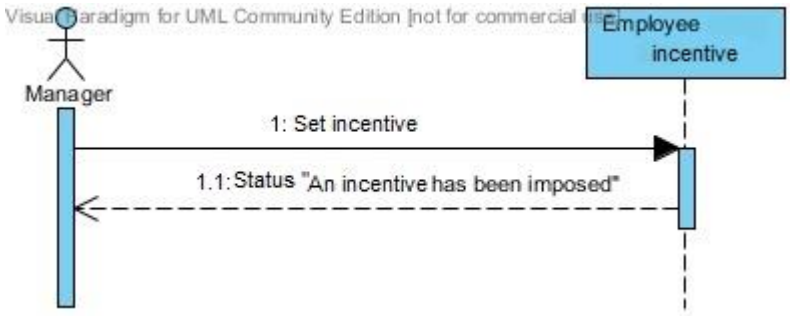
Setting penalties:

Manager uses an interface in order to **set penalty** (employee ID & name, type, reason, amount, date), and finally receive response (**status "A penalty has been imposed"**) all of those operations are done by employee penalty class.



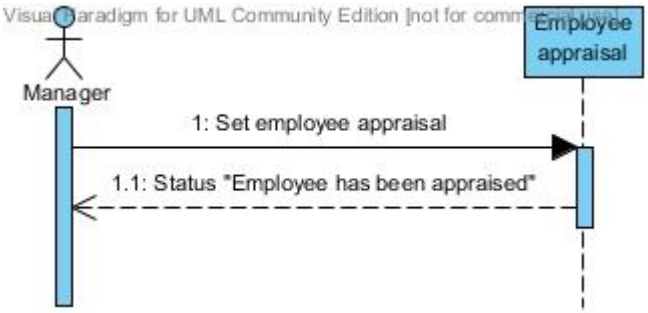
Deducting from employee salary:

Manager uses an interface in order to **set salary deduction** (Employee name & ID, type, amount & date) and finally receive response (**status "A salary deduction has been imposed"**) all of those operations are done by employee salary deduction class.



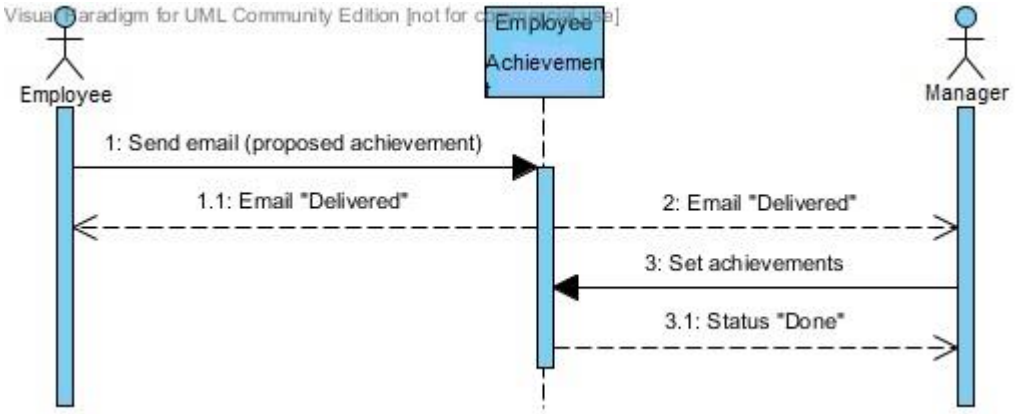
Set incentives for employees:

Manager uses an interface in order to **set incentives** (Employee name & ID, type, reason, amount & date) and finally receive response (**status "An incentive has been imposed"**) all of those operations are done by employee incentive class.



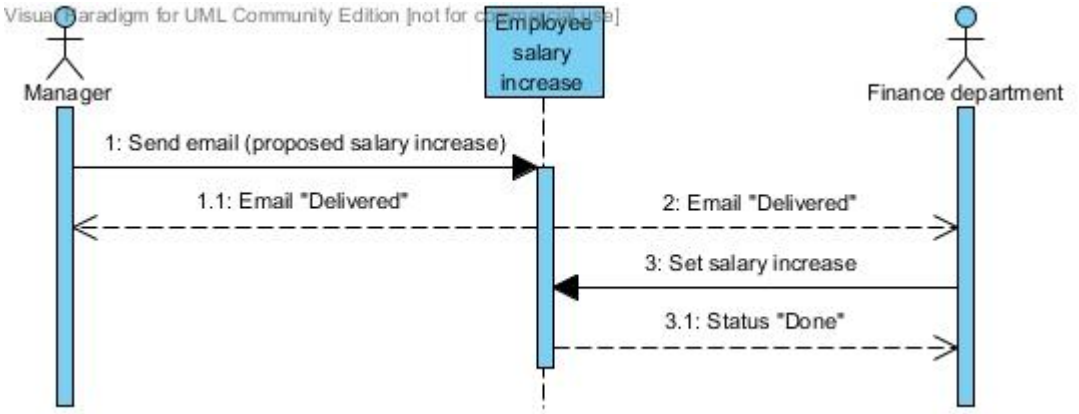
Appraising the employee:

Manager uses an interface in order to **set salary deduction** (Employee name & ID, appraisal & date) and finally receive response (**status "Employee has been appraised"**) all of those operations are done by employee appraisal class.



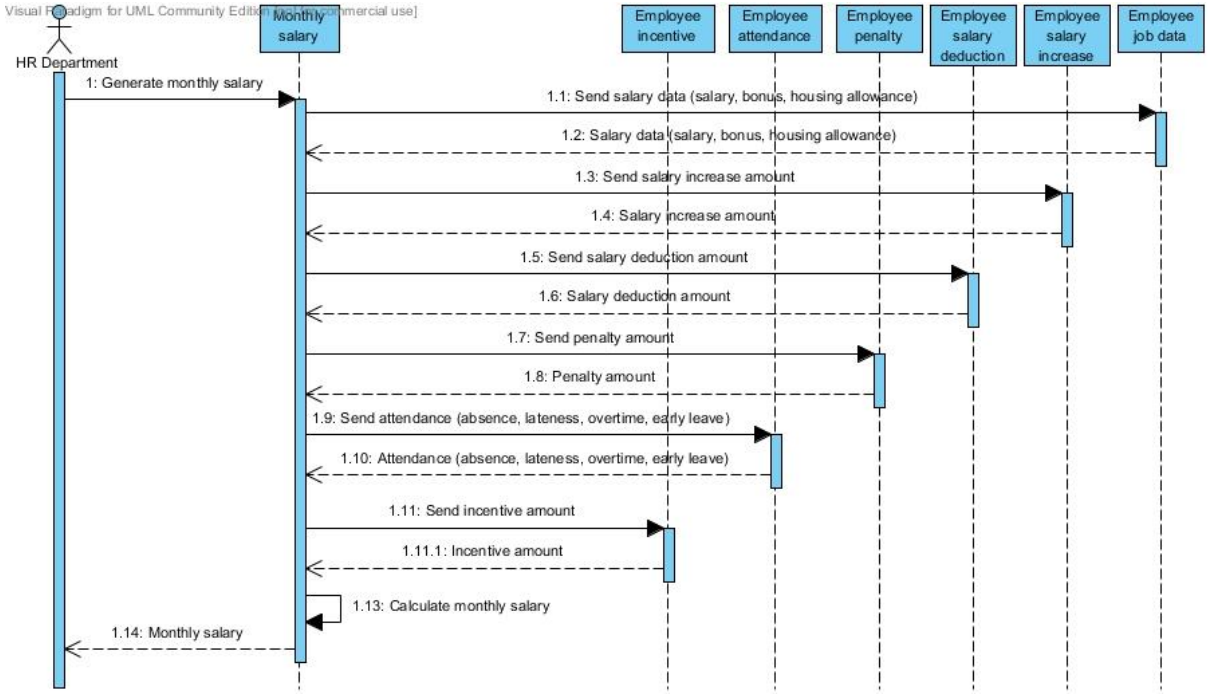
Sending proposed achievements & setting the achievements:

Employee uses an interface in order to **send email (proposed achievement)** & receive response (**Email "Delivered"**) and after the **manager** receives the email uses an interface in order to **set achievement** (employee name & ID, date, achievement type such as courses taken, qualifications taken and/or achievement made, and achievement name such as java, CMA, etc.) & finally receive response (**status "Done"**) all of those operations are done by employee achievement class.



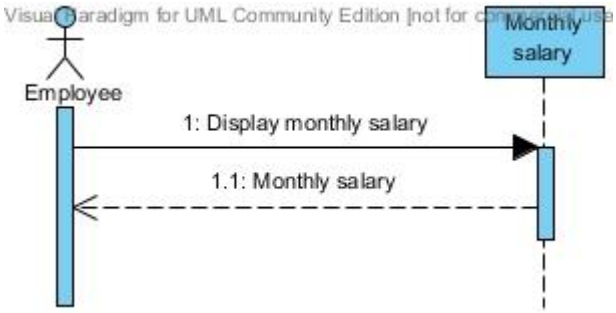
Sending proposed salary increase for employee & setting the salary increase:

Manager uses an interface in order to **send email (proposed salary increase)** & receive response (**Email "Delivered"**) and after the **manager** receives the email uses an interface in order to **set salary (**employee ID & name, amount, reason & date) and finally receive response (**status "Done"**) all of those operations are done by employee salary increase class.



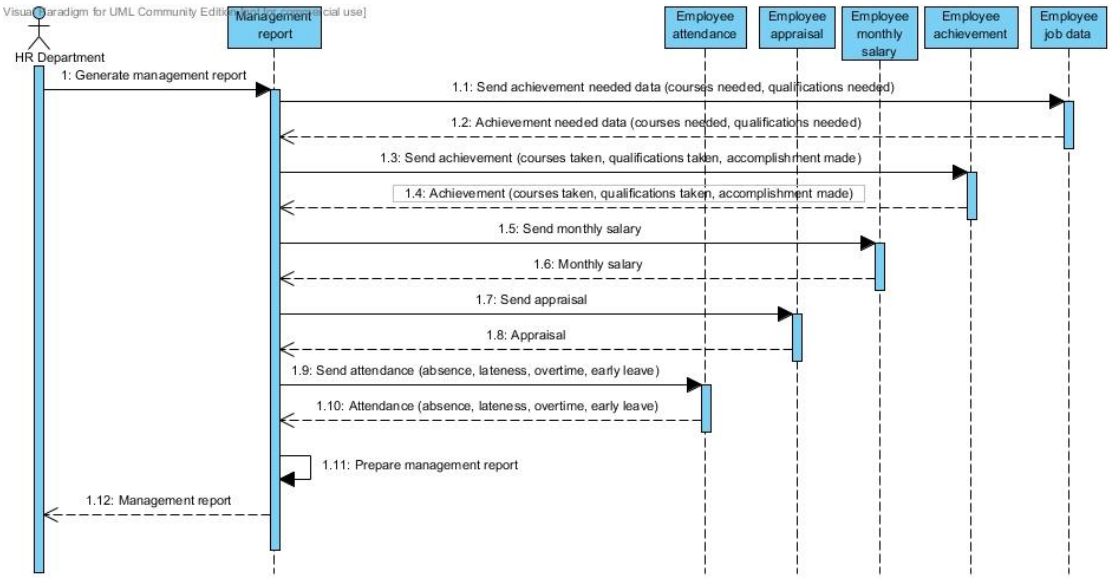
Calculating monthly salary:

HR department generate monthly salary then **employee monthly salary class** first gives order to employee job data class in order to send the salary data and as an response employee job data class sends the salary, bonus & housing allowance, second gives order to employee salary increase class in order to send the salary increase amount and as an response employee salary increase class sends the salary increase amount, third gives order to employee salary deduction class in order to send the salary deduction amount and as an response employee salary deduction class sends the salary deduction amount, forth gives order to employee penalty class in order to send the penalty amount and as an response employee penalty class sends the penalty amount, fifth gives order to employee attendance class in order to send the attendance and as an response employee attendance class sends the employee absence, lateness, overtime & early leave, seventh gives order to the employee incentive class in order to send the incentive amount and as an response employee incentive class sends the incentive amount, and finally eighth the employee monthly salary class after receiving all the required data **calculate monthly salary** then **set monthly salary date**, and finally display the monthly salary to HR department as an response of generating monthly salary.



Viewing monthly salary:

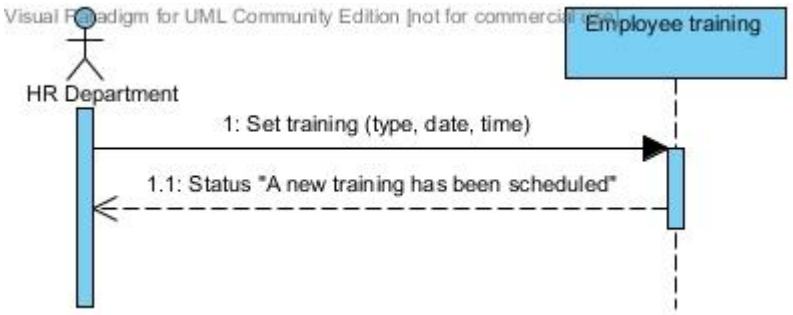
After preparing the monthly salary, **Employee** can give order to monthly salary class to **display monthly salary** and as a response HR department or Employee receive the monthly salary and its details.



Generating management report:

HR department give order to **Management report class** to **generate management report** in which management report class first gives order to employee job data class in order to send the employee job data and as an response employee job data class sends the courses needed & qualifications needed, second gives order to employee achievement class in order to send the achievements and as an response employee achievement class sends the achievements, third gives order to employee monthly salary class in order to send the monthly salary and as an response employee monthly salary class sends the monthly salary, forth gives order to employee appraisal class in order to send the appraisal and as an response employee appraisal class sends the appraisal, fifth gives order to employee attendance class

in order to send the attendance and as an response employee attendance class sends the employee absence, lateness, overtime & early leave, sixth the management report class after receiving all the required data **prepare management report** and then **set management report date**. After preparing the management report and setting the date, display management report to the HR department as response of generating management report.

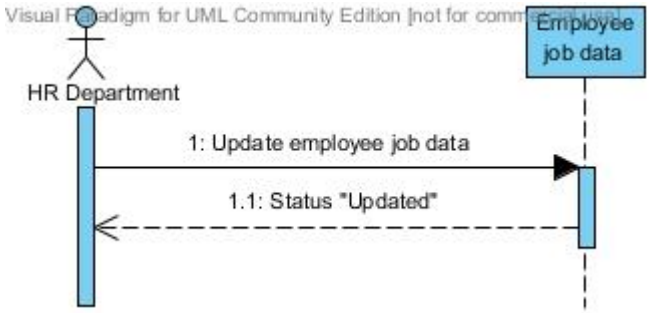


Designing trainings:

HR department uses an interface in order to **set training** (type, date, time) & receive response (**status "A new training has been scheduled"**) all of those operations are done by employee training class.

Viewing employee training:

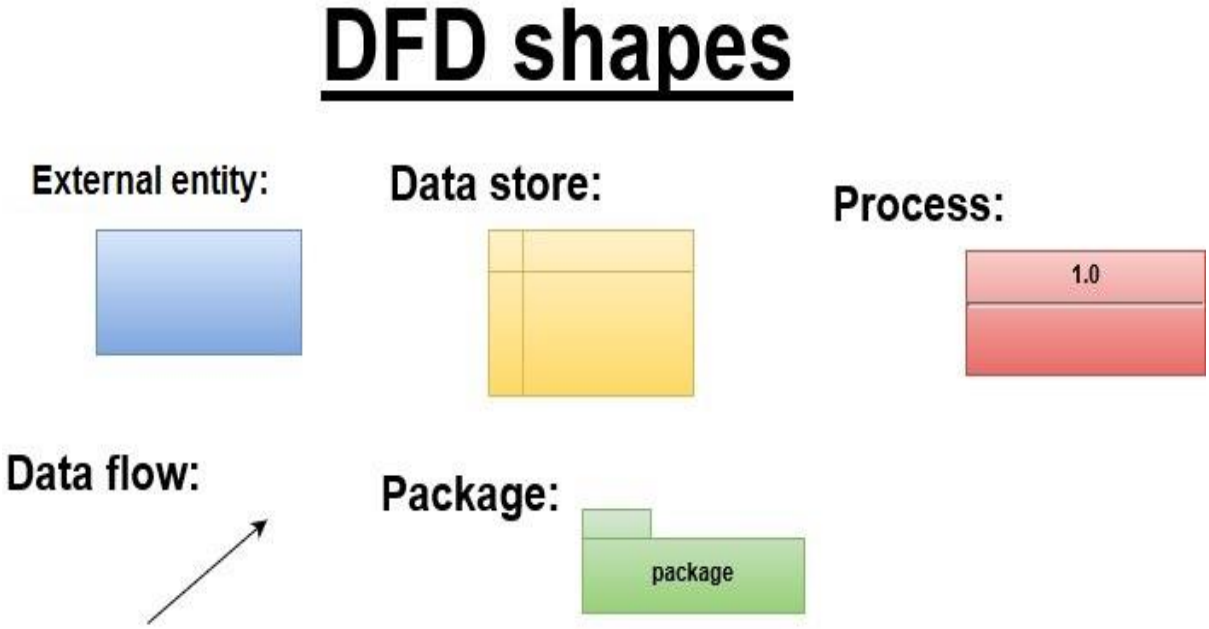
After designing the trainings, the **Employee** can give order to training class to **display training** and as a response Employee receive the employee training type, date & time.

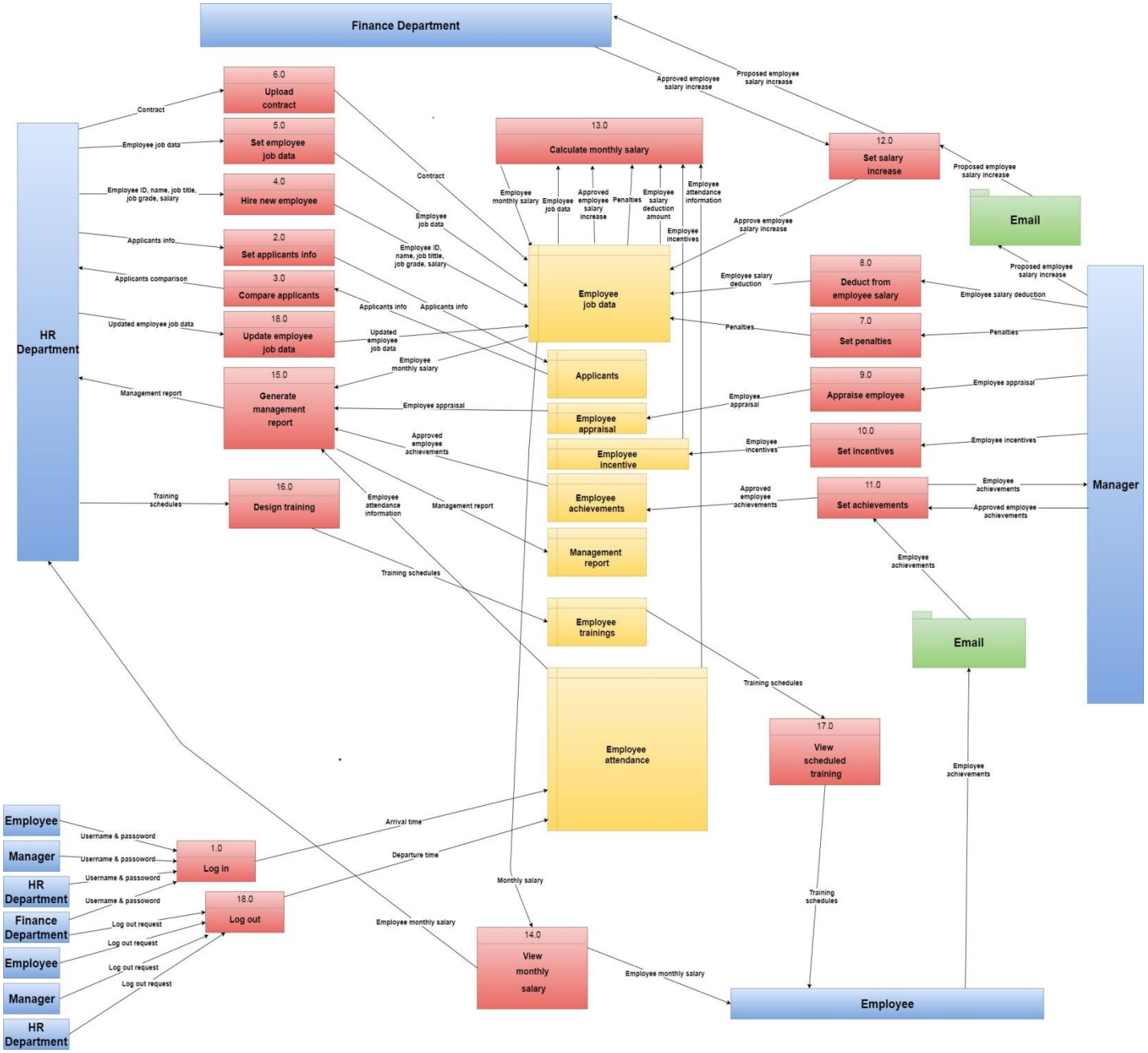


Updating employee job data:

HR department uses an interface in order to **update employee job data** & receive response (**status "Updated"**) all of those operations are done by employee job data class.

10.0 DFD Level-0





10.1 DFD level-0 Documentation

Employee is a normal employee who is not managers, not in HR department or finance department. They have limited access to the system features in which they only permitted to **view** his/her **scheduled training** & **monthly salary**, and **send** his/her **achievements**.

Manager is a person maybe marketing manager, operation manager, HR manager, finance manager, etc. who has the right to **appraise** his/her direct employees who report to him/her, **penalize** & **deduct** from salary his/her direct employees who report to him/her, **set achievements** for his/her employee, and **send salary increase request** for his/her employee.

HR Department is the department which has access to most of the system features such as **hiring new employee**, **upload contract**, **filtering applicants, updating employee job data** if change happened, **viewing monthly salary**, **scheduling trainings**, and **generating management report**.

Applicant is the person who has access to applying for a job or role features such as **filling applicant form**, and **uploading CV**.

Finance Department is the department which has access to approving finance features such as **setting salary increase**.

Note

There are some external entities repeated or duplicated to ease the way DFD level-0 can be viewed in which if the external entities are not duplicated it will be very hard to view or understand the DFD level-0 for which it would be so complex to deal with the DFD level-0 if not duplicating the actors.

❖ **Process (1.0): Log in**

Description: All normal **employees**, **managers**, **financial employees** and **HR employees** must log in by **entering** their **username & password** in order to identify if he is allowed to enter the system and his/her access. After log in, his/her **arrival time** is **sent** to the **employee attendance data store** in order to automatically calculate lateness or if he/she didn't log in this means he/she is absence.

❖ **Process (2.0): Fill applicants form**

Description: The **applicant sends** the **applicants info** (such as name, phone number, address, objectives, experiences, skills) to store the applicants' info in the **applicants' data store**.

❖ **Process (3.0): Upload CV**

Description: The **applicant sends** the **CV** to store it in the **applicants' data store**.

❖ **Process (4.0): Filter applicants**

Description: The **HR department** after setting the applicants info then the HR can filter all applicants **receives** the **filtered applicants' info** (recommending the best candidates that the company should hire based on their skills, experience) form the **applicants' data store**.

❖ **Process (5.0): Hire new employee**

Description: The **HR department** when hiring a new employee **sends** the **employee job data (**ID, name, job title, job grade, salary, as job description, department, gender, date of birth, address, phone number, marital status, courses needed, qualifications needed**)** to store the employee basic data in the **employee job data store**.

❖ **Process (6.0): Upload contract**

Description: The **HR department** when hiring a new employee & setting his/her job data **sends** the **employee contract** to store it in the **employee job data class**.

❖ **Process (7.0): Set penalties**

Description: The **Manager sends** the **penalties** (containing the penalty amount from employee due to lack of complying with the manager or his/her team, and the deducted amount will be used when making salary calculation) to store in the **employee job data store**.

❖ **Process (8.0): Deduct from employee salary**

Description: The **Manager sends** the **employee salary deduction** (containing the salary deduction amount from employee due to payment of different types of insurances, and the deducted amount will be used when making salary calculation) to store in the **employee job data store**.

❖ **Process (9.0): Appraise employee**

Description: The **Manager sends** the **employee appraisal** (containing the manager evaluating and providing feedback on the employee job performance that provides a basis for pay increase, bonus and promotions in which it is very important to help employees improve their performance) to store in the **employee appraisal data store**.

❖ **Process (10.0): Set achievements**

Description: The **Employee** when making a new accomplishment, new course taken or qualification taken **sends** the **proposed employee achievements** to the **set achievements process** via ***email*** to approve & set the employee new achievements made **(**to make sure the data entered is correct and already happened) by the **Manager** in order to store the **approved employee achievements** in the **employee achievements data store**.

❖ **Process (11.0): Set salary increase**

Description: The **Manager** when he/she sees that his/her employee deserve a salary increase **sends** the **proposed employee salary increase** to the **set salary increase process** via ***email*** to approve & set the employee new salary increase **(**to make sure the salary increase the company can handle and the employee really deserve) by the **Finance department** in order to store the **approved employee salary increase** in the **employee job data store**.

❖ **Process (12.0): Calculate monthly salary**

Description: Calculating monthly salary process **receives** the **approved employee job data** (containing employee salary, bonus & housing allowance) from the **employee job data store**, **approved employee salary increase** if there is salary increase from the **employee job data store**, **employee salary deduction amount** if there is salary deduction from **employee job data store**, **employee penalties amount** if there is penalty from **employee job data store** and **employee attendance information** (containing the overall employee attendance, no. of leaving early, no. of overtime, no. of absence, etc.) from the **employee attendance data store** in order to calculate the employee monthly salary by the system , and then **sends** the **employee monthly salary** to the **employee job data store** to store it.

❖ **Process (13.0): View monthly salary**

Description: **HR department** and **Employee** can view the monthly salary by **receiving** monthly salary from **employee job data store**.

❖ **Process (14.0): Generate management report**

Description: Generating management report process **receives** the **employee monthly salary** (containing employee monthly salary) from the **employee job data store**, **employee job data** (containing the courses & qualification needed to be taken) from employee job data store, **approved employee achievements** (containing the approved achievements such as

courses taken, qualifications taken and/or accomplishment made) from **employee achievements data store**, **employee attendance information** (containing the overall employee attendance, no. of leaving early, no. of overtime, no. of absence, no, of annual leave valid, no. of absence valid, no. of sick leave valid, etc.) from **employee attendance data store** and **employee appraisal** from **employee appraisal data store**, and **sends** the **management report** (employee attendance, employee job data, approved achievements made, employee appraisal and monthly salary) to the **HR department** to view the report.

❖ **Process (15.0): Design training**

Description: The **HR department sends trainings schedules** (containing training type, date & time) in order to store it in the **employee trainings data store** for the purpose of making each employee able to view his/her trainings schedules.

❖ **Process (16.0): View scheduled trainings**

Description: **Employee** can view the training schedules (type, date, time) by **receiving** scheduled trainings from **employee training data store**.

❖ **Process (17.0): Update employee job data**

Description: The **HR department sends** the **updated employee job data** (containing what he/she wants to update) to store the change in the **employee job data store**.

❖ **Process (18.0): Log out**

Description: All normal **employees**, **managers**, **financial employees** and **HR employees** must log out by **sending log out request**. After log out, his/her **departure time** is sent to the **employee attendance data store** in order to automatically calculate leaving early or if he/she didn't logout on time the system will calculate his/her overtime.

- -

Every data store has initial list of data fields

Data field:

Employee job data store: Employee job data (ID, name, department, job grade, job tittle, job description, salary, bonus, housing allowance, phone number, address, date of birth, gender, marital status, courses need to take it, qualifications need to take it, contract), monthly **salary**, **salary deduction**, **penalties**, and **salary increase**.

Employee appraisal data store: Employee appraisal, and the **date of appraisal**.

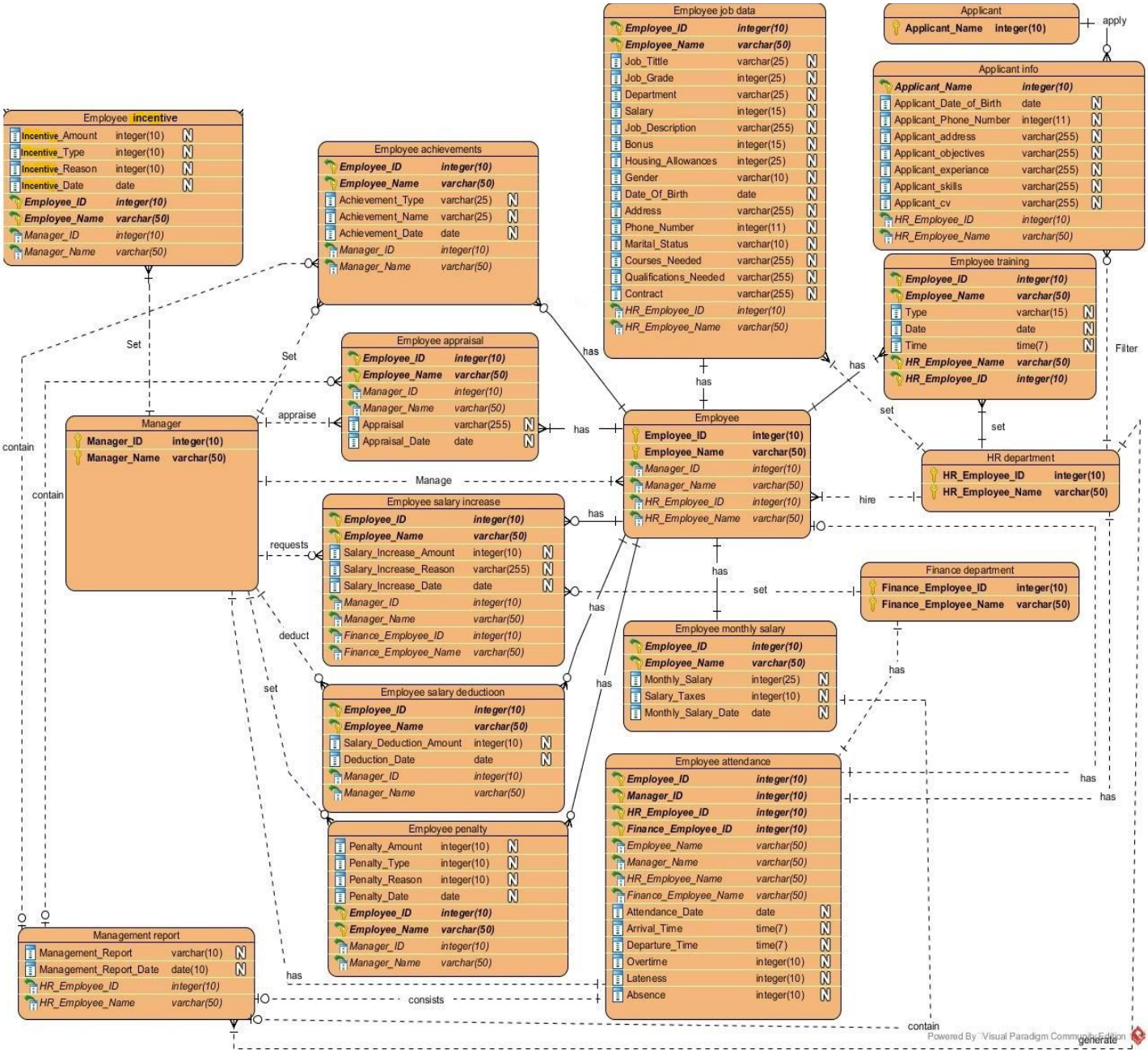
Employee achievements: Courses taken, **qualifications taken**, **accomplishment made**, and **the date of the achievement made**.

Employee trainings: Training type (course, qualification, seminar or etc.), **training date**, **training time**.

Employee attendance: Employee **arrival time**, **departure time**, **overtime**, **lateness**, **absence**, **early leave** and **attendance date.**

Management report: Management report, and its **date.**

11.0 Entity relationship diagram (ERD)



11.1 ERD Documentation

- **Entity: Employee**

  Primary keys: Employee ID & Employee Name

  Foreign key: Manager ID, Manager Name, HR employee ID & HR employee name

  Attributes: Employee ID, Employee Name

  Relations: Each employee **has** only his/her **employee job data**, **has** only his/her **employee monthly salary**, **has** zero or more **employee salary increase**, **has** zero or more **employee salary deduction**, **has** zero or more **employee penalty, has** only one **employee attendance**, **has** zero or more **employee achievement**, **has** one or more **employee appraisal**, and **has** one or more **employee training**.

- **Entity: Manager**

  Primary keys: Manager ID & Manager Name

  Foreign key: -----

  Attributes: Manager ID, Manager Name

  Relations: Each manager **manage** one or more **employee**, **appraise** one or more in **employee appraisal**, **set** one or more achievements in **employee achievements**, **request** zero or more salary increase in **employee salary increase**, **deduct** zero or more from salary in **employee salary deduction**, **penalize** zero or more in **employee penalty**, and **has** only one **employee attendance**.

- **Entity: HR department**

  Primary keys: HR Employee ID & HR Employee Name

  Foreign key: -----

  Attributes: HR Employee ID & HR Employee Name

  Relations: Each HR employee **hire** one or more **employee**, **set** one or more employee **job data** for new employee, **set** one or more **employee training**, and **has** only one **employee attendance** for which **Applicant** is set by the HR in which HR **set** applicant info for one or more applicant**.**

- **Entity: Finance department**

  Primary keys: Finance Employee ID & Finance Employee Name

  Foreign key: -----

  Attributes: Finance Employee ID & Finance Employee Name

  Relations: Each finance employee **set** one or more salary increase for one or more employee in **employee salary increase**, and **has** only one **employee attendance**.

- **Entity: Applicant**

  Primary keys: Applicant Name

  Foreign key: -----

  Attributes: Applicant Name

  Relations: Each applicant **applies** for a one or more job or role in **applicant info entity**.

- **Entity: Employee job data**

  Primary keys: Employee ID & Employee Name

  Foreign key: HR employee name, HR employee ID

  Attributes: Employee name , ID, address (where he or she lives), Phone number (one or more phone number), date of birth (when is he/she born), gender (male or female), marital status (single or married), department (works for which department) , job grade (his/her job grade), job tittle (his/her job tittle), job description (responsibilities, authorities, etc.), contract (attached contract), courses needed (list of courses needed to be taken), qualification needed (list of qualifications needed to be taken), salary (the approved salary), housing allowances (the amount to be paid every month) & bonus (the amount to be paid every month if needed).

  Relations: **Employee job data** is set by the HR in which HR **set** employee job data for one or more employee**.**

- **Entity: Applicant info**

  Primary keys: Applicant Name

  Foreign key: Applicant name

  Attributes: Applicant name, address (where he or she lives), Phone number (one or more phone number), date of birth (when is he/she born), experiences (such as HR

manager, IT admin, system analyst, etc.), and skills (such as communication, analytical, leadership, presentation, etc.)

Relations: **Applicant applies** for one or more job in applicants' info, and **HR department filter** one or more applicant info in order to hire the best**.**

- **Entity: Employee monthly salary**

  Primary keys: Employee ID & Employee Name

  Foreign key: finance employee name & finance employee ID

  Attributes: Employee monthly salary (approved monthly salary after being calculated), Employee salary taxes (the amount if taxes to be deducted from every monthly salary), and Employee monthly salary date (when the monthly salary is paid)

  Relations: Each employee **has** one only **monthly salary**.

- **Entity: Employee salary increase**

  Primary keys: Employee ID & Employee Name

  Foreign key: Manager ID, manager name, finance employee name & finance employee ID

  Attributes: Employee salary increase amount (approved salary increase amount), and Employee salary increase date (when the salary increase is approved)

  Relations: **Employee salary increase** is set by the finance employee in which Finance set zero or more salary increase, **employee has** one or more **salary increase**, and is **requested** by only one manager.

- **Entity: Employee salary deduction**

  Primary keys: Employee ID & Employee Name

  Foreign key: Manager ID & manager name

  Attributes: Employee salary deduction amount (the amount to be deducted from monthly salary), Employee salary deduction type (the type of deduction such as medical insurance, etc.), and Employee salary deduction date (when the salary deduction is made)

  Relations: **Employee salary deduction** is **deducted** by only one manager. Each employee **has** zero or more **salary deduction**.

- **Entity:** **Employee penalty**

  Primary keys: Employee ID & Employee Name

  Foreign key: Manager ID & manager name

  Attributes: Employee penalty amount (the amount to be deducted from monthly salary), Employee penalty type (the type of penalty which can be manual or automatic penalty), Employee penalty reason (the reason for penalizing the employee), and Employee penalty date (when the salary penalty is made)

  Relations: **Employee penalty** is **deducted** by only one manager. Each employee **has** zero or more **penalty**.

- **Entity:** **Employee incentive**

  Primary keys: Employee ID & Employee Name

  Foreign key: Manager ID & manager name

  Attributes: Employee incentive amount (the amount to be added to monthly salary), Employee incentive type (the type of incentive which can be manual or automatic incentive), Employee incentive reason (the reason for setting incentive for the employee), and Employee incentive date (when the salary incentive is made)

  Relations: **Employee incentive** is **added** by only one manager. Each employee **has** zero or more **incentive**.

- **Entity:** **Employee attendance**

  Primary keys: Employee ID, Manager ID, HR Employee ID, Finance Employee ID

  Foreign key: Employee ID, Employee Name, Manager ID, Manager Name, HR Employee ID, HR Employee Name, Finance Employee ID& Finance Employee Name

  Attributes: Month attendance (the attendance of which month), Employee arrival time (when is he/she arrived), Employee departure time (when is he/she left), Employee overtime (the total no. of overtime hours), Employee lateness (the total no. of lateness hours), Employee absence (the total no. of absence days), Employee early leave (the total no. of early leave hours)

  Relations: Each employee, manager, HR employee and finance employee **has** only one **employee attendance**.

- **Entity: Employee achievements**

  Primary keys: Employee ID & Employee Name

  Foreign key: Manager Name & Manager ID

  Attributes: Achievement type (course, qualification or accomplishment made), Achievement name (such as java, CMA, etc.), and achievement date (when achievement is made)

  Relations: Employee achievements are set by one Manager after approving it. Each employee **has** zero or more **employee achievement**.

- **Entity: Employee appraisal**

  Primary keys: Employee ID & Employee Name

  Foreign key: Manager Name & Manager ID

  Attributes: Employee appraisal (attached appraisal document), and Employee appraisal date (when appraisal is made)

  Relations: Employee appraisal is set by one Manager. Each employee **has** zero or more **employee appraisals.**

- **Entity: Employee trainings**

  Primary keys: Employee ID & Employee Name

  Foreign key: HR Employee Name & HR Employee ID

  Attributes: Training type (course, qualification, seminar or etc.), training date (when training is made), and training time (the time for the training)

  Relations: Employee training is set by on HR employee. Each employee **has** one or more **employee training.**

- **Entity: Management report**

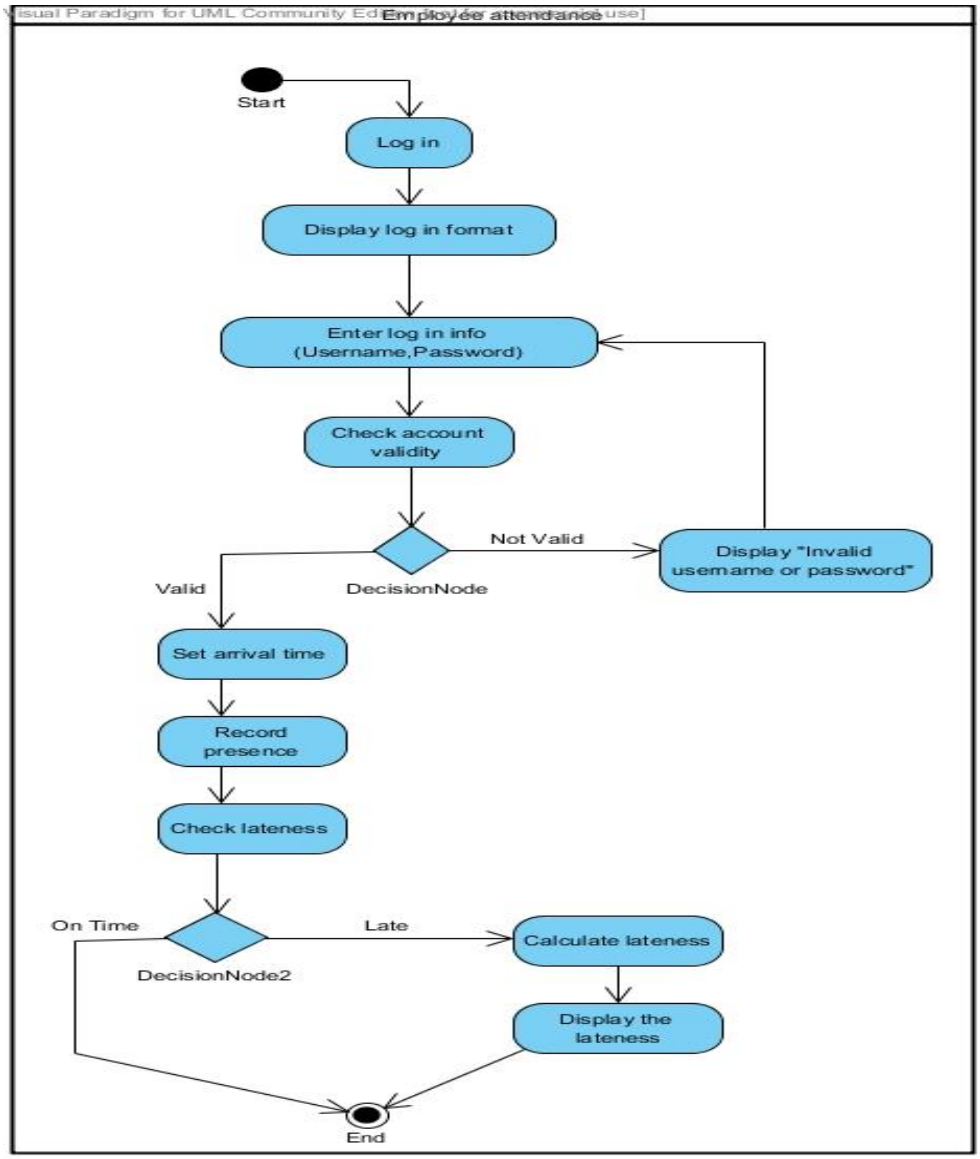
  Primary keys: Management report date

  Foreign key: HR Employee Name & HR Employee ID

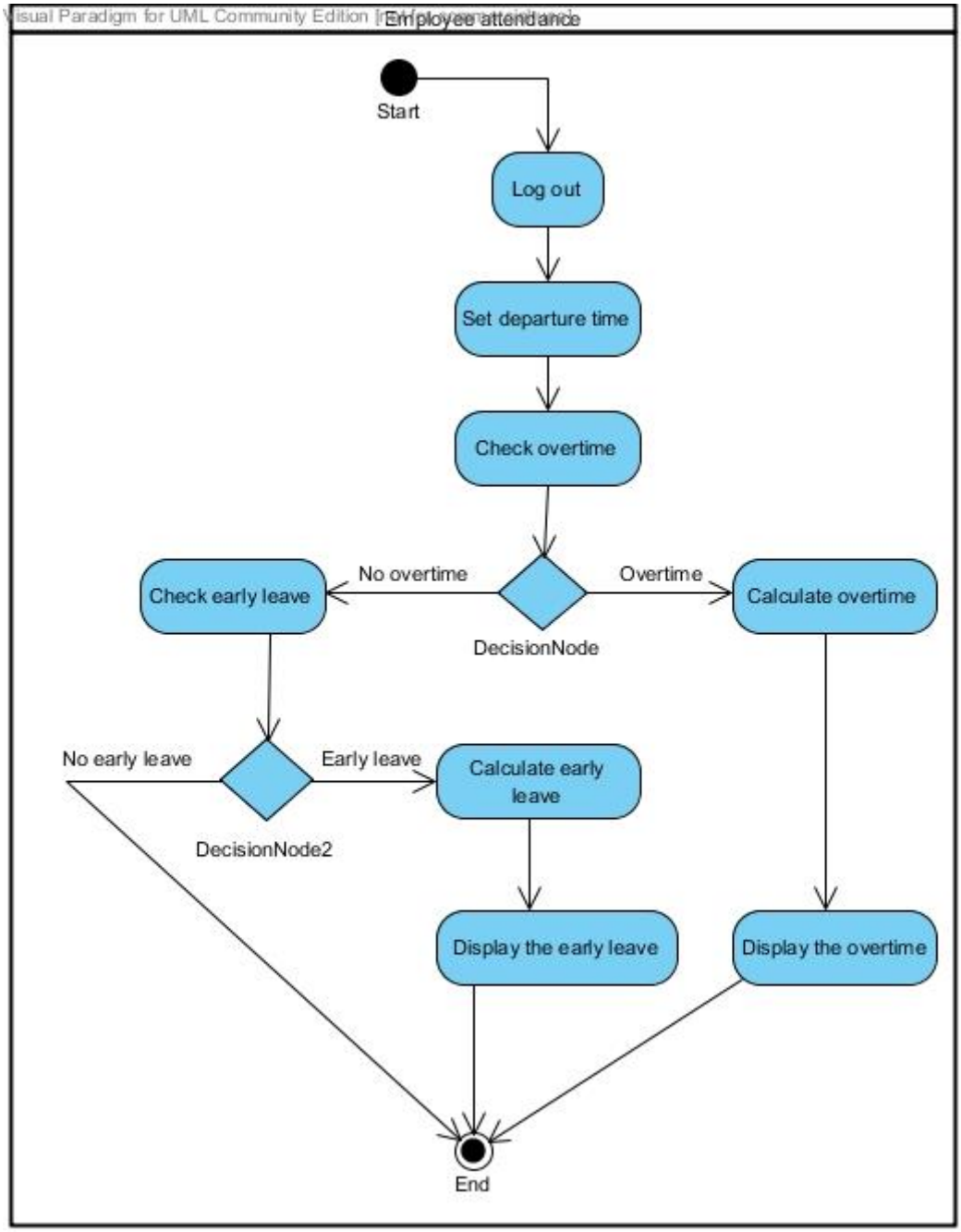
  Attributes: Management report and Management report date

  Relations: Management report is **generated** by one HR employee. Management report **contains** zero or more **employee achievement**, zero or more **employee appraisal,** one only **employee monthly salary,** and one or more **employee attendance**

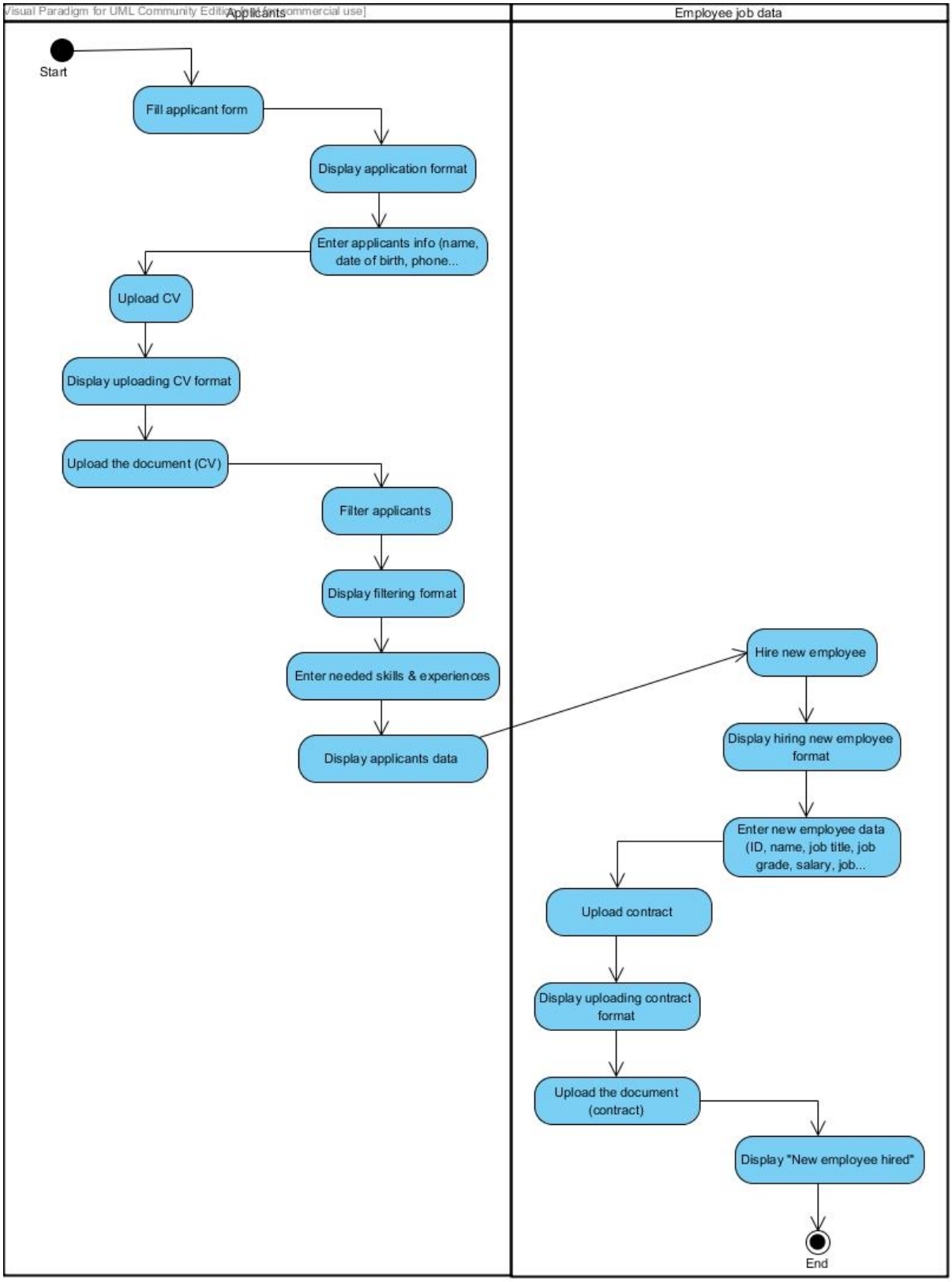
12.0 Activity model



When the employee wants to **log in** to the system, the system first **displays log in format** in order for the employee to **enter his/her username and password**. Then the system **checks the account validity** in which either *the account (username or password)* is **not valid** so the system **displays "Invalid username or password"** and return to the screen to enter again the username and password or *the account (username and password)* is **valid** so the system **set his/her arrival time** which is the time that he/she logged into the system, **record presence** in order to calculate his/her monthly absence, and finally **check lateness** in which either *the employee* is **late** so the system **calculates his/her lateness** and then **displays his/her lateness** then he/she access the system features or *the employee* is **on time** so he/she directly access the system features.



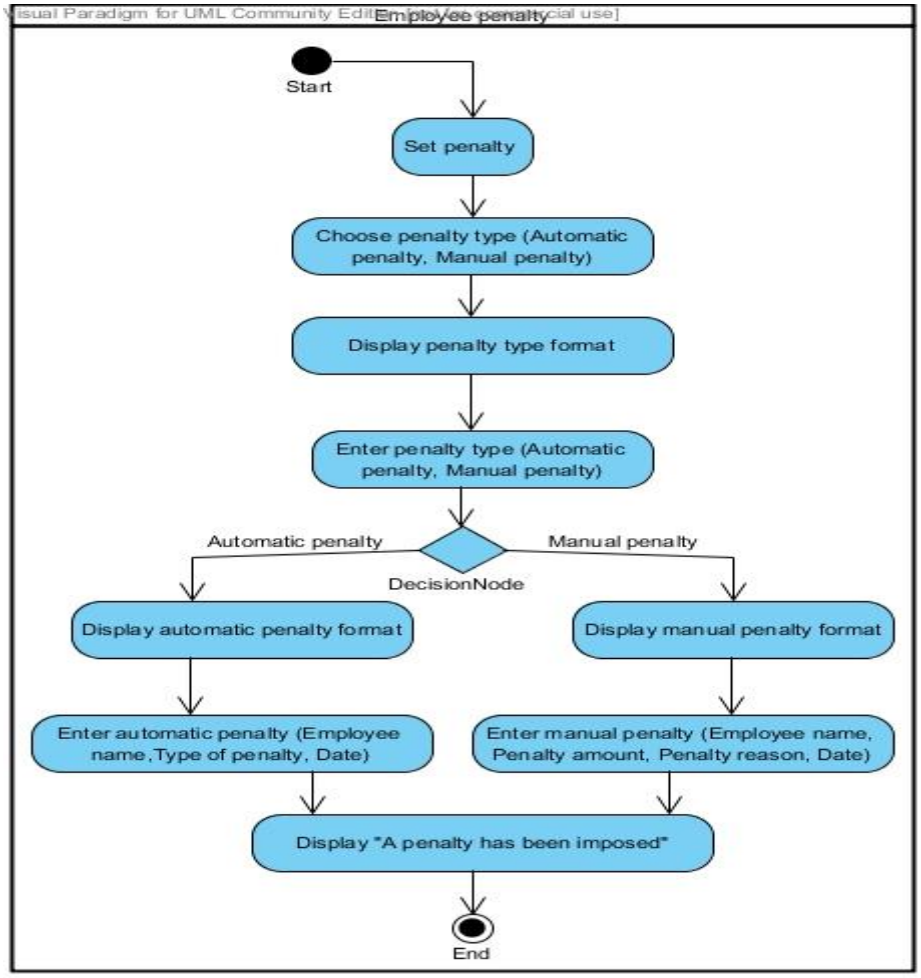
When the employee wants to **log out** from the system, the system first **set his/her departure time** which is the time that he/she logged out from the system Then the system **checks overtime** in which he/she logged out form the system after the stated time to leave work for which either *there* is **overtime** so the system **calculates his/her overtime** and then **displays his/her overtime** or *there* is **no overtime** so the system then **checks early leave** in which he/she logged out from the system before the stated time to leave work for which either *there* is **early leave** so the system **calculates his/her early leave** and then **displays his/her early leave** or *there* is **no early leave** so he log out directly from the system without calculating overtime or early leave because he/she logged out on the stated time to leave work.


Visual Paradigm for UML Community Edition [not for commercial use]
Applicants
Employee job data
Start
Fill applicant form
Display application format
Enter applicants info (name, date of birth, phone...
Upload CV
Display uploading CV format
Upload the document (CV)
Filter applicants
Display filtering format
Enter needed skills & experiences
Display applicants data
Hire new employee
Display hiring new employee format
Enter new employee data (ID, name, job title, job grade, salary, job...
Upload contract
Display uploading contract format
Upload the document (contract)
Display "New employee hired"
End

When Applicants want to apply for role or job in the company, they **fill the applicant form** so the system **displays application format** in order to **enter applicants' info** (the applicant name, date of birth, phone number, address, objectives, experiences & skills) then they **upload the CV** so the system **displays uploading CV format** in order to **upload the CV.** Therefore, the HR department **filter all applicants** based on their skills & experiences needed for a specific role or job then receive from the system the applicants who have the needed skills & experiences for the specific role or job so the system **displays filtering format** in order to **enter applicants' needed experiences & skills** and **display the applicant's data** who meets the chosen skills & experiences.

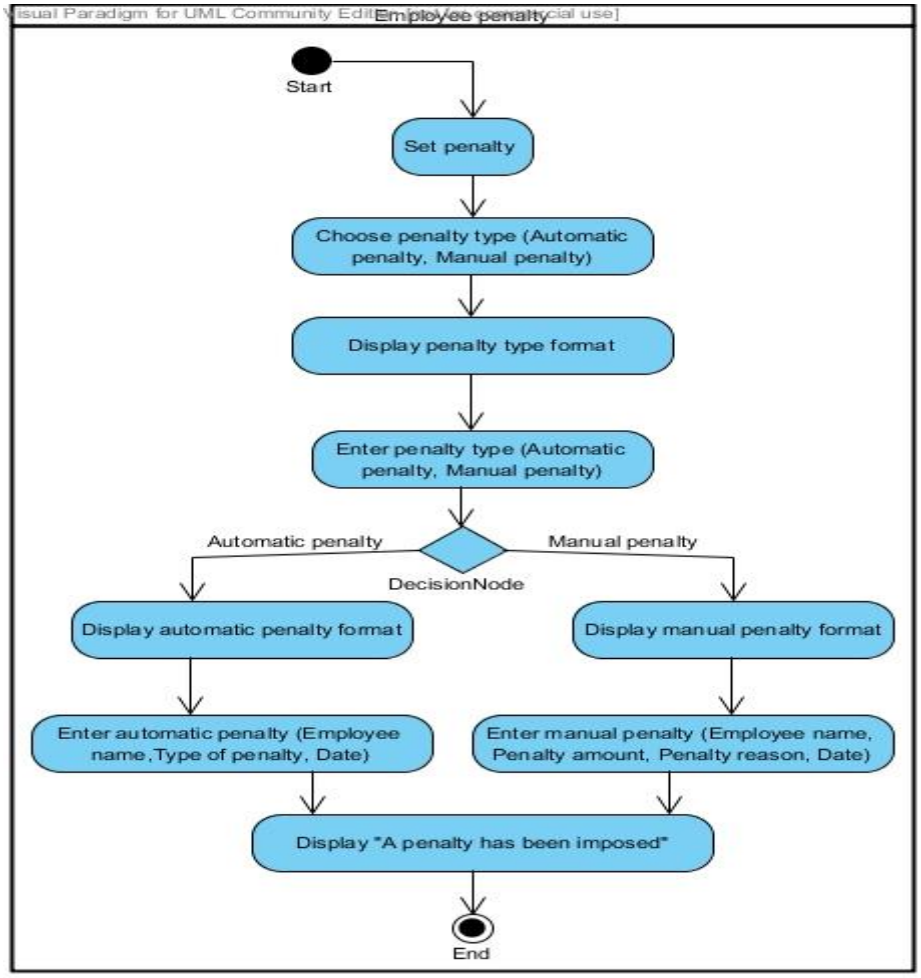
After choosing between the candidates to hire the best employee, the HR department **hire the new employee** so the system **displays hire new employee format** in order to **enter the new employee data** (ID, name, job title, job grade, salary, job description, department, bonus, housing allowance, gender, date of birth, phone number, marital status, courses needed, qualifications needed), and finally HR **upload the new employee contract** so system **displays upload contract format** in order to **upload the new employee contract** (To have a soft copy form his/her contract on the system).

Finally, after entering all new employee job data, the system **displays "New employee hired"** in order to inform that this new employee job data had successfully saved in the system for future use.



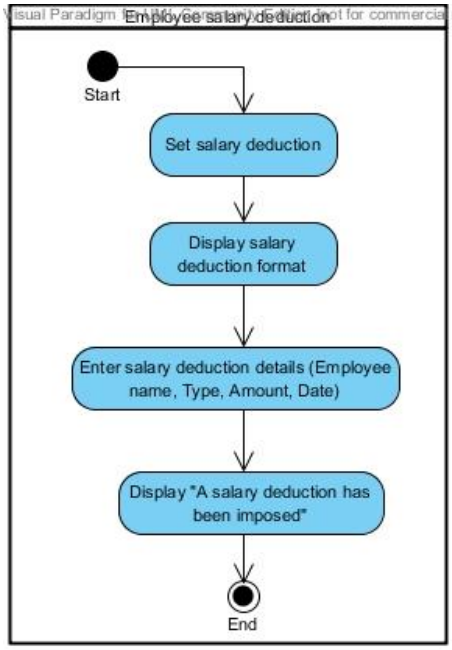
When a manager wants to **set a penalty** on an employee, he/she must **choose the penalty type** (Automatic penalty or Manual penalty) so the system **displays penalty type format** in order to enter penalty type in which he/ she enter Automatic penalty or Manual penalty for which if he/she choose either **automatic penalty** the system **displays automatic penalty format** in order to **enter the automatic penalty** by choosing the employee name (the employee whom the manager want to penalize), date and type such as penalty due to incompliance with the company's policy, penalty due to incompliance with the manager's decision, penalty due to incompliance with his/her team, etc. that *manager selects the type of penalty the system will automatically deduct the amount stated for this penalty type*, or **manual penalty** the system **displays manual penalty format** in order to **enter the manual penalty** by choosing the employee name (the employee whom the manager want to penalize), date, entering the penalty amount to be deducted and entering the reason for imposing penalty on this employee.

Finally, the system **displays "A penalty has been imposed"** to insure that the penalty has been deducted successfully from the employee.

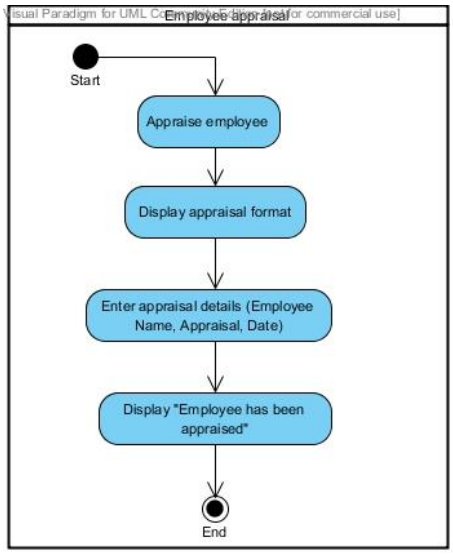


When a manager wants to **set a incentive** on an employee, he/she must **choose the incentive type** (Automatic penalty or Manual incentive) so the system **displays incentive type format** in order to enter incentive type in which he/ she enter Automatic incentive or Manual incentive for which if he/she choose either **automatic incentive** the system **displays automatic incentive format** in order to **enter the automatic incentive** by choosing the employee name (the employee whom the manager want set incentive for), date and type that *manager selects the type of incentive the system will automatically add the amount stated for this incentive type*, or **manual incentive** the system **displays manual incentive format** in order to **enter the manual incentive** by choosing the employee name (the employee whom the manager want to set incentive for), date, entering the incentive amount to be added and entering the reason for imposing incentive for this employee.

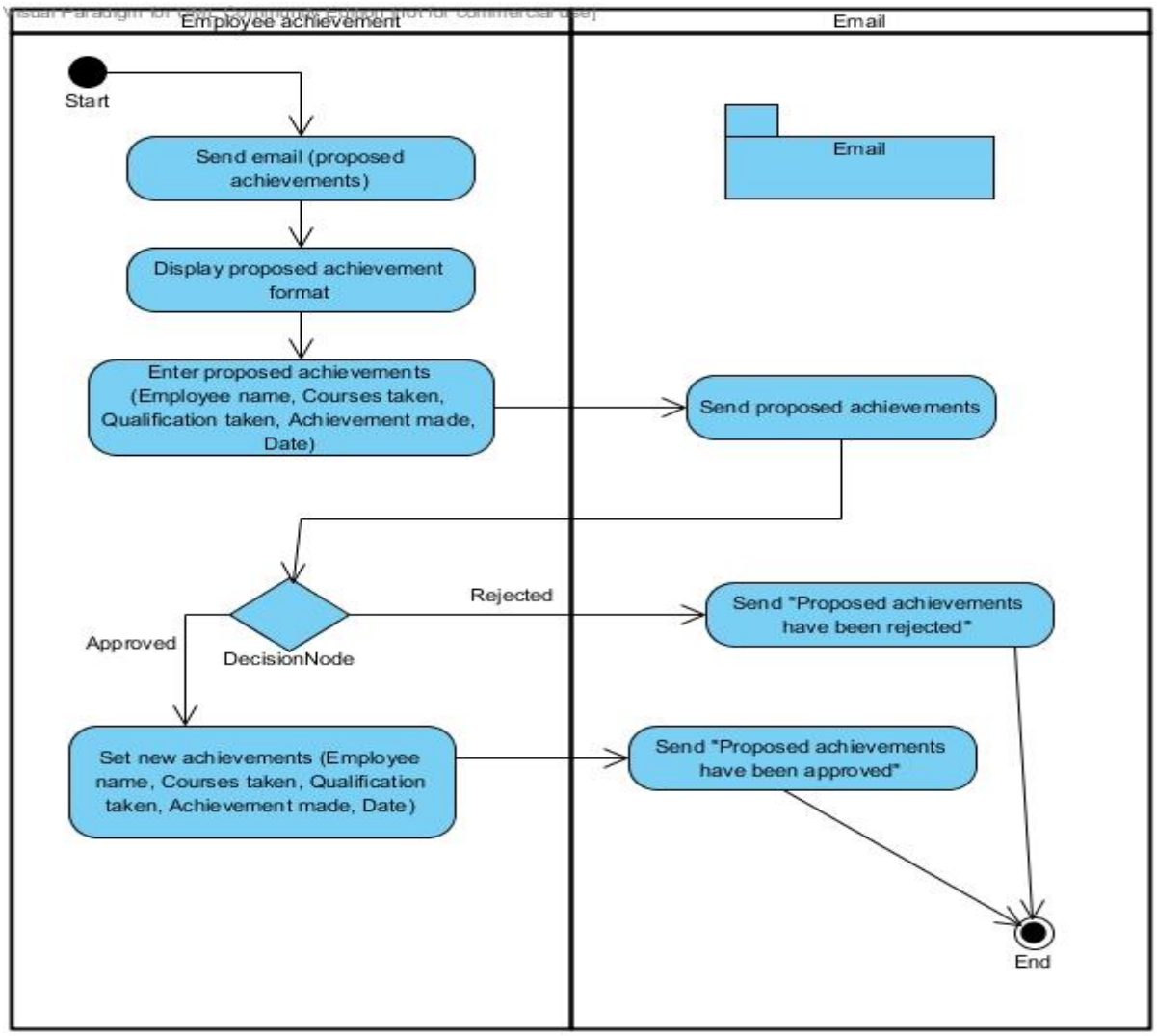
Finally, the system **displays "An incentive has been imposed"** to insure that the incentive has been added successfully to the employee monthly salary.



When a manager wants to **set a salary deduction**, the system **displays salary deduction format** in order to **enter salary deduction details** such as Type (medical insurance, social security insurance, etc.), Amount, Employee name and Date. Then the system **displays "A salary deduction has been imposed"** to ensure that the salary deduction is imposed successfully.

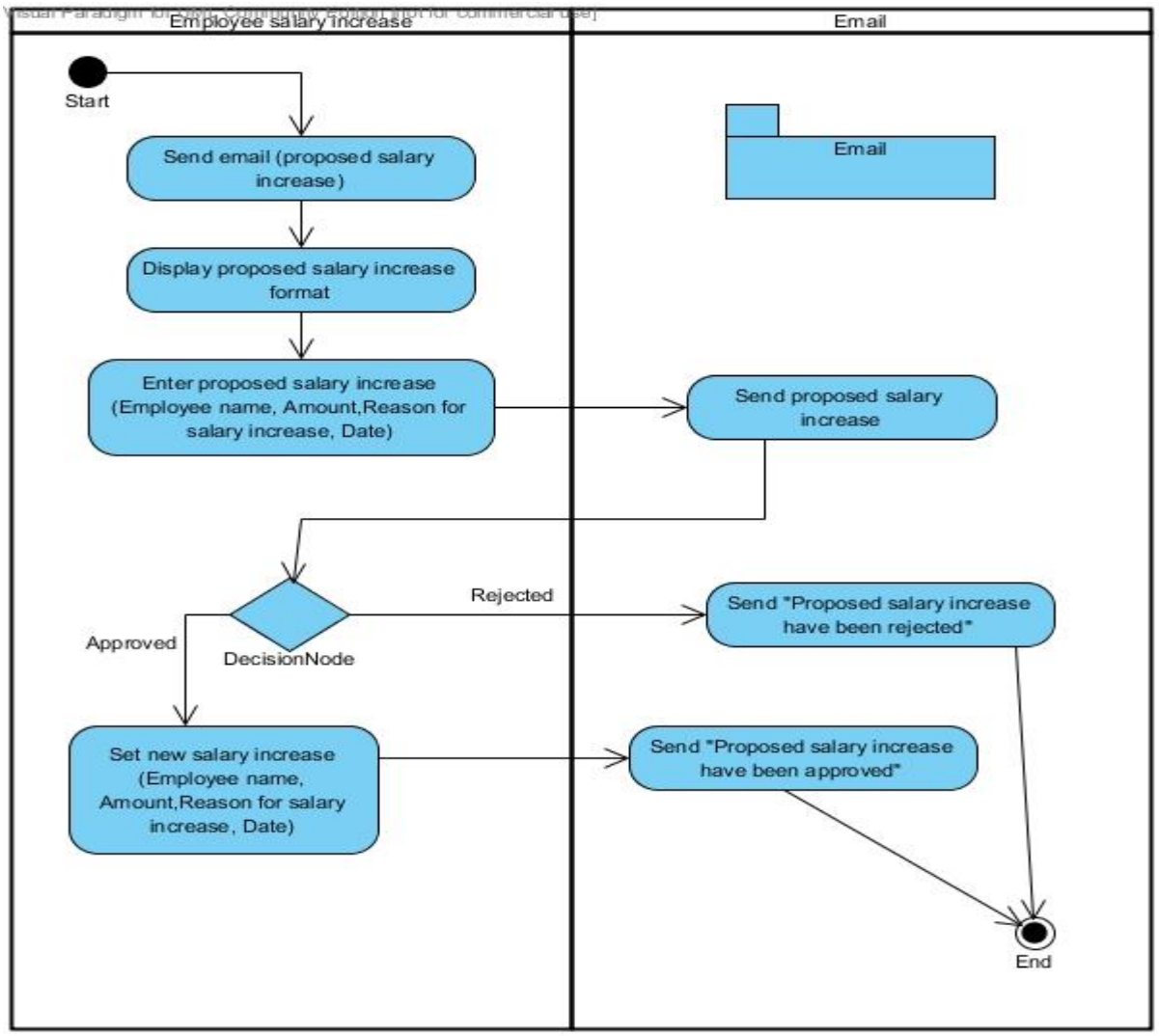


When a manager wants to **appraise an employee**, the system **displays appraisal format** in order to **enter appraisal details** such as Appraisal (the appraisal itself containing the what the manger want to write in this employee appraisal), Employee name and Date. Then the system **displays "Employee has been appraised"** to ensure that the employee is appraised successfully.



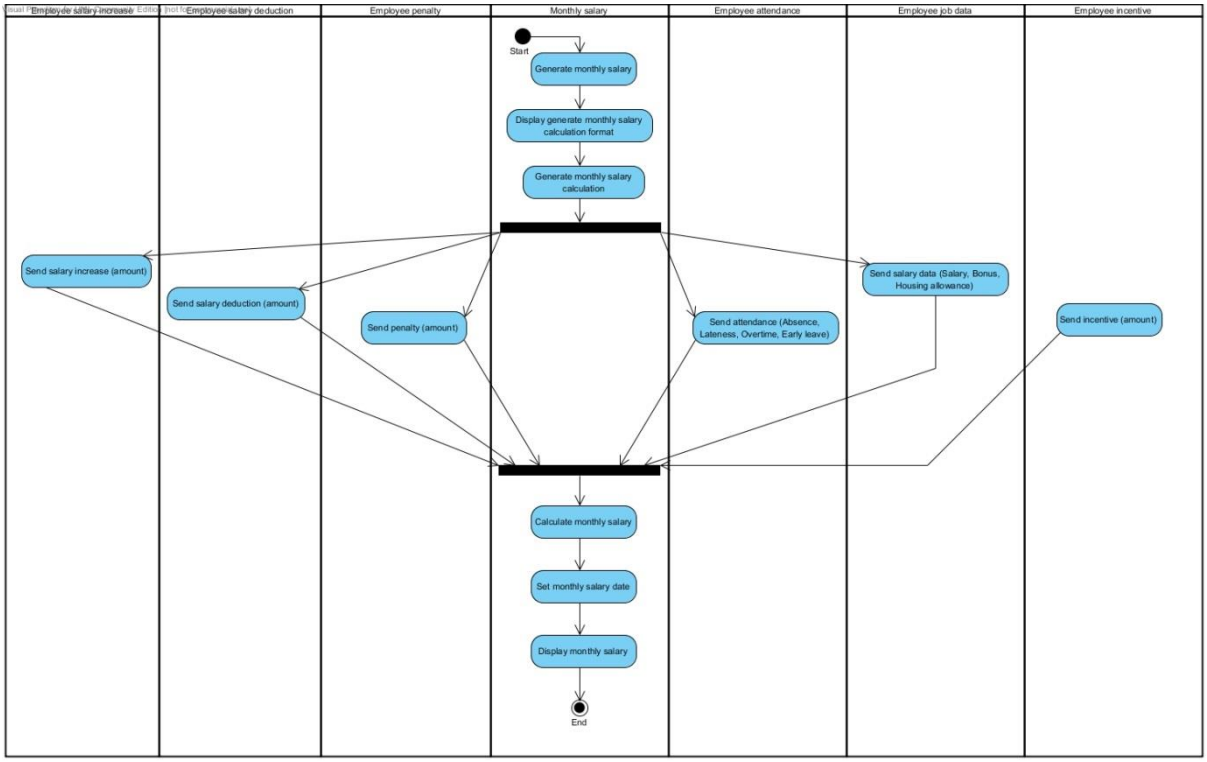
When an employee wants **to send his/her email with his/her proposed new achievements** in order for the manager to approve them and set those new achievements, the system **displays proposed achievement format** in order to **enter proposed achievements** (Employee name, courses taken, qualifications taken, achievement made, date of the achievement). Then **send the proposed achievements** via ***email*** to his/her direct manager in order to approve or reject the proposed achievements.

Finally, either the manager **approve** the proposed achievements he/she must **set those new achievements** (Employee name, courses taken, qualifications taken, achievement made, date of the achievement) and then **send "Proposed achievements have been approved"** via ***email*** to the *employee* to inform him/her that his/her proposed achievements have been approved or the manager **reject** the proposed achievements he/she should **send "Proposed achievements have been rejected"** via ***email*** to the *employee* to inform him/her that his/her proposed achievements have been rejected.



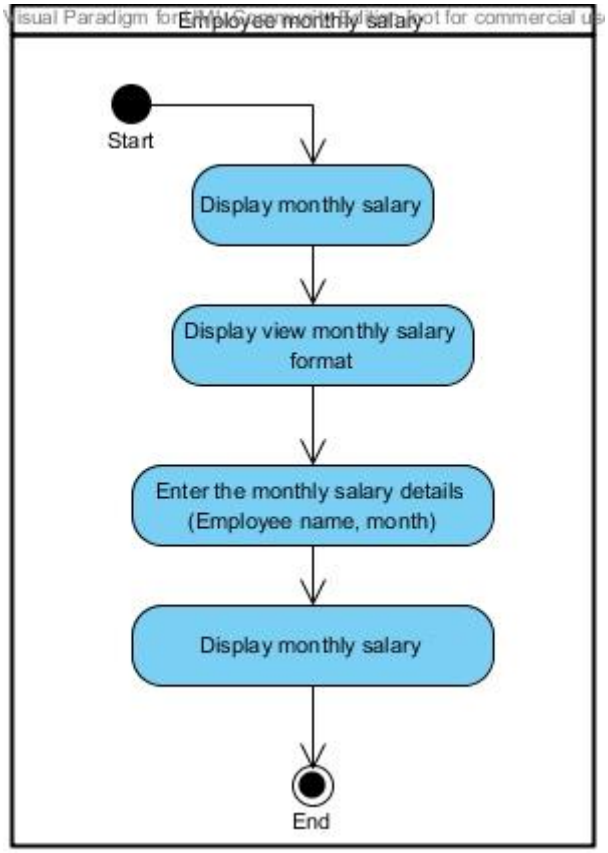
When a manager wants to **send email with the proposed salary increase for an employee** in order for the finance department to approve them and set that new salary increase, the system **displays proposed salary increase format** in order to **enter proposed salary increase** (Employee name, Amount, Reason for salary increase, Date). Then **send the proposed salary increase** via ***email*** to the finance department in order to approve or reject the proposed salary increase.

Finally, either the finance department **approve** the proposed the salary increase he/she must **set the salary increase** (Employee name, Amount, Reason for salary increase, Date) and then **send "Proposed salary increase has been approved"** via ***email*** to the *manager* to inform him/her that his/her proposed salary increase has been approved or the finance department **reject** the proposed salary increase he/she should **send "Proposed salary increase has been rejected"** via *email* to the *manager* to inform him/her that his/her proposed salary increase has been rejected.

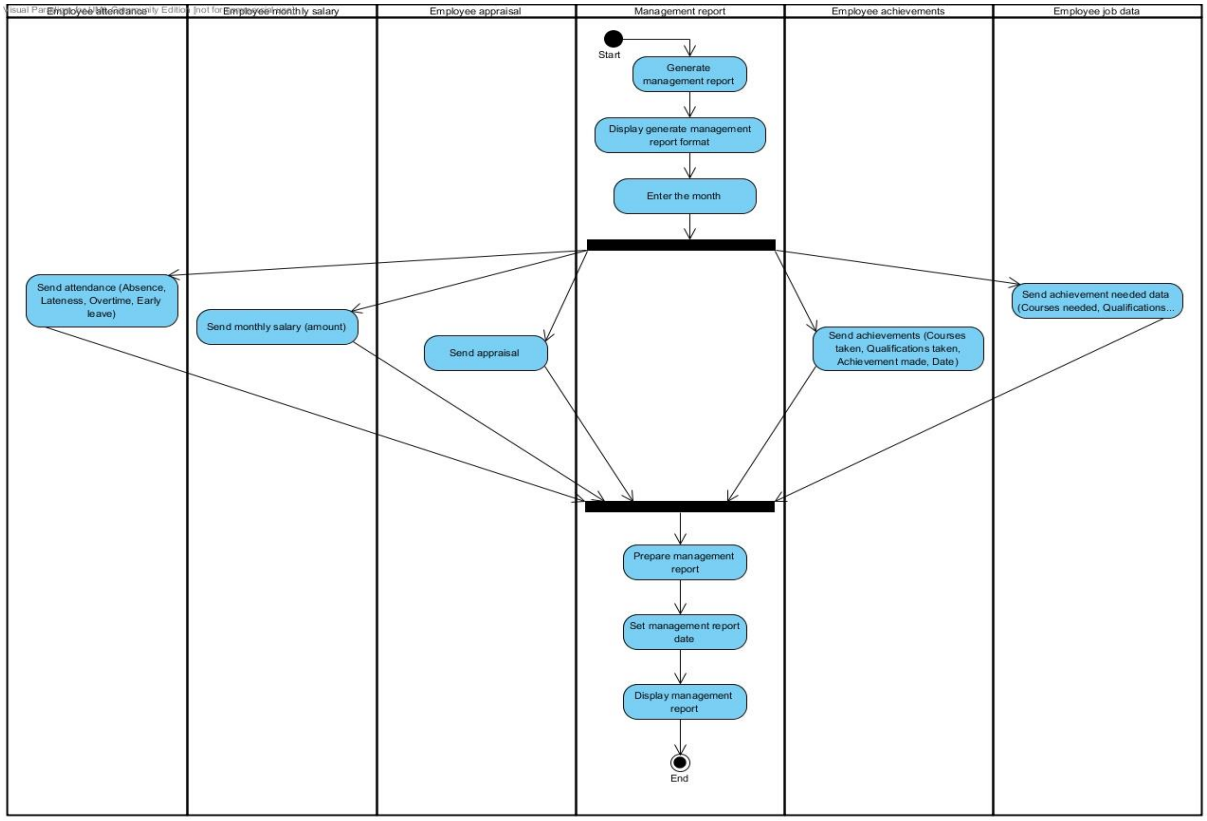


When the HR department wants to **generate the monthly salary** for all employees, the system **displays generate monthly salary format** in order to **generate monthly salary calculation**. After generating monthly salary calculation, *employee salary increase class* **sends salary increase (amount)** if there is any new salary increase in this month for every specific employee, *employee salary increase class* **sends salary deduction (amount)** if there is any new salary deduction in this month for every specific employee, *employee penalty class* **sends penalty (amount)** if there is any new penalty in this month for every specific employee, *employee attendance class* **sends attendance (absence, lateness, overtime, early leave)** if there is any absence, lateness, overtime and/or early leave in this month for every specific employee, *employee job data class* **sends salary data (salary, bonus, housing allowance)** if there is any bonus or housing allowance for every specific employee in this month, and *employee incentive class* **sends incentive (amount)** if there is any new incentive in this month for every specific employee.

Finally, after receiving all the required data in order to calculate the employee monthly salary, the system **calculate monthly salary** for every specific employee based on his/her data sent from the above classes then the system **set the monthly salary date** for this monthly salary to differentiate it from every month monthly salary, and finally the system **display the monthly salary** in order for the HR department to verify that everything is right and been included.

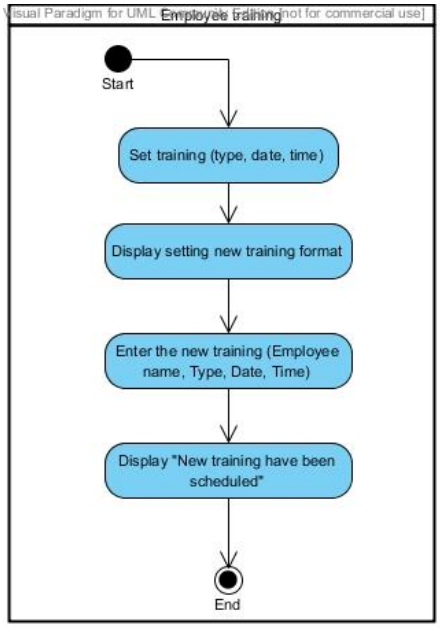


When an employee wants to **display his/her monthly salary**, the system **displays view monthly salary format** in order to **enter the monthly salary details (Employee name, month).** After entering the monthly salary details required, the system **display the employee monthly salary** in order for the employee or the HR department to view it.

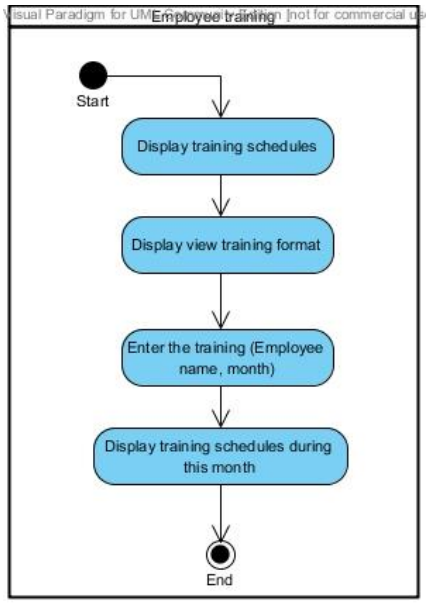


When the HR department wants to **generate management report** for all employees, the system **displays generate management report format** in order to **enter the month** in which you want the management report to be generated for. After generating management report request, *employee attendance class* **sends attendance (absence, lateness, overtime, early leave)** if there is any absence, lateness, overtime and/or early leave in this month for every specific employee, *employee monthly salary class* **sends monthly salary (amount)** in this month for every specific employee, *employee appraisal class* **sends appraisal** if there is any new appraisal in this month for every specific employee, *employee achievement class* **sends achievement (courses taken, qualifications taken, achievement made, date)** if there is any new achievement made in this month for every specific employee, *employee job data class* **sends achievement needed data (courses needed, qualifications needed)** which are the data set by HR department for every specific employee to improve himself/herself.

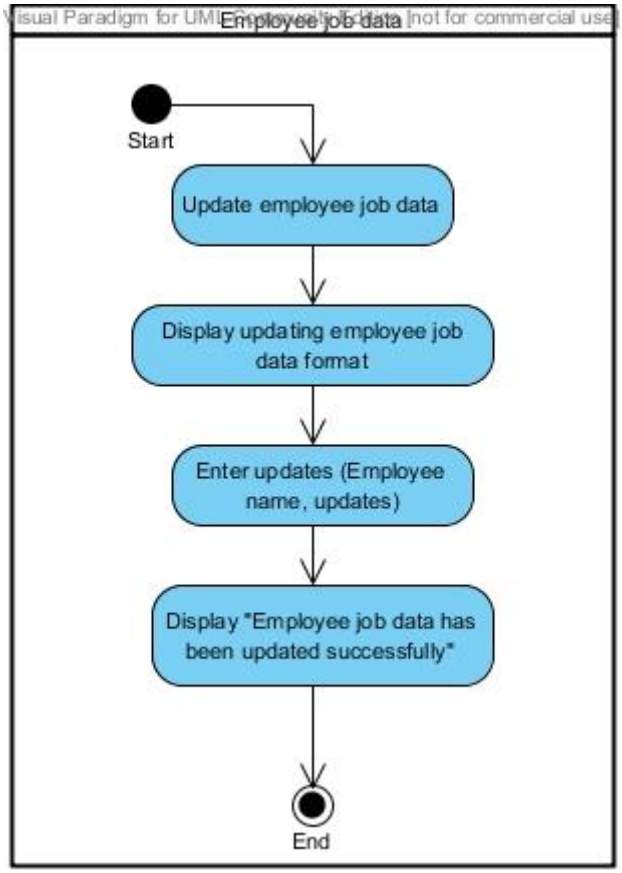
Finally, after receiving all the required data in order to generate the management report, the system **prepare the management report** for every specific employee based on his/her data sent from the above classes then the system **set the management report date** for this management report to differentiate it from every month management report, and finally the system **display the management report** in order for the HR department to verify that everything is right and been included.



When HR department wants to **set new training**, the system **display setting new training format** in order to **enter the new training (Employee name, Type, Date, Time)** in which the HR department choose between different types of training, date, time and assign it to a specific employee for which the system **displays "New training have been scheduled"** to inform that the training have been scheduled successfully.



When an employee wants to **display his/her training schedules**, the system **displays view training format** in order to **enter the training (Employee name, month).** After entering the training details required, the system **displays the training schedules during this month** in order for the employee to view scheduled trainings.



When HR department wants to **updates any employee job data**, the system **display updating employee job data format** in order to **enter updates (Employee name, updates)** in which the HR department updates any job data such as new address, new phone number, new bonus, new housing allowance, new course needed to be taken, etc. for a specific employee for which the system **displays "Employee job data has been updated successfully"** to inform that the employee job data have been updated successfully in the system.

13.0 Validation test

In the case of this educational graduation project proposal, I intentionally did substitute the stake holders in the validation (forward integration) test by the features of already existing systems in the market. This is to secure being with the market needs. In that sense, my two suggested software packages are Paycor, and ADP. These two packages have similar features to my system and also have some different features to mine.

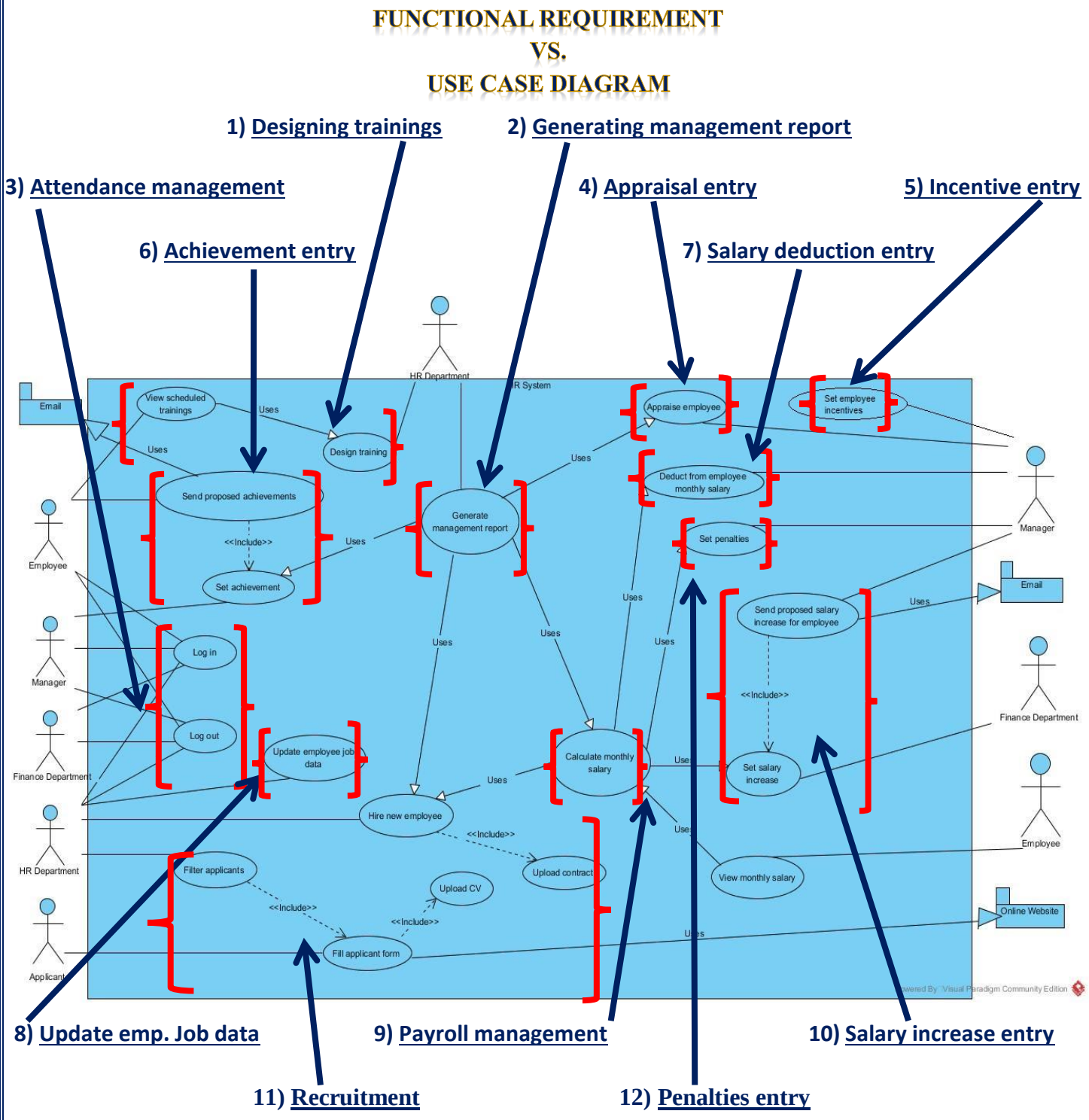
ID	User Requirements	Forward traceability
Paycor	❖ **Recruitment:** refers to the overall process of attracting, selecting and appointing suitable candidates for jobs within an organization ❖ **Payroll management:** is the administration of the financial record of employees' salaries, wages, bonuses, net pay, and deductions. ❖ **Time management:** is the process of organizing and planning the employee attendance (arrival time, departure time, absence, lateness, overtime, early leave, etc.). ❖ **Benefits management:** creating and implementing benefits plans for current employees or offering new plans to job candidates. ❖ **Reporting:** provides relevant and timely HR and Payroll information that supports the operational requirements of departments.	R-1) This requirement is covered by the use cases: Hire new employee, Set employee job data R-2) This requirement is covered by the use cases: Calculate monthly salary, View monthly salary R-3) This requirement is covered by the use cases: Log in, Log out, Prepare employee monthly attendance R-4) This requirement is covered by the use cases: Design training, View scheduled trainings, appraise employee, send proposed achievements, set achievements, set proposed salary increase, Set salary increase

	❖ **Onboarding:** refers to the process of orienting new employees in a manner that aids in overall retention. It goes beyond what we've come to know as orientation. This process focuses on helping employees to become acclimated to their new workplace in a timely fashion and bringing them "on board" with regard to company culture, understanding of job function and overall comfort level.	R-5) This requirement is covered by the use case: Generate management report New-R1) **Salary deduction** This requirement is covered by the use cases: Set salary deduction New-R2) **Penalties** This requirement is covered by the use cases: Set penalties
ADP	❖ **Payroll management:** is the administration of the financial record of employees' salaries, wages, bonuses, net pay, and deductions. ❖ **Time management:** is the process of organizing and planning the employee attendance (arrival time, departure time, absence, lateness, overtime, early leave, etc.). ❖ **Talent management:** refers to the anticipation of required human capital for an organization and the planning to meet those needs. ❖ **Benefits management:** creating and implementing benefits plans for current employees or offering new plans to job candidates.	R-1) This requirement is covered by the use cases: Calculate monthly salary, View monthly salary R-2) This requirement is covered by the use cases: Log in, Log out, Prepare employee monthly attendance R-3) This requirement is covered by the use cases: Design training, View scheduled trainings R-4) This requirement is covered by the use cases: Appraise employee, send proposed achievements, set achievements, set proposed salary increase, Set salary increase

	❖ **Labor management:** is comprised of enterprise tools that help businesses better plan their daily work and processes for better delivery of products and services. ❖ **Reporting:** provides relevant and timely HR and Payroll information that supports the operational requirements of departments. ❖ **Benchmarking:** often used to describe performance assessment is benchmarking which seeks to assess the competences of an organization against "best in class" wherever that is to be found.	R-5) This requirement is covered by the use case: Generate management report New-R1) **Salary deduction** This requirement is covered by the use cases: Set salary deduction New-R2) **Penalties** This requirement is covered by the use cases: Set penalties

14.0 The verification test (Traceability)

Traceability provides the ability to find the origin of each functional requirement in the approved project scope, and to follow the development of each functional requirement as it progresses through the life of a project.

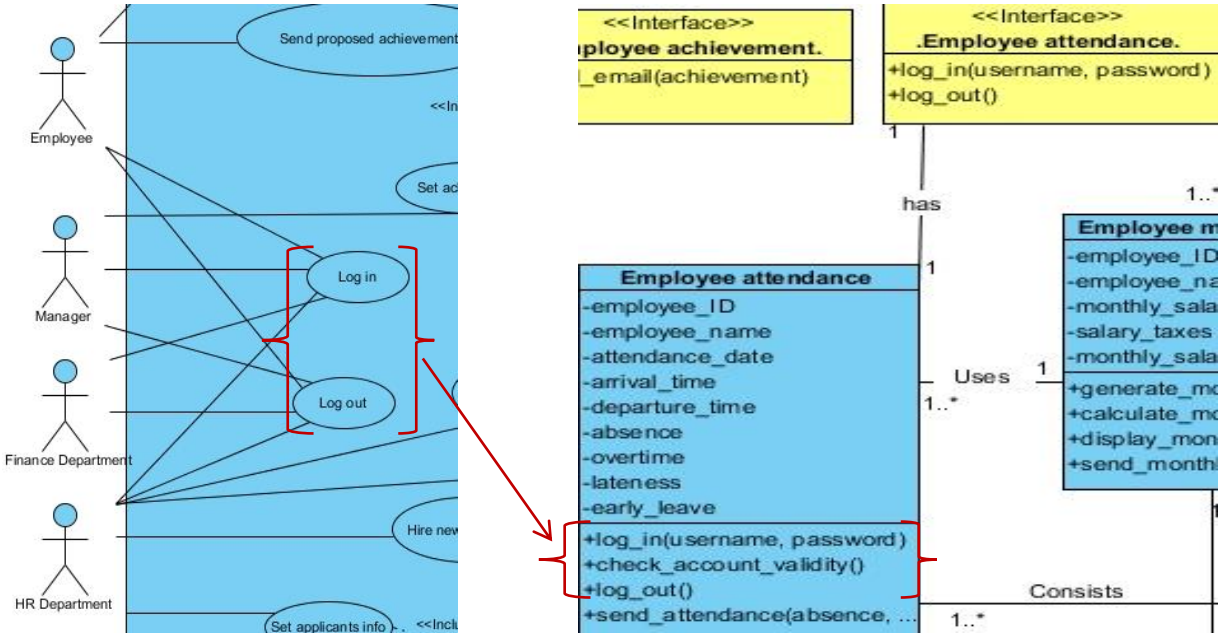


The use case diagram represent the user functional requirements as shown above in which the above use cases represent my project functional requirements represented in the scope of my project such as Employee attendance management, Recruitment & hiring new employees, Employee appraisal entry, Employee salary deduction entry, Employee penalties entry, Employee salary increase entry, Employee achievements entry, Payroll management, Generating management report, Design training for employee, and Update employee job data.

USE CASE DIAGRAM VS. CLASS DIAGRAM

A class or group of classes in an class diagram should work together in order to implement a use case or a group of use cases.

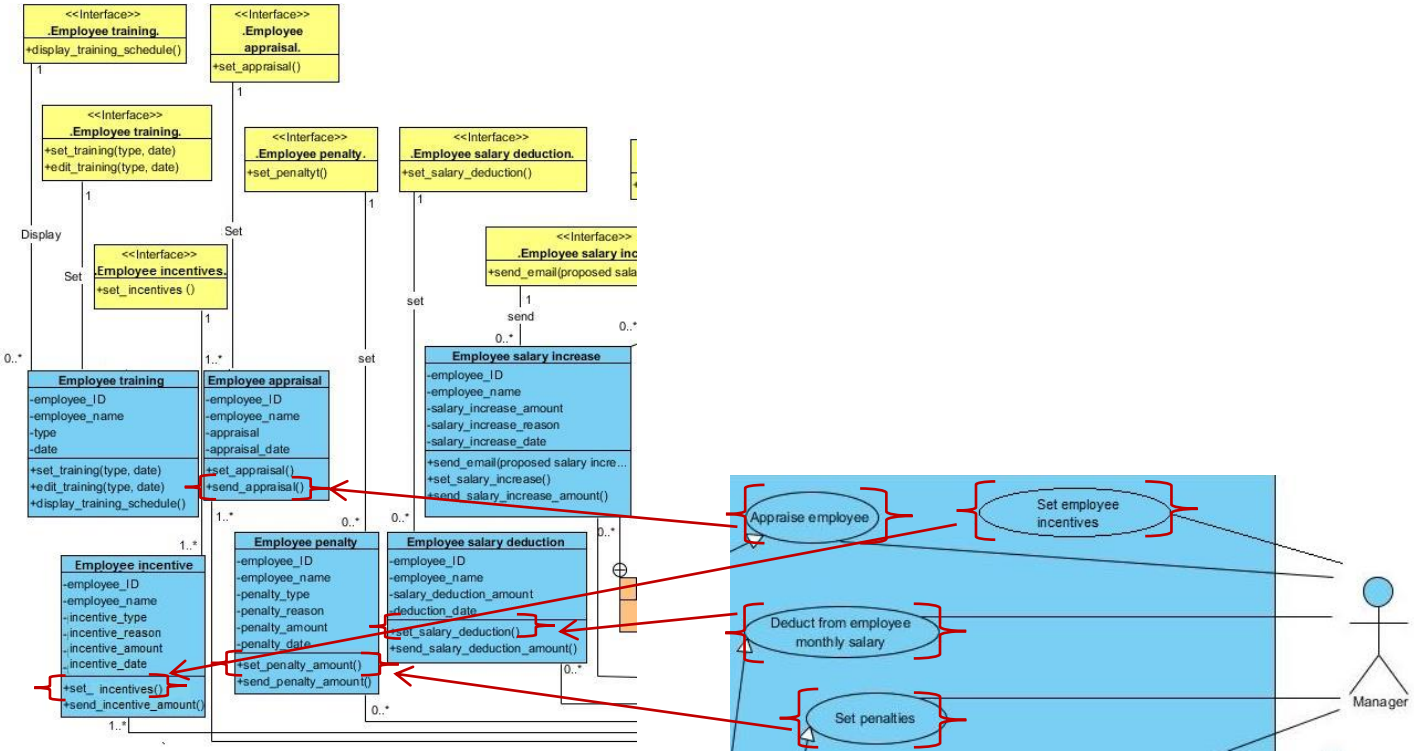
Use Case Diagram **Class Diagram**



The real environment of this business case (Attendance management) contains the following classes which are “Interface” Employee attendance class, and Employee attendance class presented above. This list of classes (“Interface” Employee attendance class, and Employee attendance class) serve the implementation of these use cases (log in & log out) in which the system checks for the account validity when he/she log in to the system to verify the username and password are valid.

Class Diagram

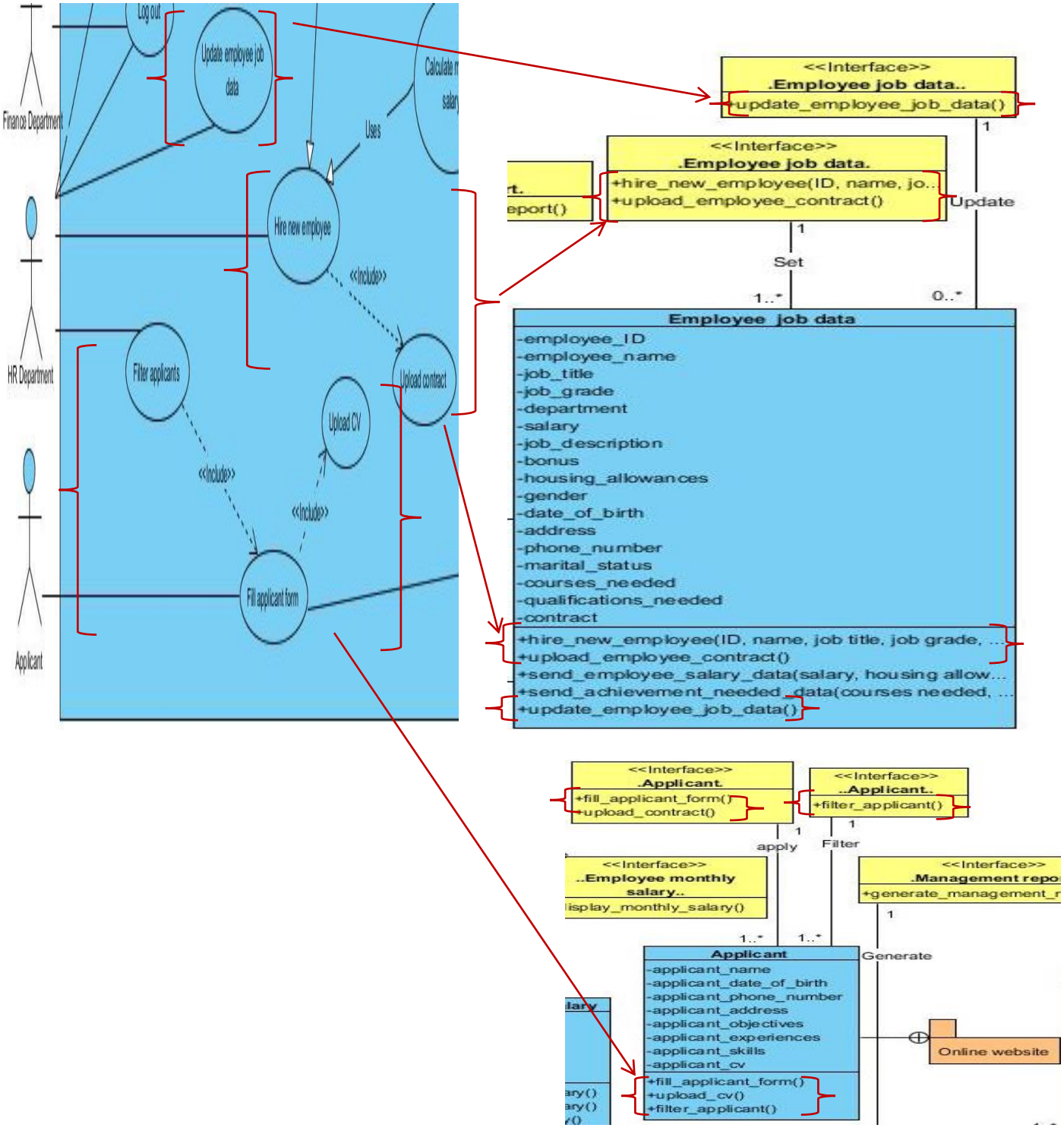
Use Case Diagram



The real environment of those business cases (Appraisal entry, salary deduction entry , incentives entry & penalties entry) contains the following classes which are “Interface” Employee salary deduction class, Employee salary deduction class, “Interface” Employee appraisal class, Employee appraisal class, “Interface” Employee incentive class, Employee incentive class, “Interface” Employee penalty class and Employee penalty class.

This list of classes “Interface” Employee salary deduction class, Employee salary deduction class, “Interface” Employee appraisal class, Employee appraisal class, “Interface” Employee incentive class, Employee incentive class, “Interface” Employee penalty class and Employee penalty class) serve the implementation of these use cases (set employee incentives, appraise employee, deduct from employee salary & set penalties).

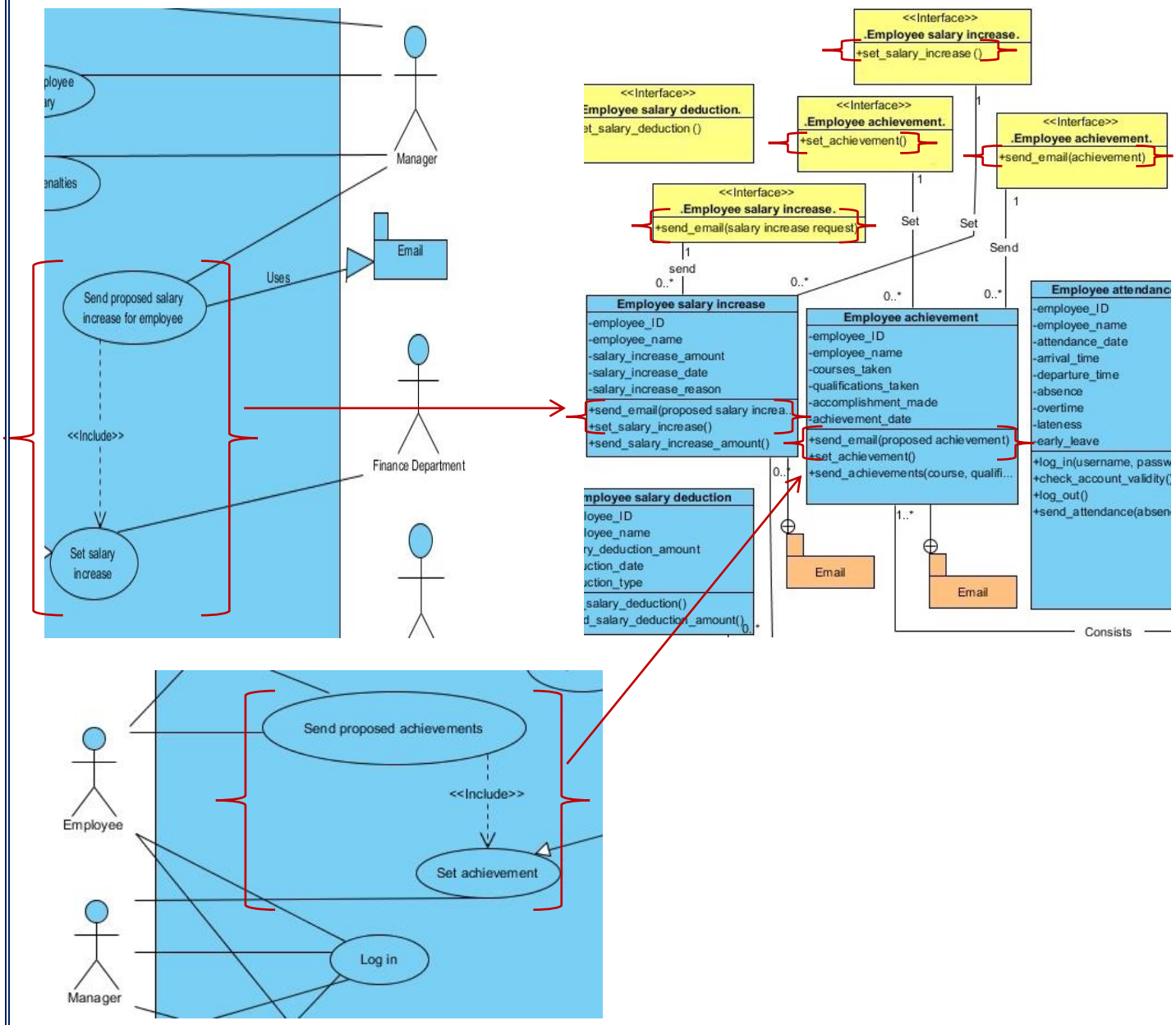
Use Case Diagram **Class Diagram**



The real environment of those business cases (Recruitment & Updating emp. job data) contains the following classes which are “Interface” Employee job data class, Employee job data class, “Interface” Applicant class & Applicant class presented above. This list of classes

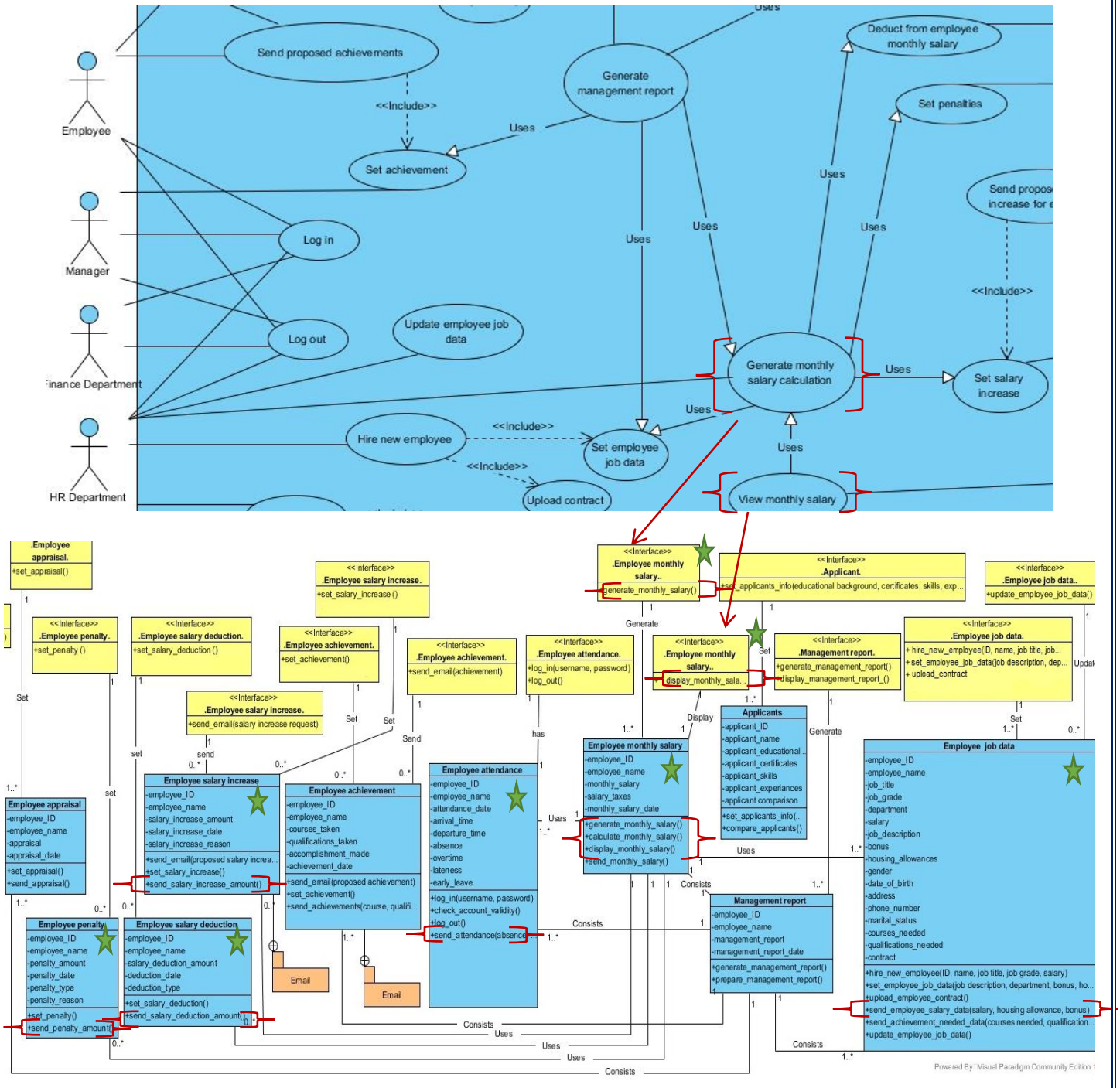
("Interface" Employee job data class, Employee job data class, "Interface" Applicant class & Applicant class) serve the implementation of these use cases (fill applicant form, upload CV, filter applicants, hire new employee, upload contract & update emp. job data).

Use Case Diagram **Class Diagram**



The real environment of those business cases (Achievement entry & Salary increase entry) contains the following classes which are "Interface" Employee salary increase class, Employee salary increase class, "Interface" Employee achievement class & Employee achievement class presented above.

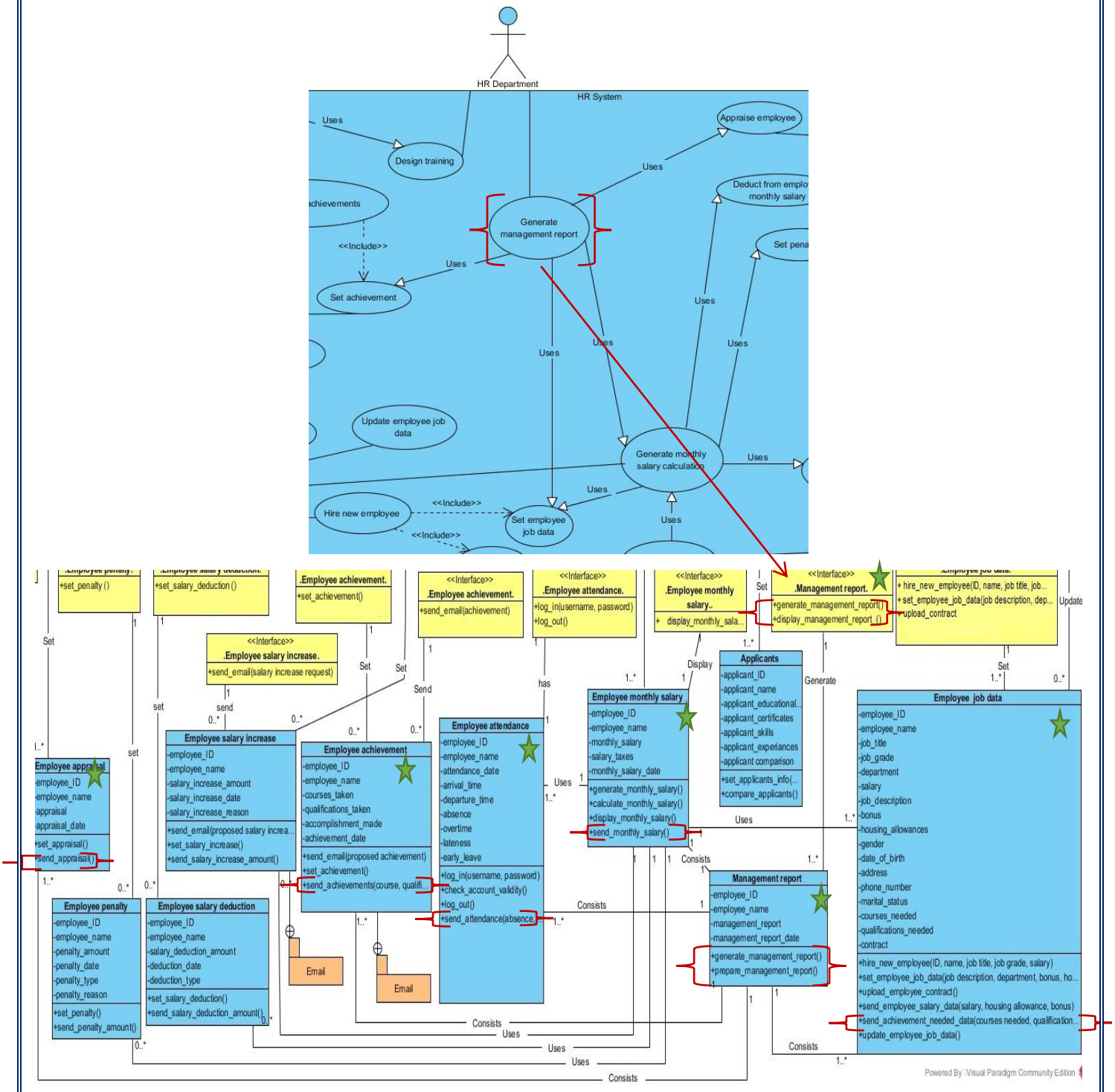
This list of classes “Interface” Employee salary increase class, Employee salary increase class, “Interface” Employee achievement class & Employee achievement class) serve the implementation of these use cases (send proposed salary increase for employee, set salary increase, send proposed achievements, & set achievements).



The real environment of this business case (Payroll management) contains the following classes which are “Interface” Employee monthly salary class, Employee monthly salary

class, Employee penalty class, Employee salary deduction class, Employee salary increase, Employee attendance & Employee job data class presented above.

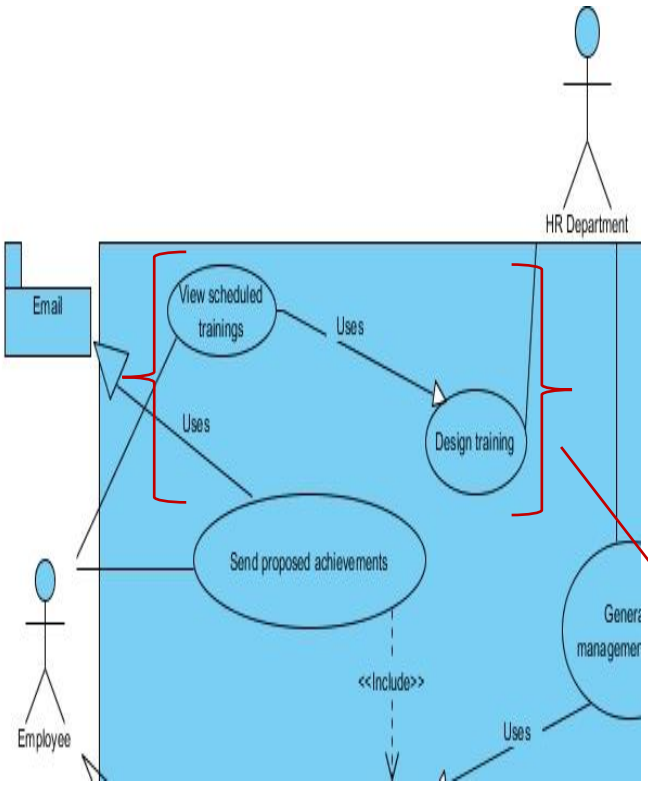
This list of classes ("Interface" Employee monthly salary class, Employee monthly salary class, Employee penalty class, Employee salary deduction class, Employee salary increase, Employee attendance & Employee job data class) serve the implementation of these use cases (generate monthly salary calculation & view monthly salary).



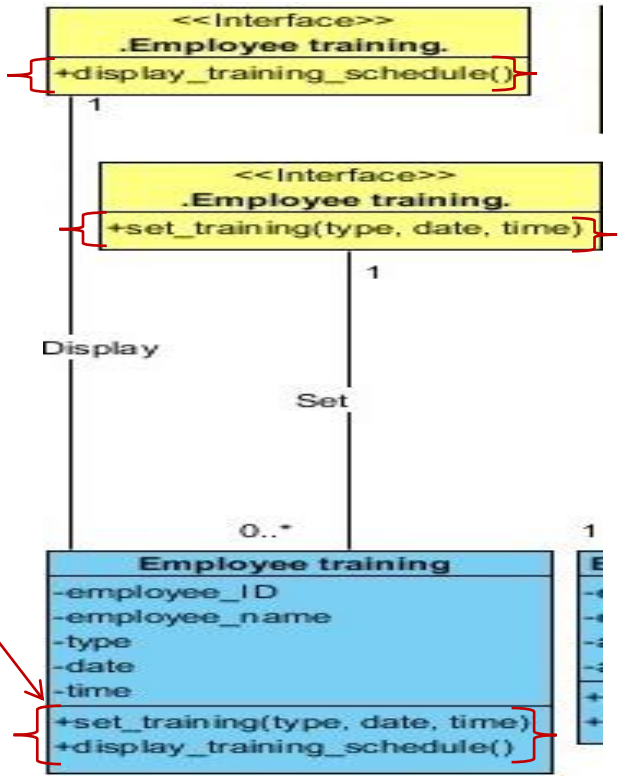
The real environment of this business case (Generating management report) contains the following classes which are “Interface” Management report class, Management report class, Employee appraisal class, Employee achievement class, Employee monthly salary class, Employee attendance class & Employee job data class presented above.

This list of classes “Interface” Management report class, Management report class, Employee appraisal class, Employee achievement class, Employee monthly salary class, Employee attendance class & Employee job data class) serve the implementation of these use cases (generate management report).

Use Case Diagram



Class Diagram

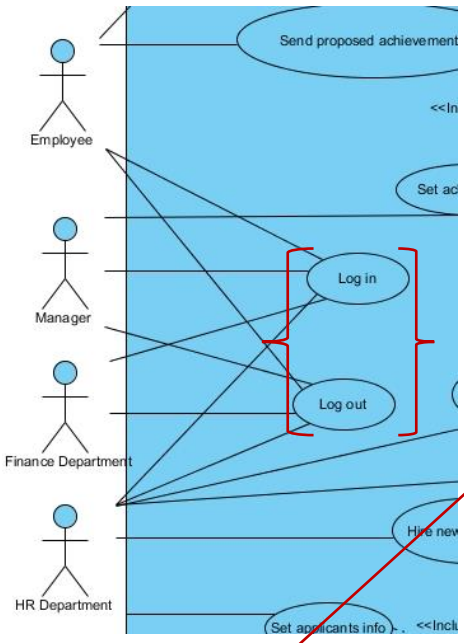


The real environment of this business case (Desiging training) contains the following classes which are “Interface” Employee training & Employee training class presented above. This list of classes (“Interface” Employee training & Employee training class) serve the implementation of these use cases (design training & view scheduled trainings).

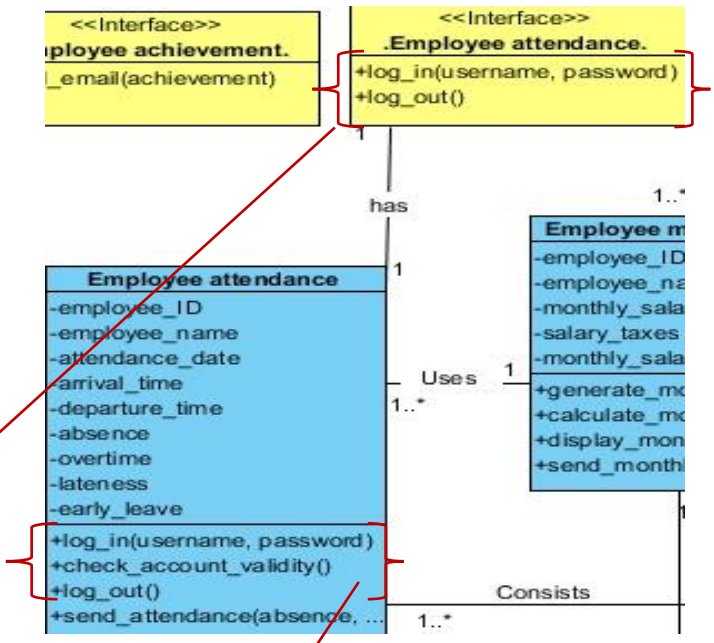
CLASS DIAGRAM VS. SEQUENCE DIAGRAM

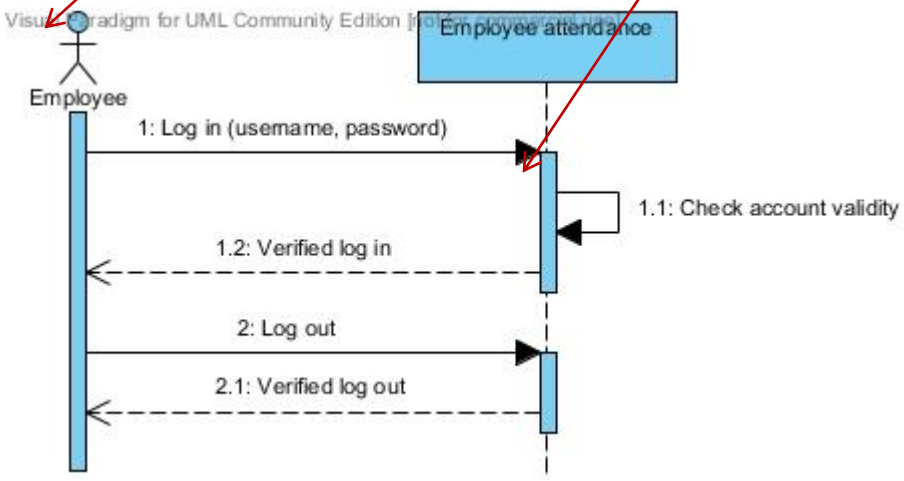
Sequence diagram in which each method between two objects represents an operation to the destination object.

Use Case Diagram



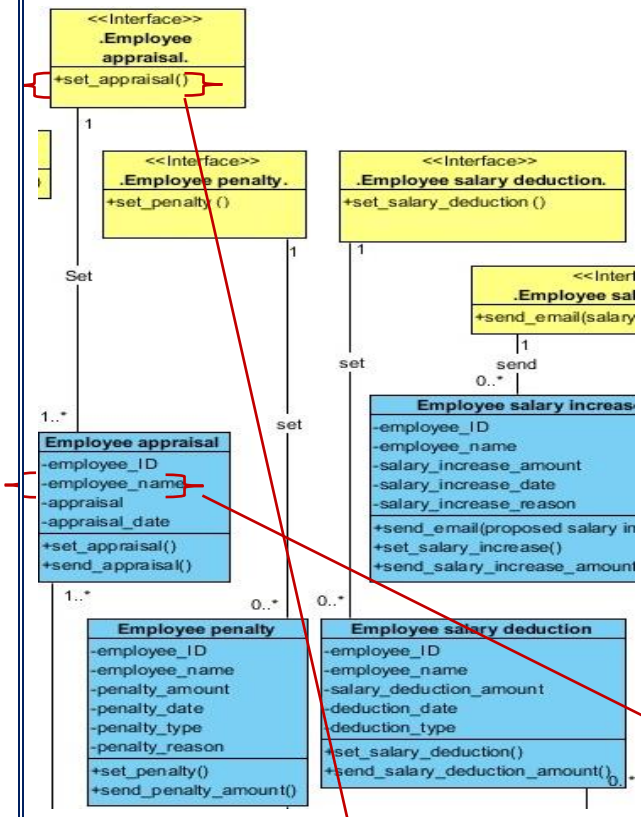
Class Diagram



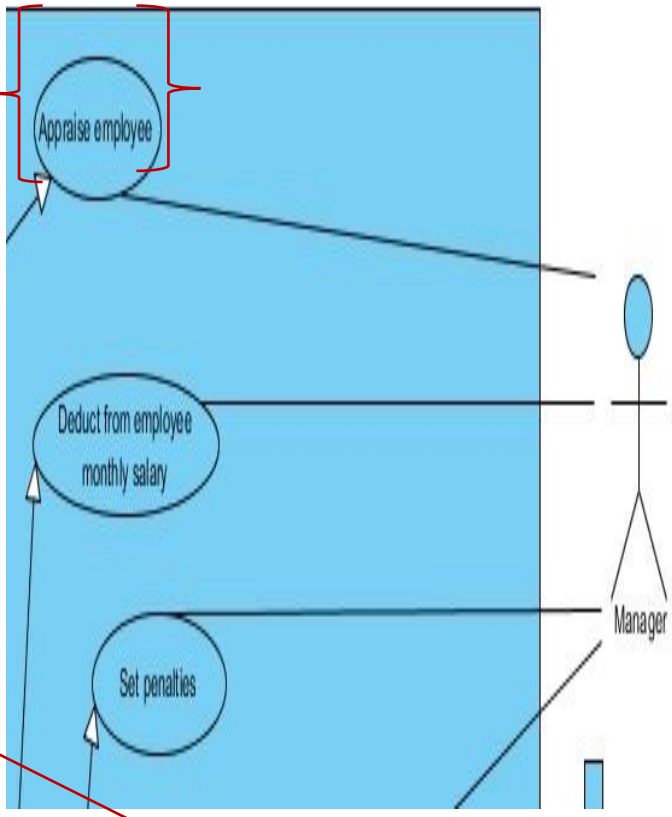


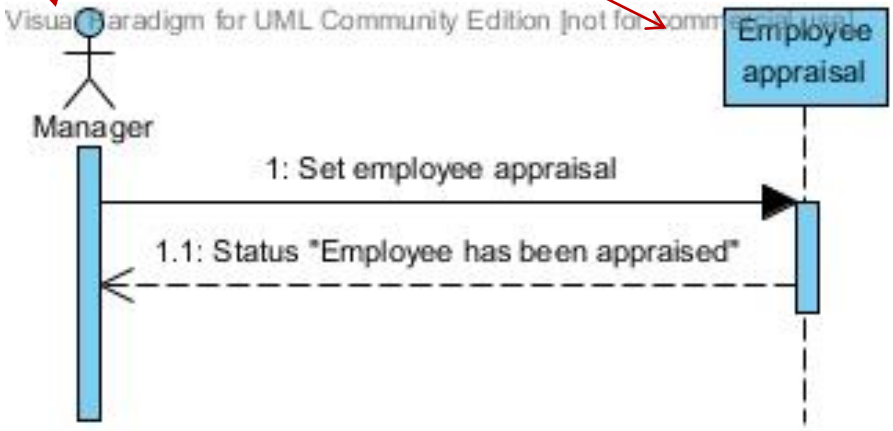
Each method between the employee and employee attendance class represents an operation such as log in, check account valdity & log out to the destination object which is the employee attendance class.

Class Diagram



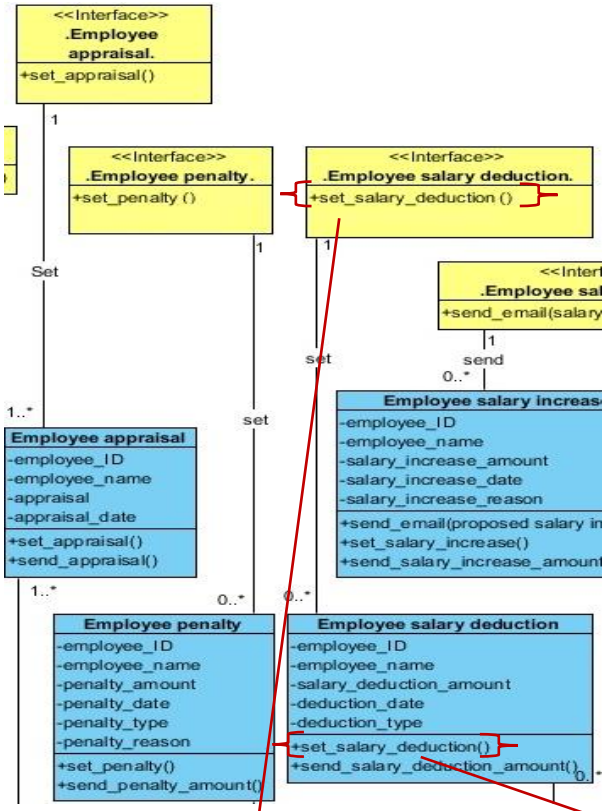
Use Case Diagra



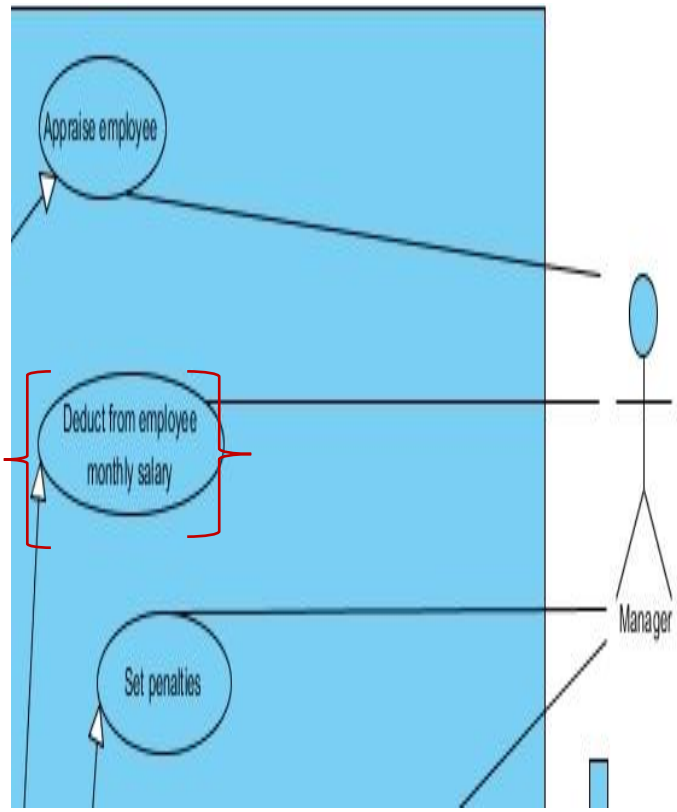


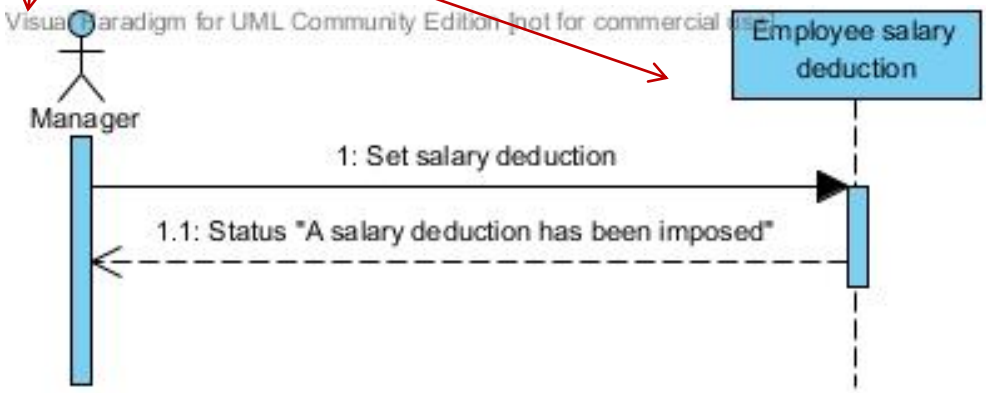
Each method between the manager and employee appraisal class represents an operation such as set employee appraisal to the destination object which is the employee appraisal class.

Class Diagram



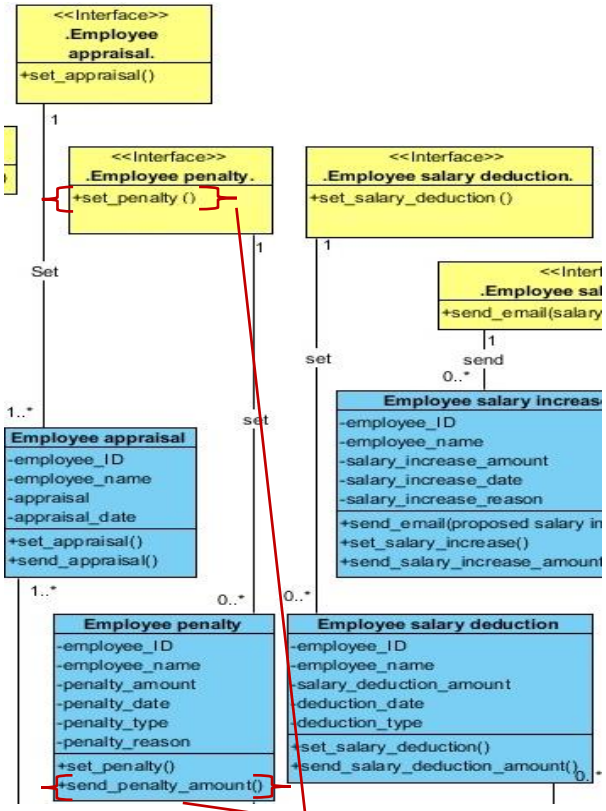
Use Case Diagram



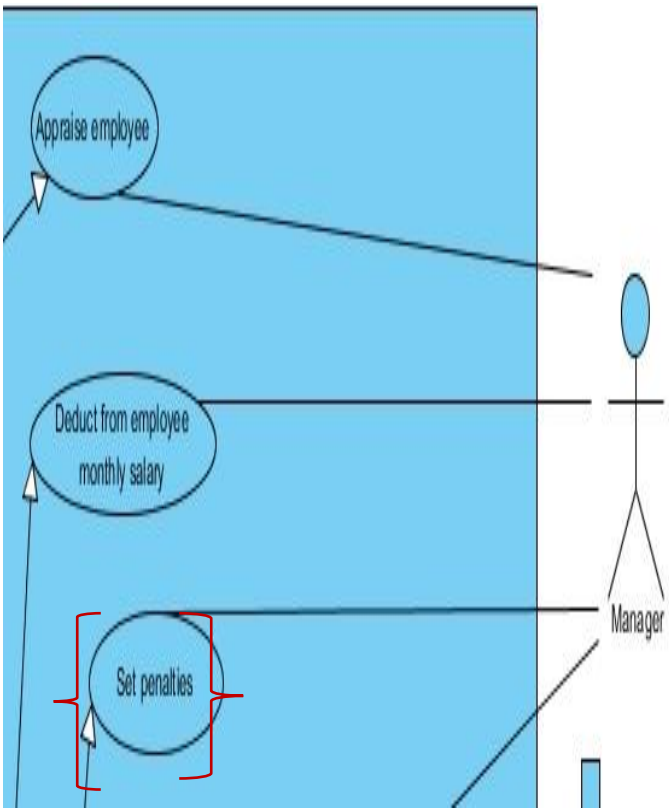


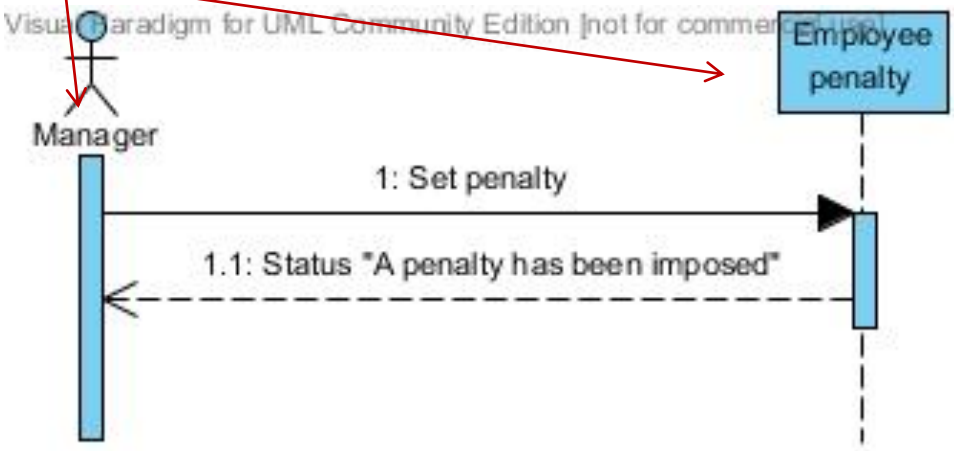
Each method between the manager and employee salary deduction class represents an operation such as set salary deduction to the destination object which is the employee salary deduction class.

Class Diagram

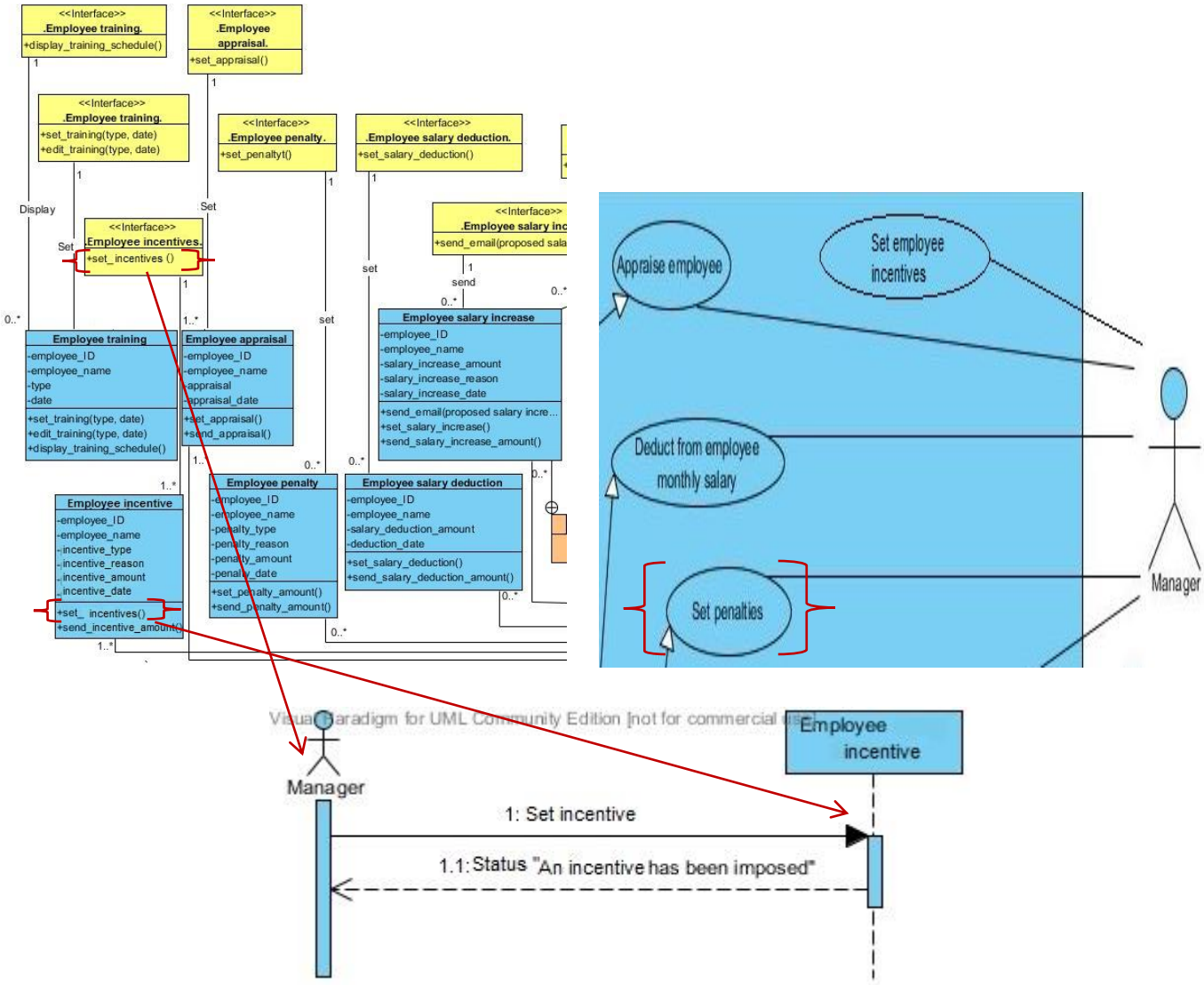


Use Case Diagram



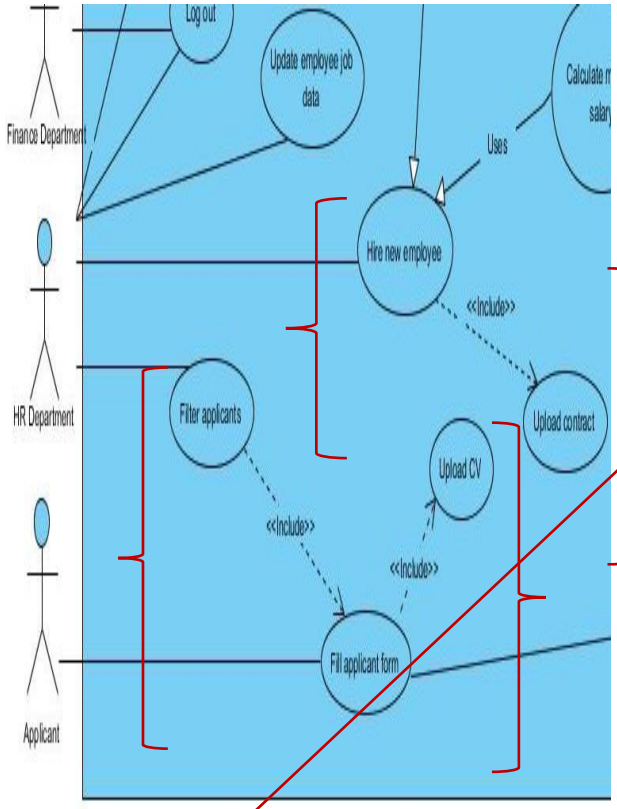


Each method between the manager and employee penalty class represents an operation such as set penalty to the destination object which is the employee penalty class.

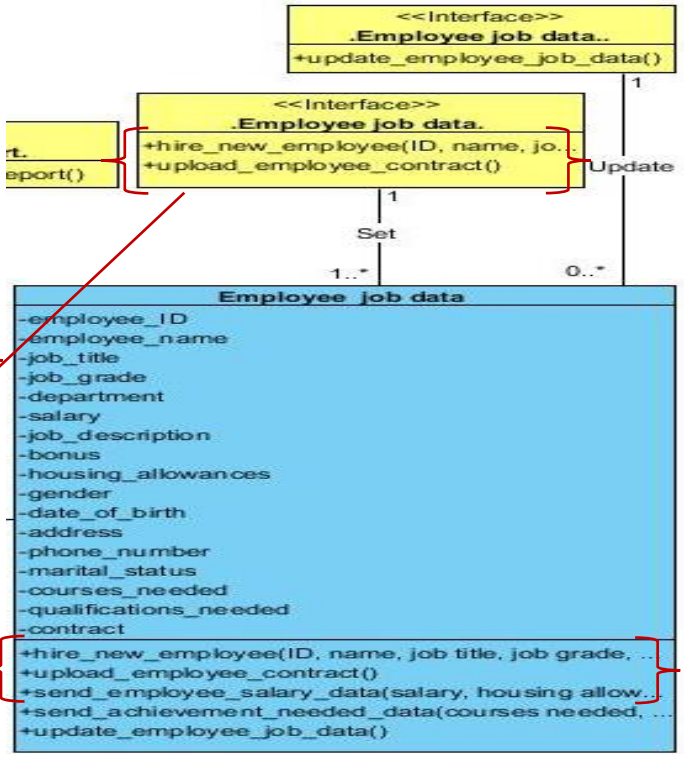


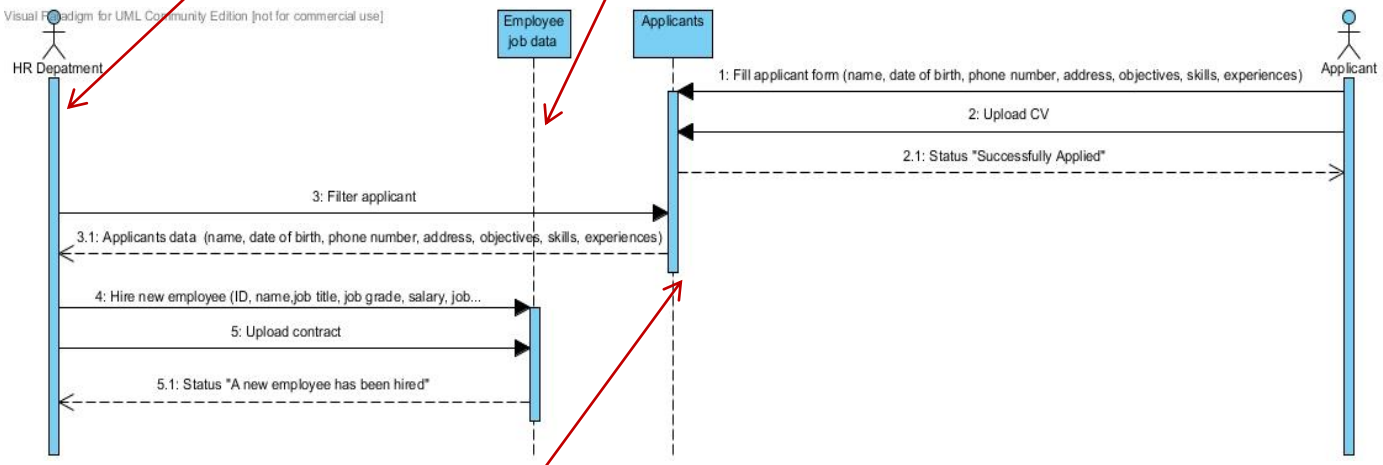
Each method between the manager and employee incentive class represents an operation such as set insentive to the destination object which is the employee incentive class.

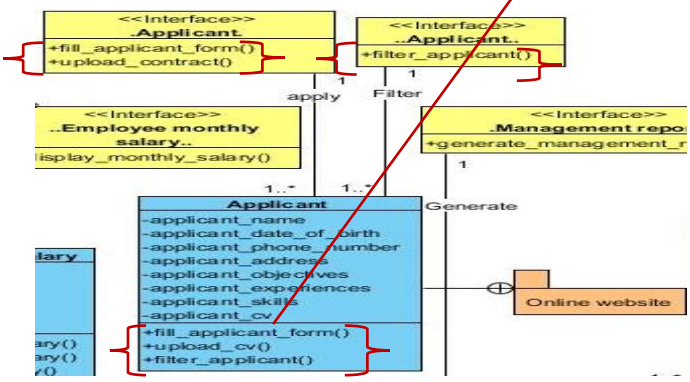
Use Case Diagram



Class Diagram

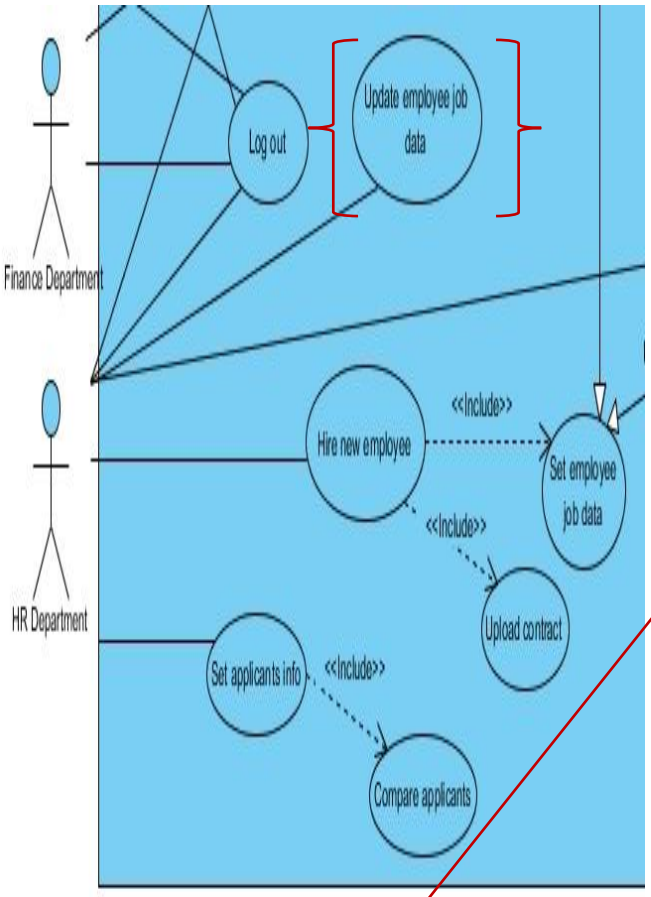




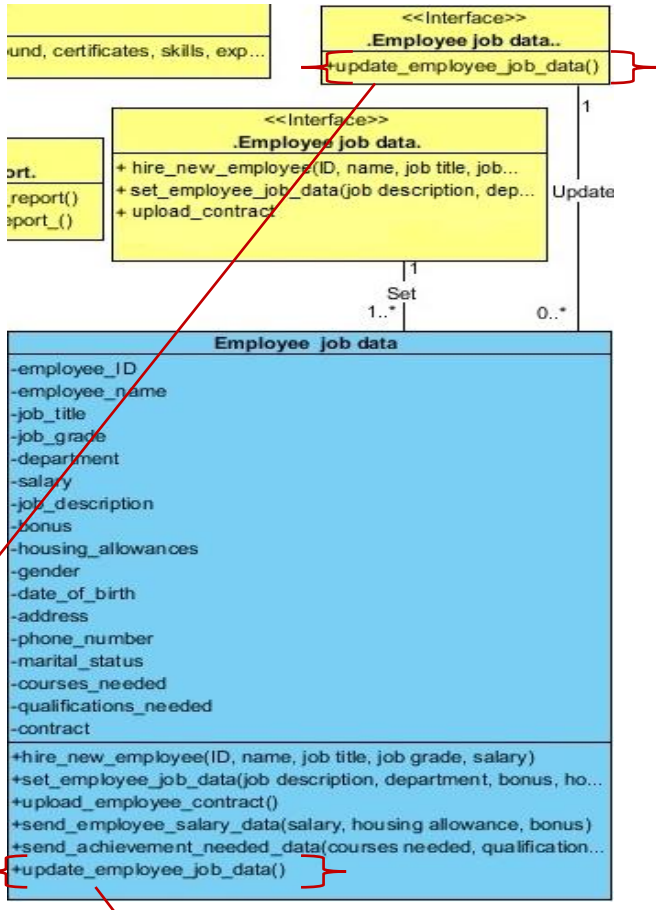


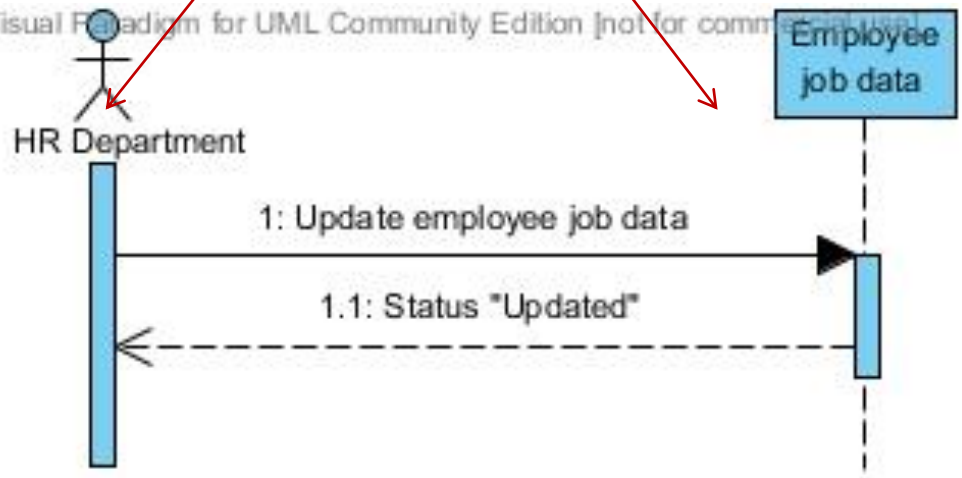
Each method between the Applicant or HR department and applicant class or employee job data class represents an operation such as fill applicants form, upload CV, filter applicants, hire new employee, & upload contract to the destination object which is the applicant class or employee job data class.

Use Case Diagram



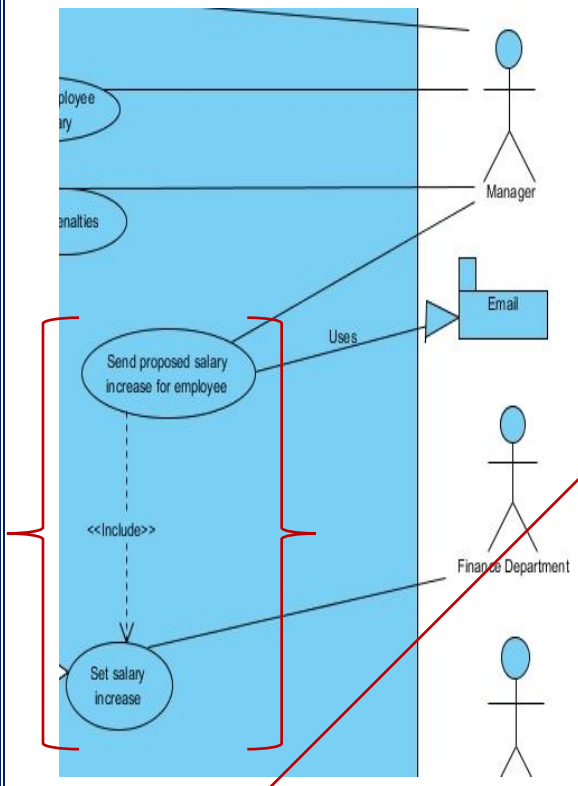
Class Diagram



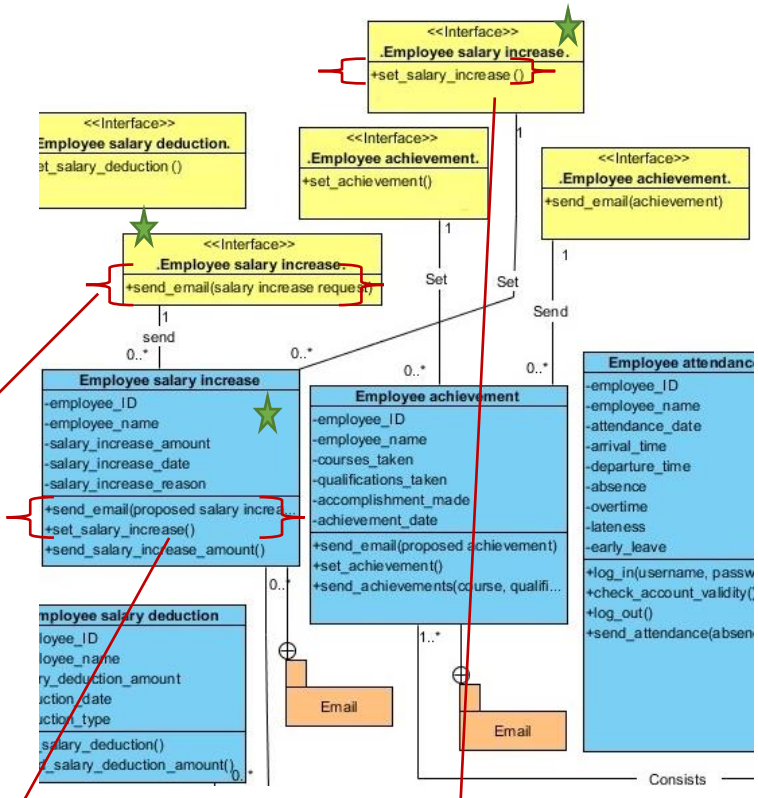


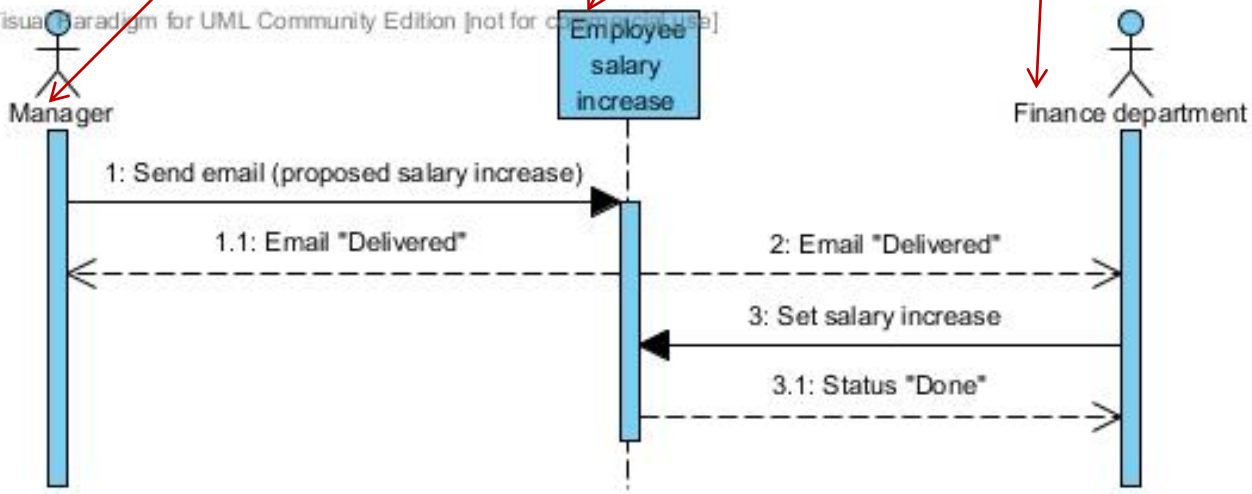
Each method between the HR department and employee job data class represents an operation such as update employee job data to the destination object which is the employee job data class.

Use Case Diagram



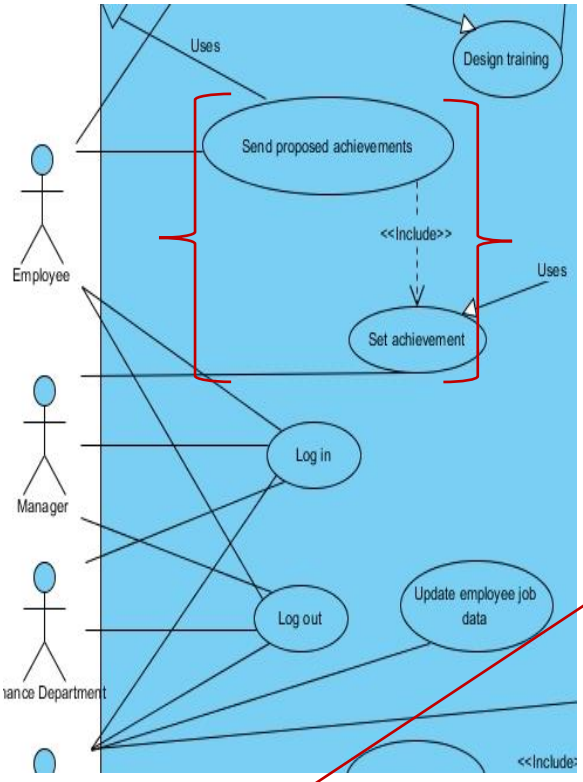
Class Diagram



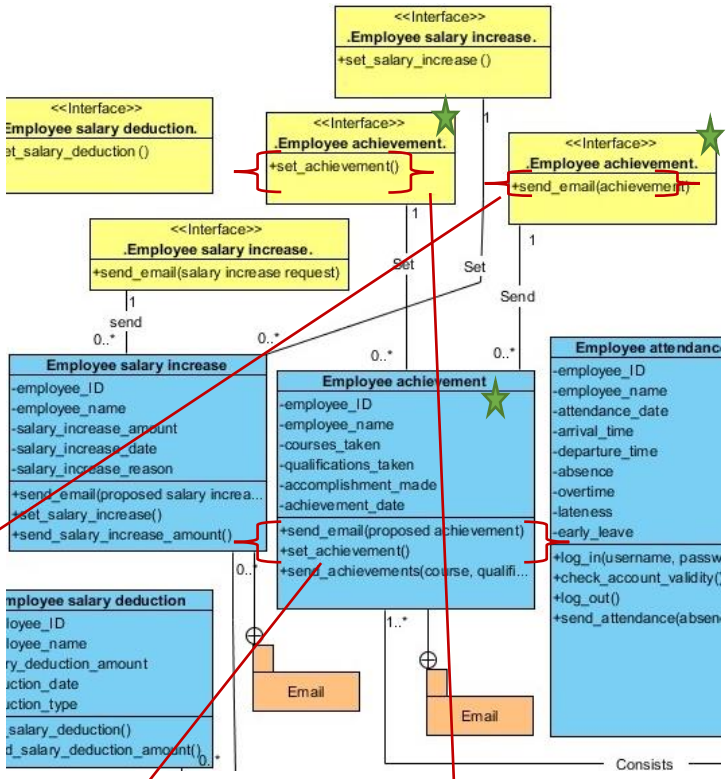


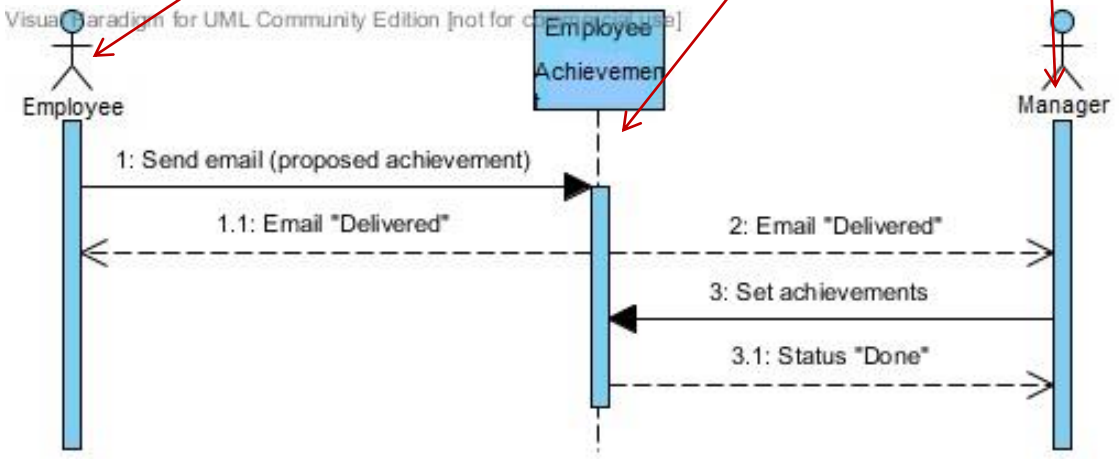
Each method between the Manager or Finance department and salary increase class represents an operation such as send email (proposed salary inrease) & set salary increase to the destination object which is the employee salary increase class.

Use Case Diagram

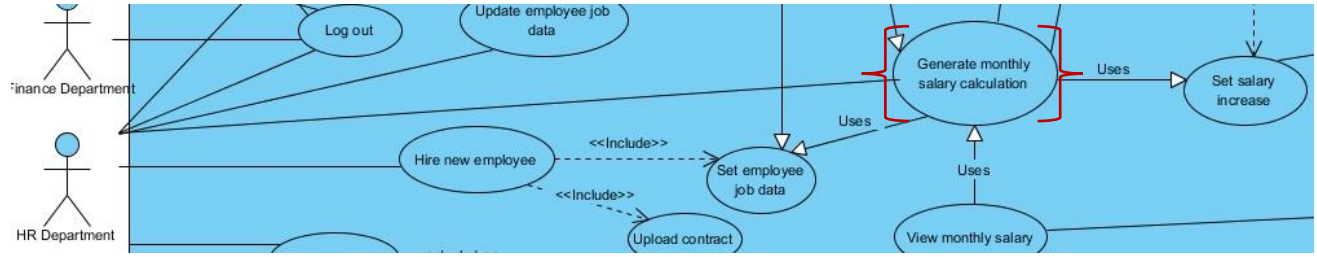


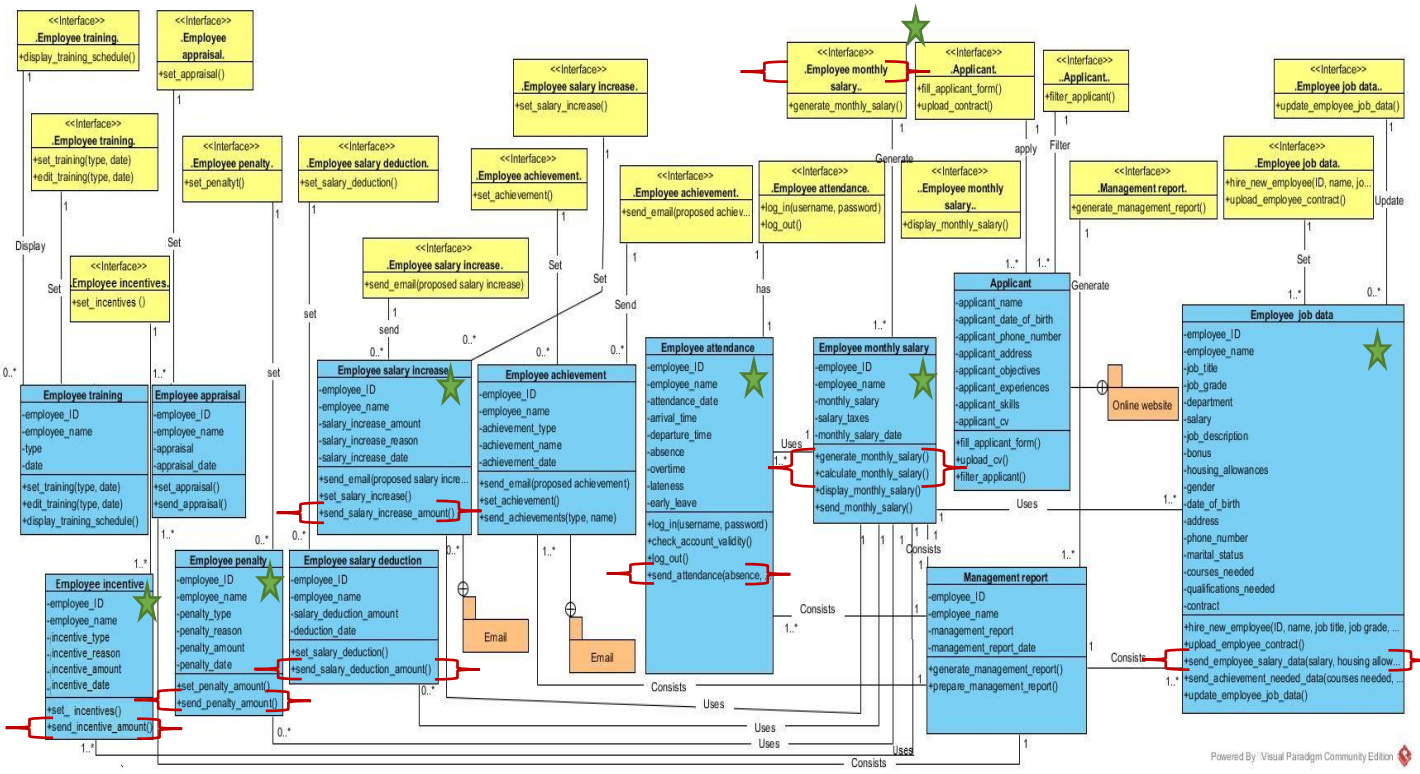
Class Diagram

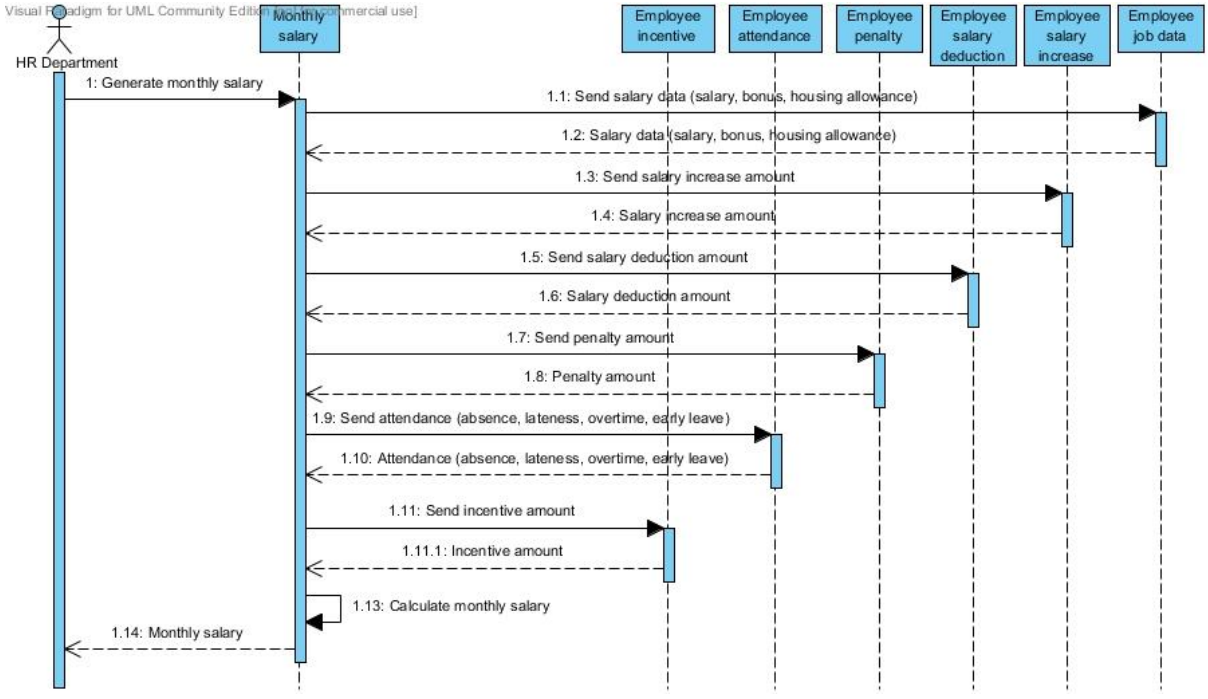




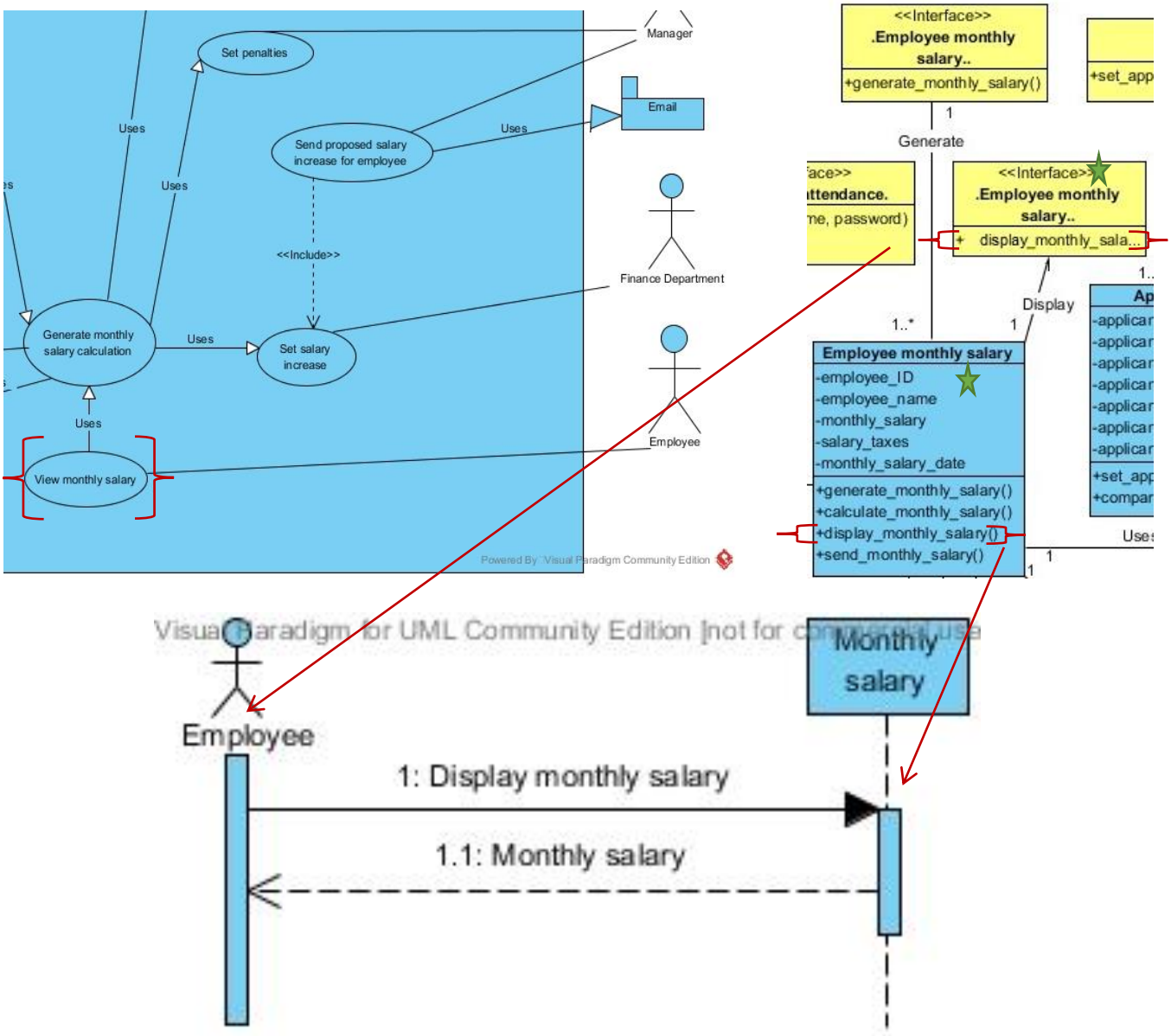
Each method between the Employee or Manager and employee achievement class represents an operation such as send email (proposed achievement) & set achievement to the destination object which is the employee achievement class.


Finance Department
HR Department
Log out
Update employee job data
Generate monthly salary calculation
Uses
Set salary increase
Uses
Hire new employee
<<Include>>
Set employee job data
<<Include>>
Upload contract
Uses
View monthly salary

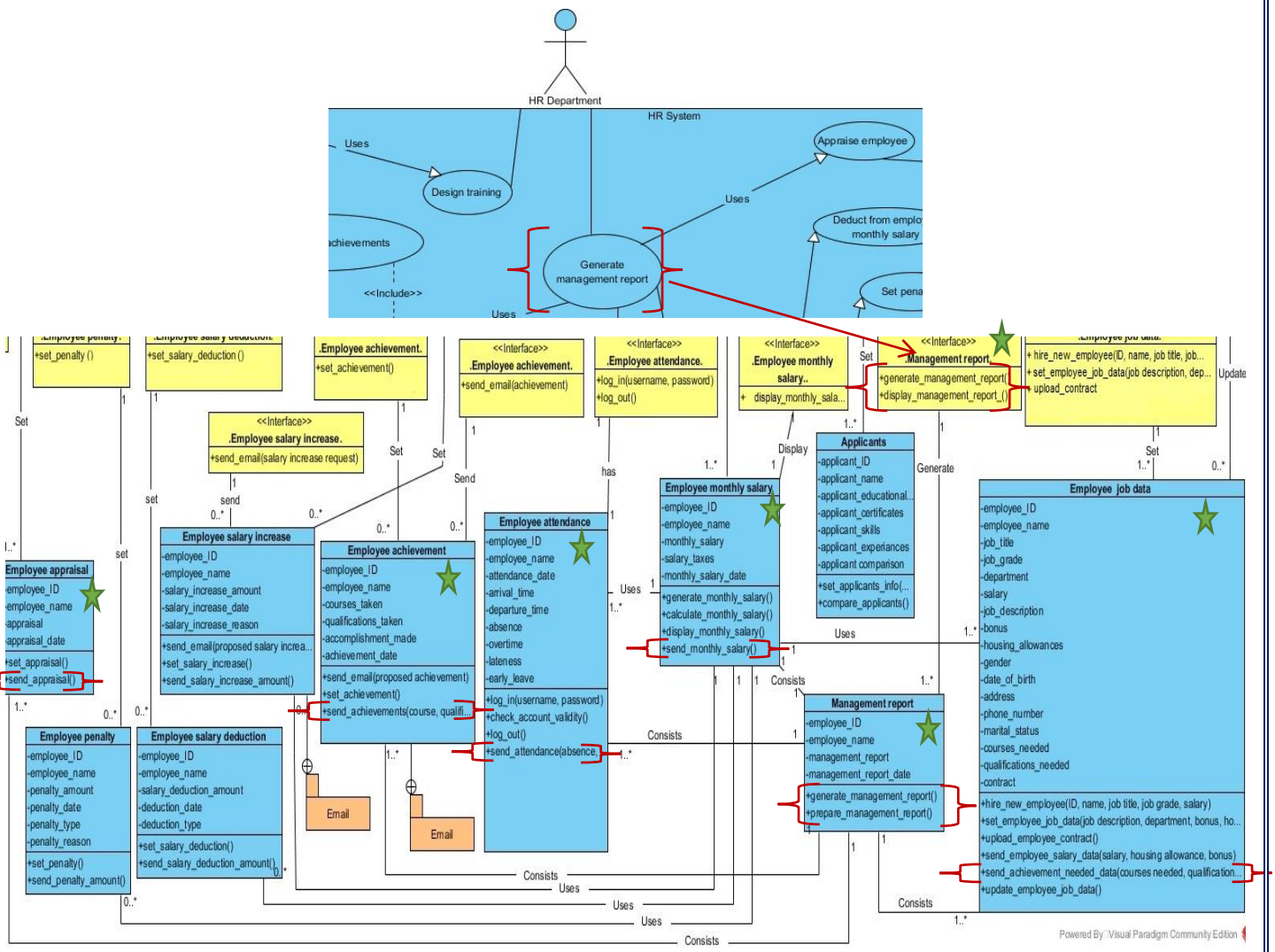

Employee training
Employee appraisal
Employee penalty
Employee salary deduction
Employee salary increase
Employee achievement
Employee attendance
Employee monthly salary
Applicant
Management report
Employee job data
Online website
Email
Email
Powered By Visual Paradigm Community Edition

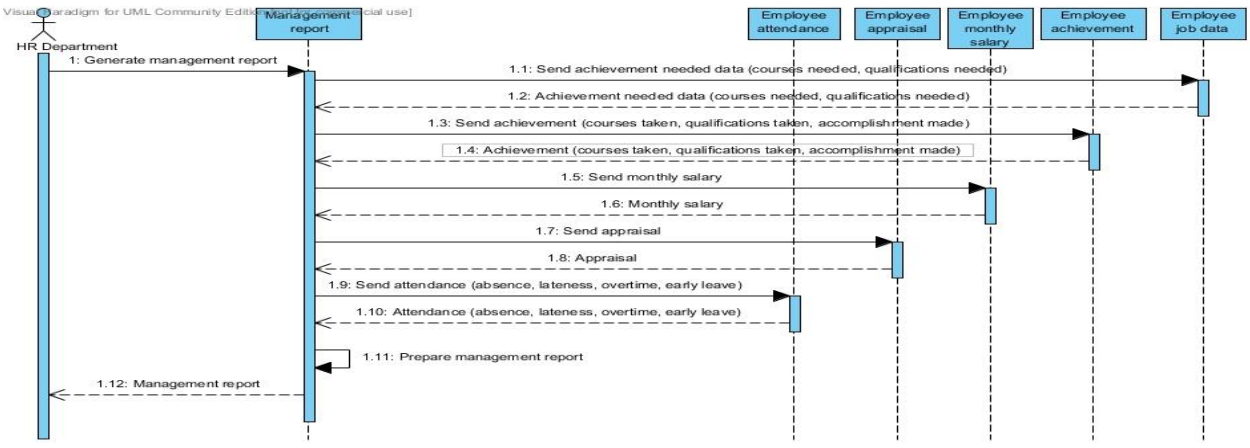

HR Department
Monthly salary
Employee incentive
Employee attendance
Employee penalty
Employee salary deduction
Employee salary increase
Employee job data
1: Generate monthly salary
1.1: Send salary data (salary, bonus, housing allowance)
1.2: Salary data (salary, bonus, housing allowance)
1.3: Send salary increase amount
1.4: Salary increase amount
1.5: Send salary deduction amount
1.6: Salary deduction amount
1.7: Send penalty amount
1.8: Penalty amount
1.9: Send attendance (absence, lateness, overtime, early leave)
1.10: Attendance (absence, lateness, overtime, early leave)
1.11: Send incentive amount
1.11.1: Incentive amount
1.13: Calculate monthly salary
1.14: Monthly salary

Each method between the HR department and monthly salary class or monthly salary class with other classes such as attendance, penalty, salary deduction, salary increase, job data classes represents an operation such as generate montly salary, send salary data, send penalties, send salary deductions, send salary increase, send attendance & calculate monthly salary to the destination object which is the employee monthly salary, job data, penalty, salary deduction, salary increase & attendance classes.



Each method between the Employee and monthly salary class represents an operation such as display monthly salary to the destination object which is the monthly salary class.

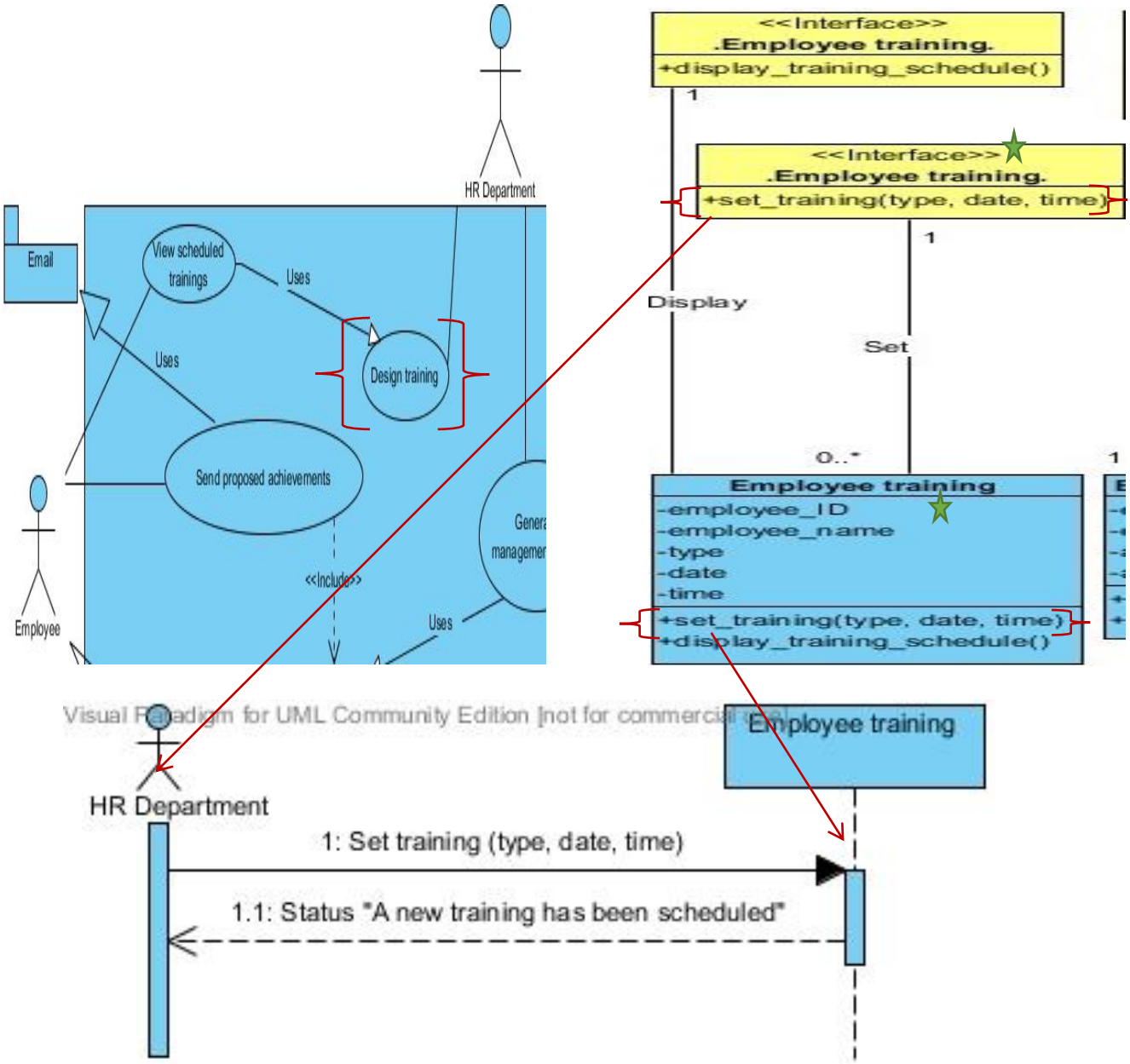




Each method between the HR department and management report class or management report class with other classes such as attendance, appraisal, monthly salary, achievement, job data classes represents an operation such as generate management report, send achievements, send appraisal, send monthly salary, send job data, send achivement needed

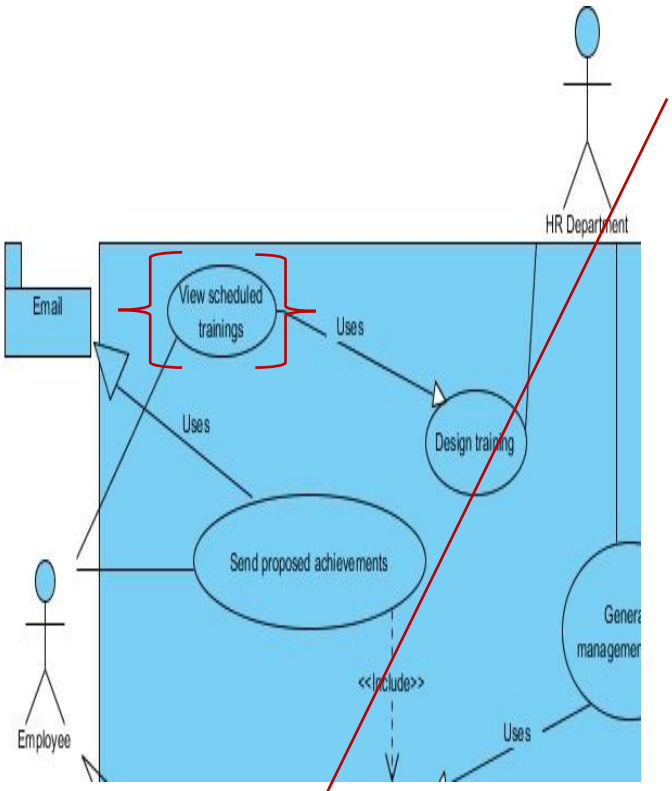
data & prepare management report to the destination object which is the employee management report, attendance, appraisal, monthly salary, achievement & job data classes.

Use Case Diagram **Class Diagram**

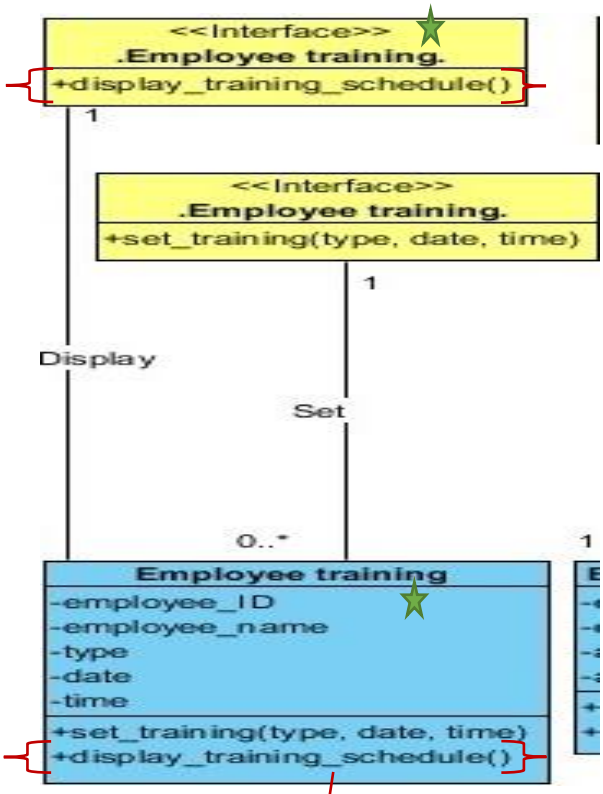


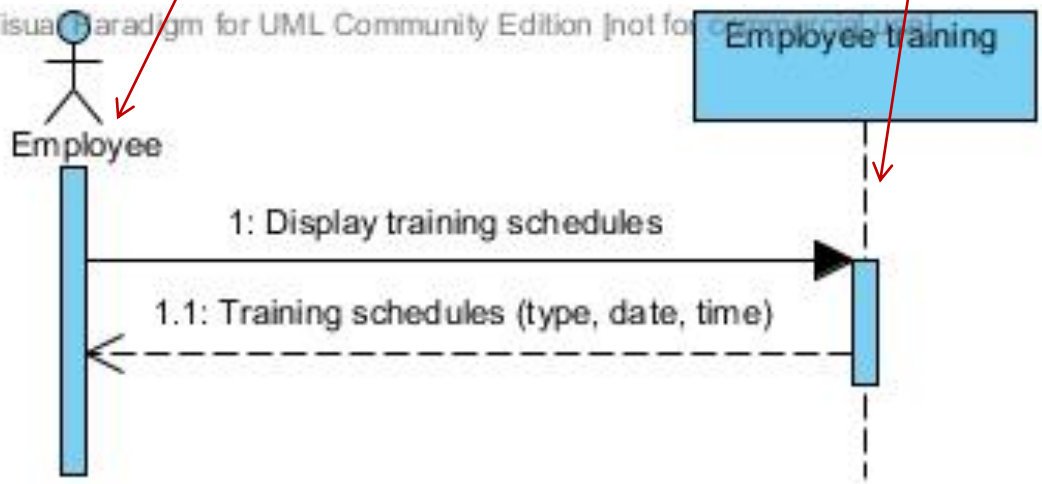
Each method between the HR department and employee training class represents an operation such as set training to the destination object which is the employee training class.

Use Case Diagram



Class Diagram

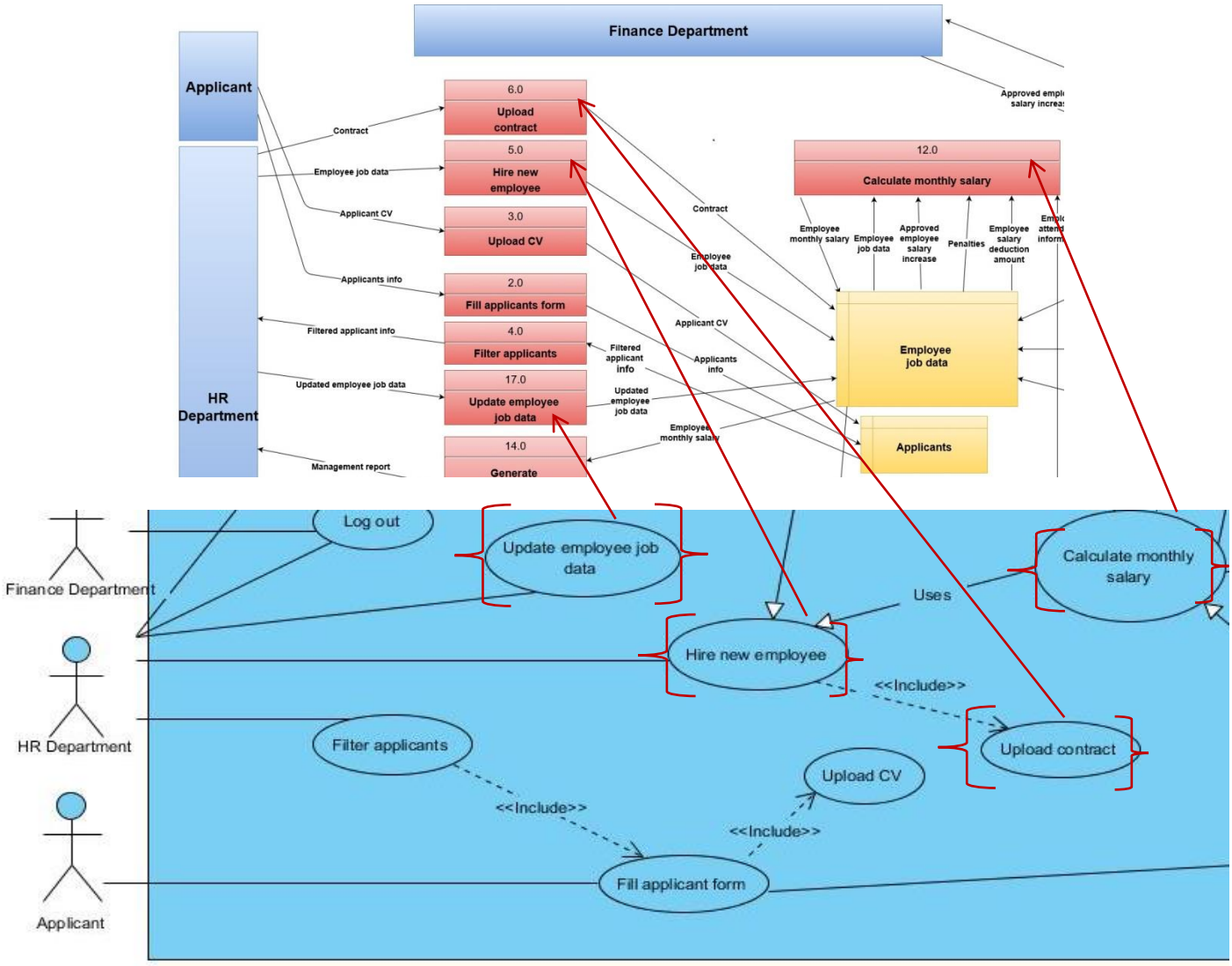




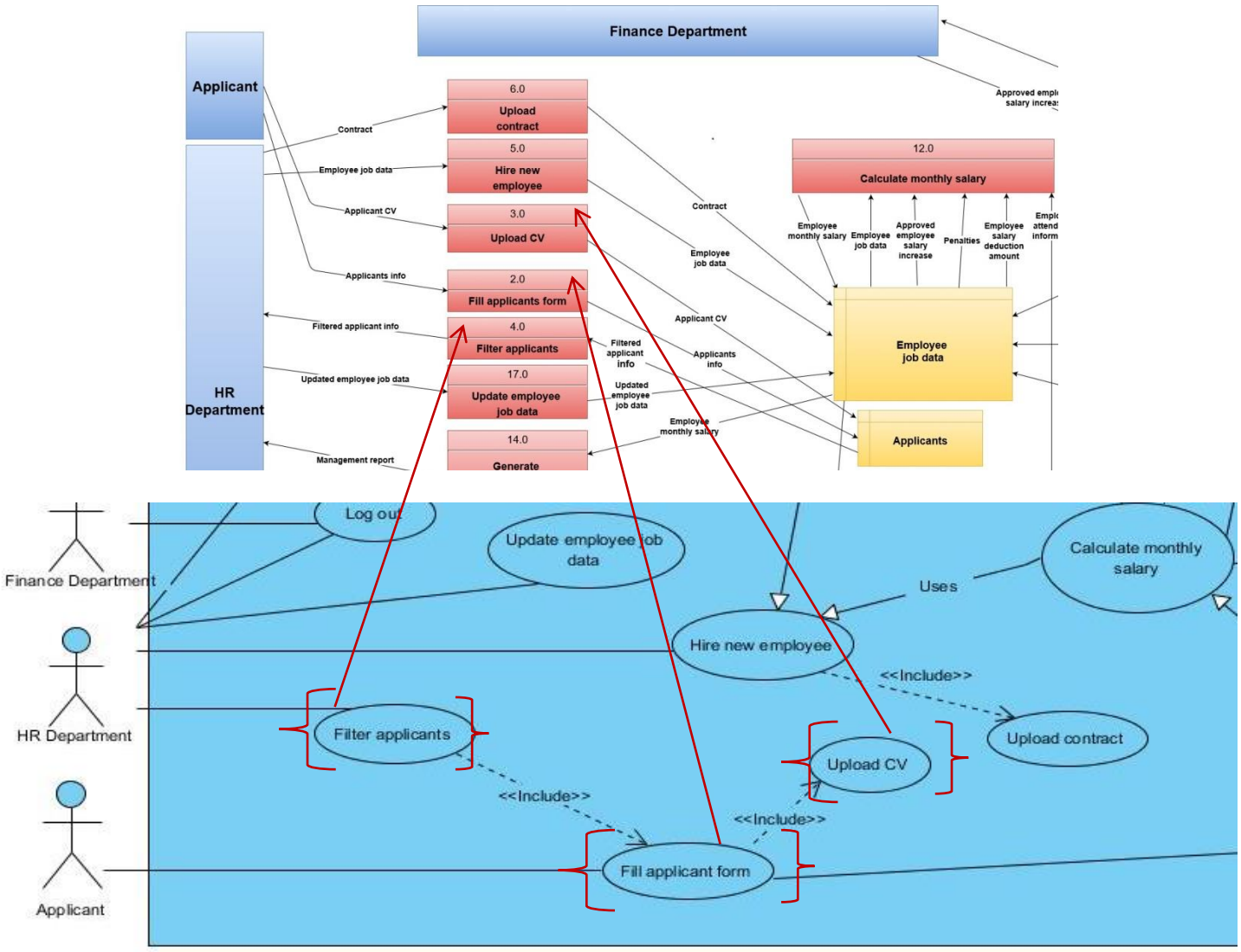
Each method between the Employee and employee training class represents an operation such as display training schedules to the destination object which is the employee training class.

USE CASE DIAGRAM
VS.
DFD LEVEL-0

On the other hand, DFD level-0 represent the processes required to accomplish those requirement which means each process in the DFD level-0 should response to a use case.

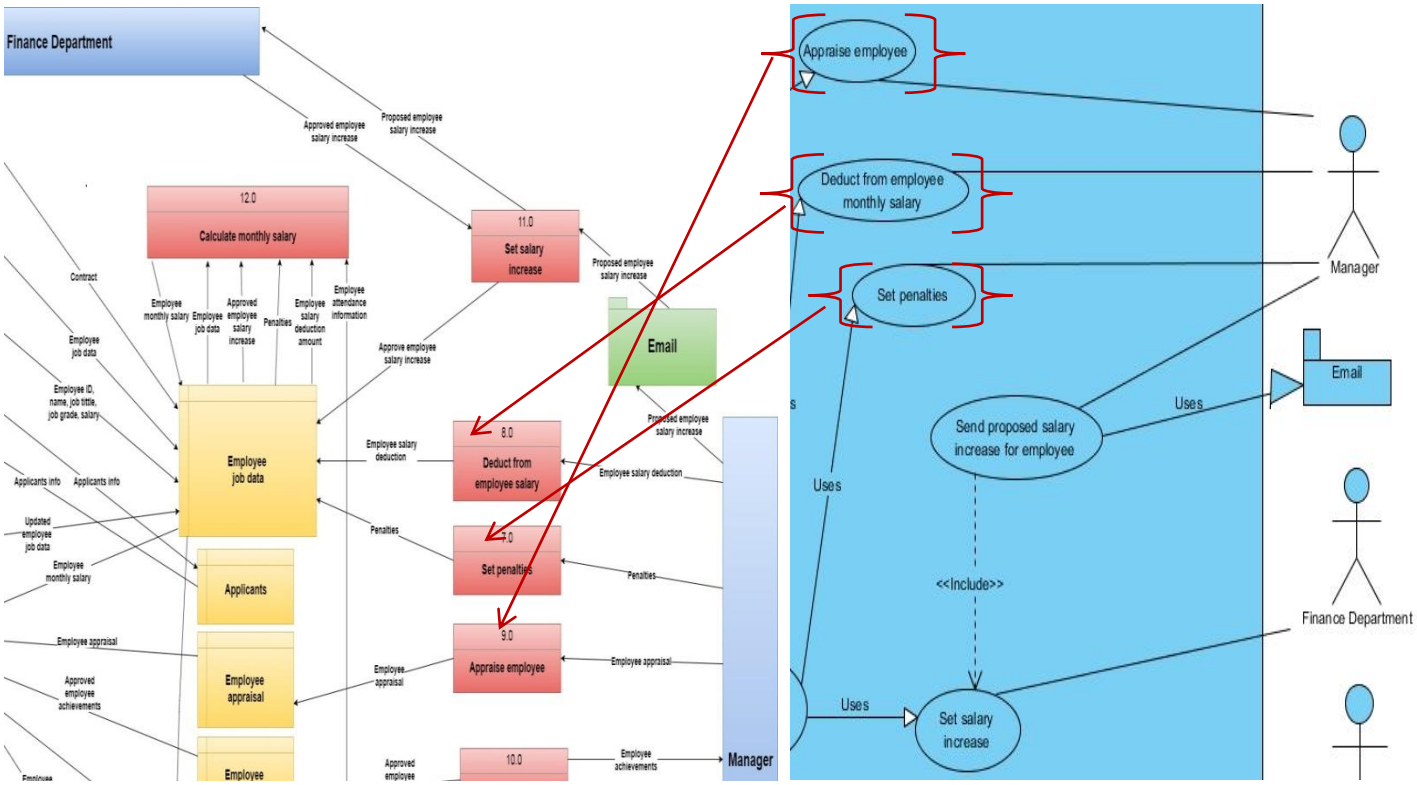


DFD level-0 represent the processes required to accomplish those requirement which means each process in the DFD level-0 above such as calculate monthly salary, hire new employee, upload contract & update employee job data should response to a use case such as generate monthly salary calculation, update employee job data, hire new employee & upload contract shown in the above use case diagram.



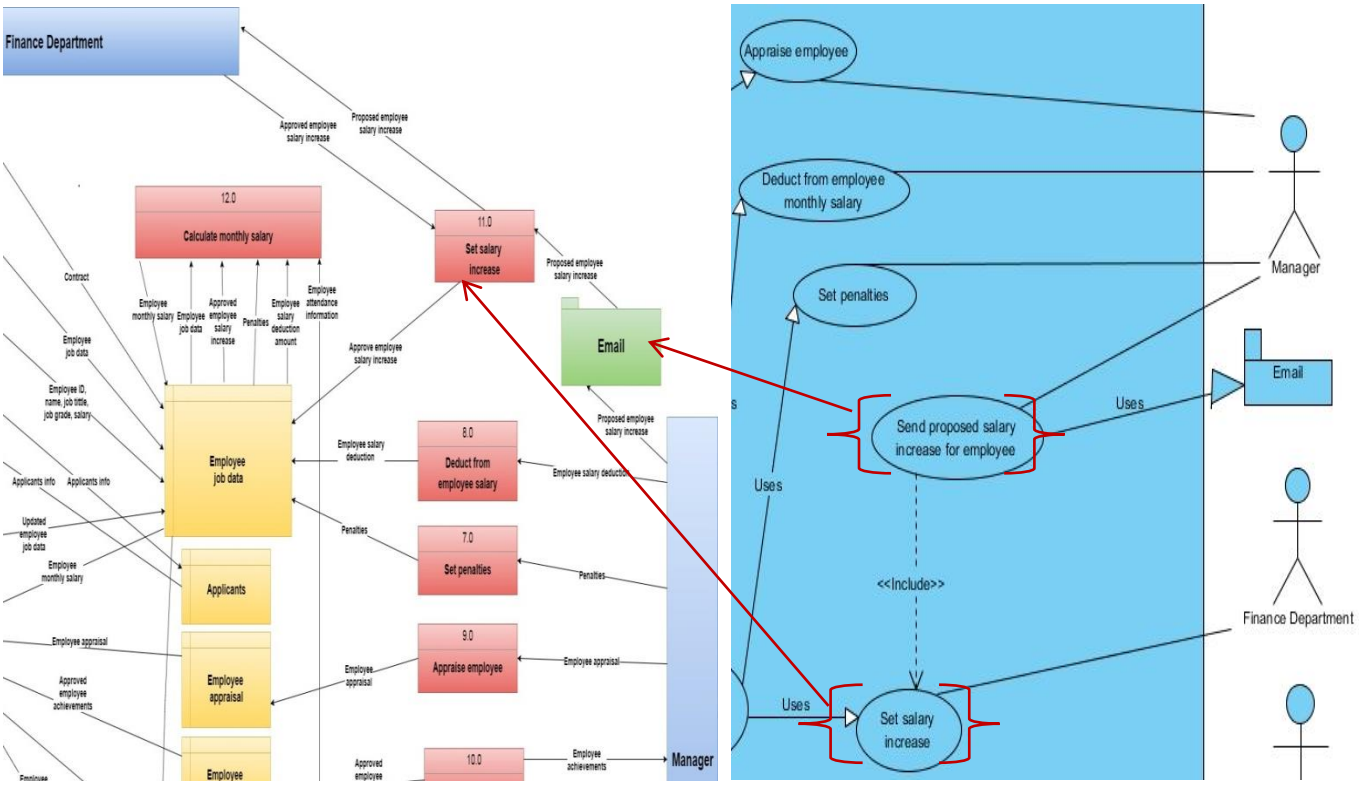
DFD level-0 represent the processes required to accomplish those requirement which means each process in the DFD level-0 above such as fill applicant form, upload CV & filter applicants should response to a use case such as fill applicant form, upload CV & filter applicants shown in the above use case diagram.

DFD Level-0 **Use Case**

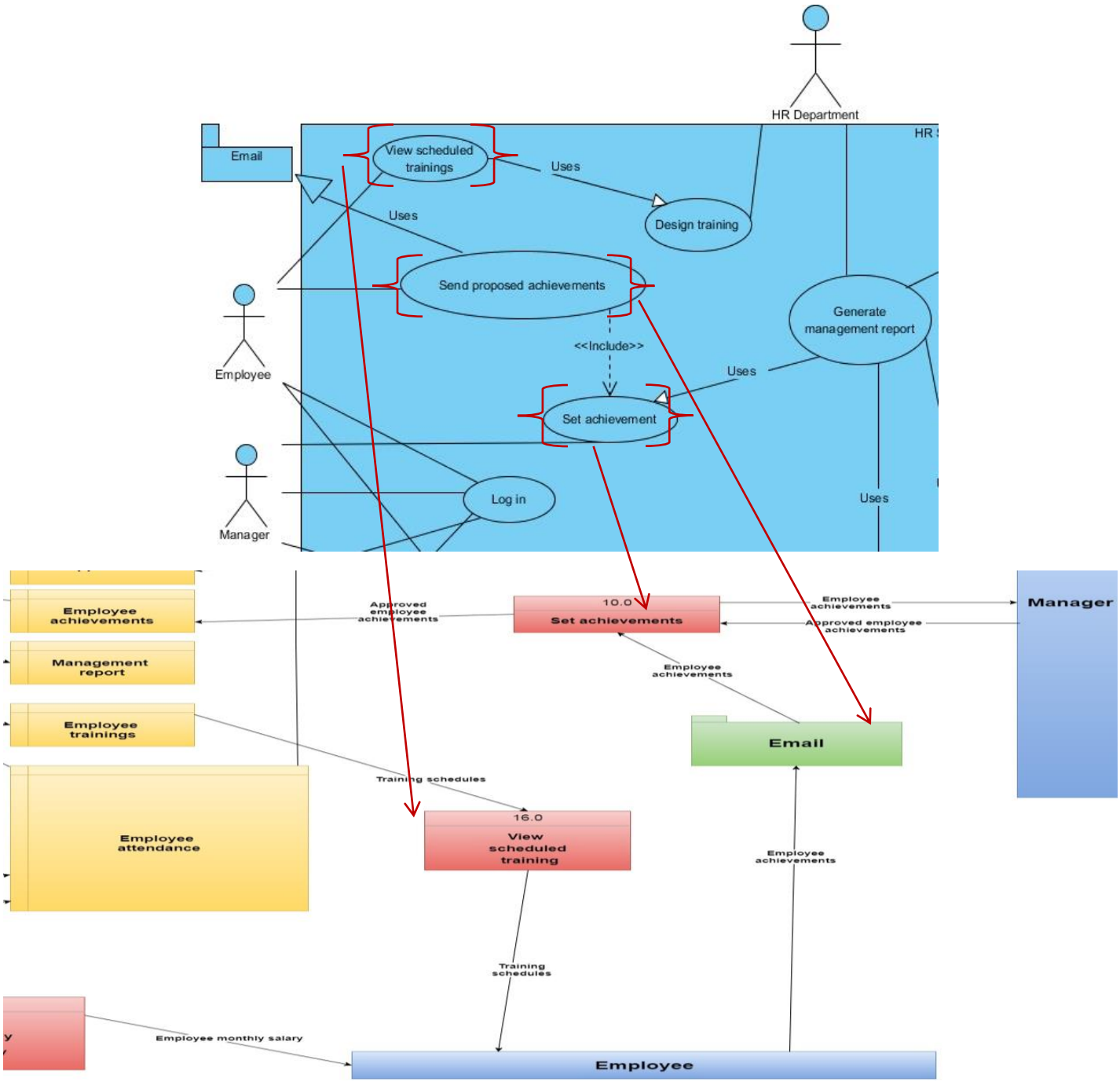


DFD level-0 represent the processes required to accomplish those requirement which means each process in the DFD level-0 above such as appraise employee, deduct from employee salary & set penalties should response to a use case such as appraise employee, deduct from employee salary & set penalties shown in the above use case diagram.

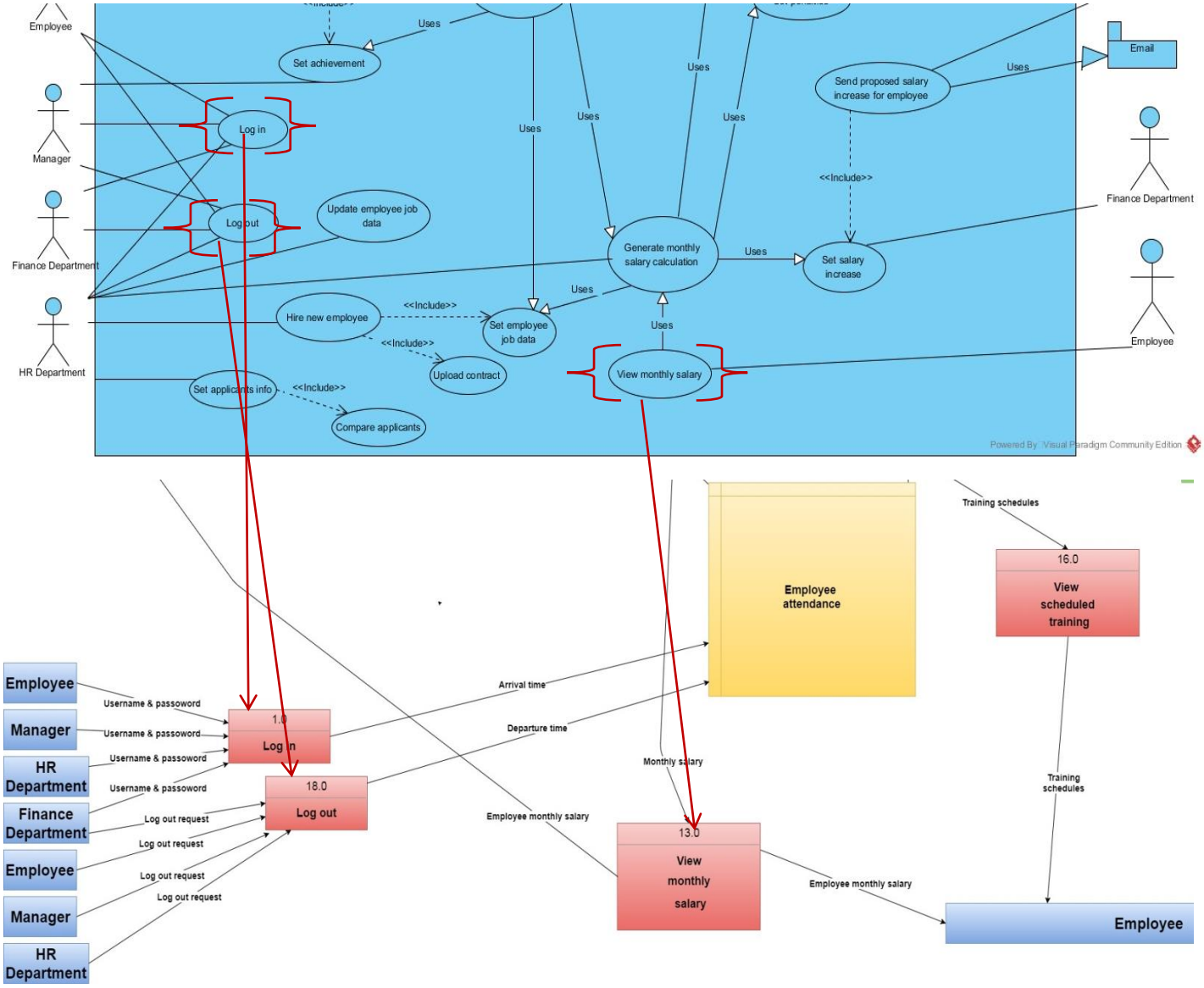
DFD Level-0 **Use Case**



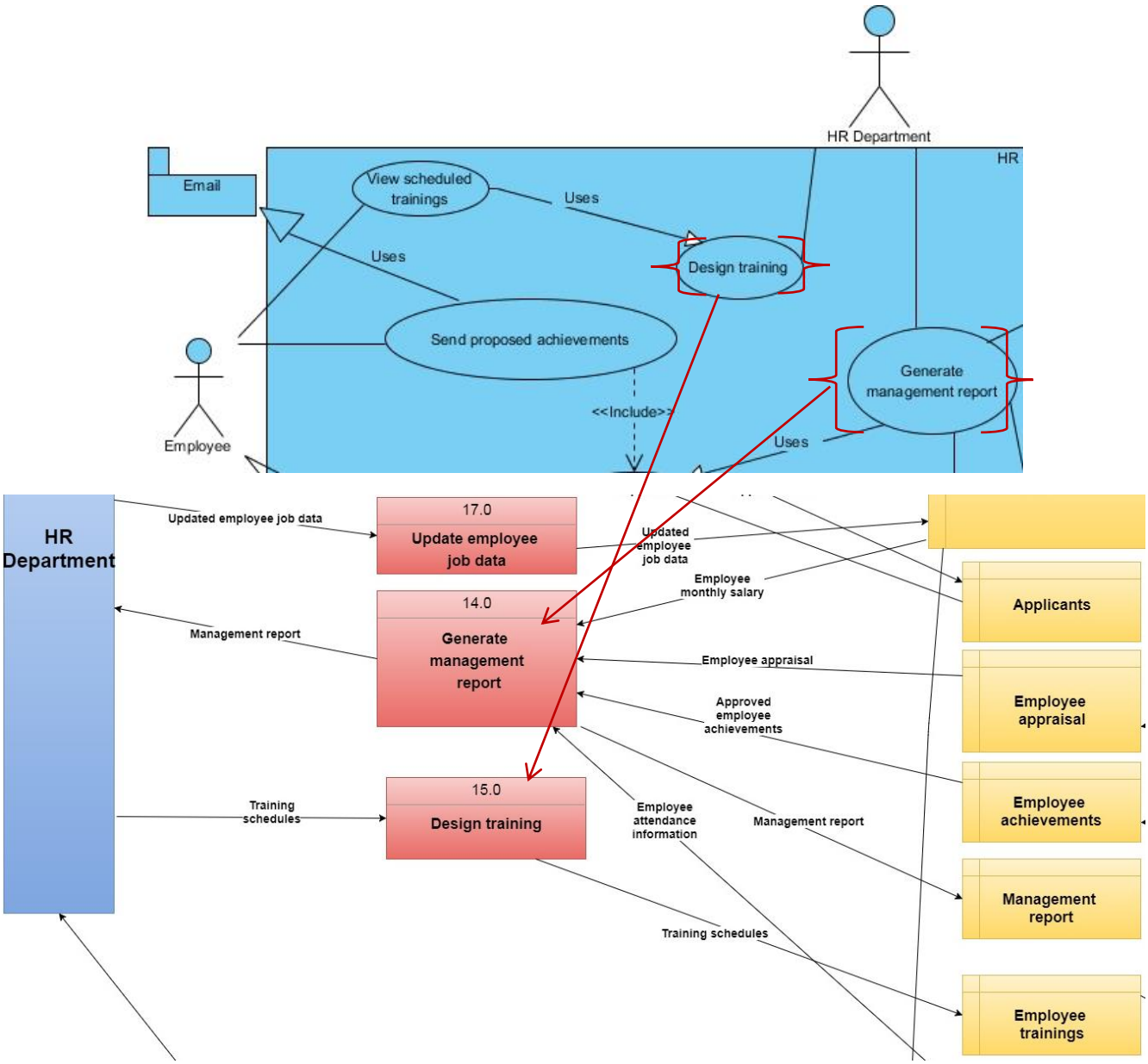
DFD level-0 represent the processes required to accomplish those requirement which means each process in the DFD level-0 above such as email package & set salary increase should response to a use case such as send proposed salary increase for employee & set salary increase shown in the above use case diagram.



DFD level-0 represent the processes required to accomplish those requirement which means each process in the DFD level-0 above such as view scheduled training, email package, & set salary increase should response to a use case such as view scheduled training, send proposed achievements & set achievement shown in the above use case diagram.



DFD level-0 represent the processes required to accomplish those requirement which means each process in the DFD level-0 above such as view monthly salary, log in & log out should response to a use case such as view monthly salary, log in & log out shown in the above use case diagram.

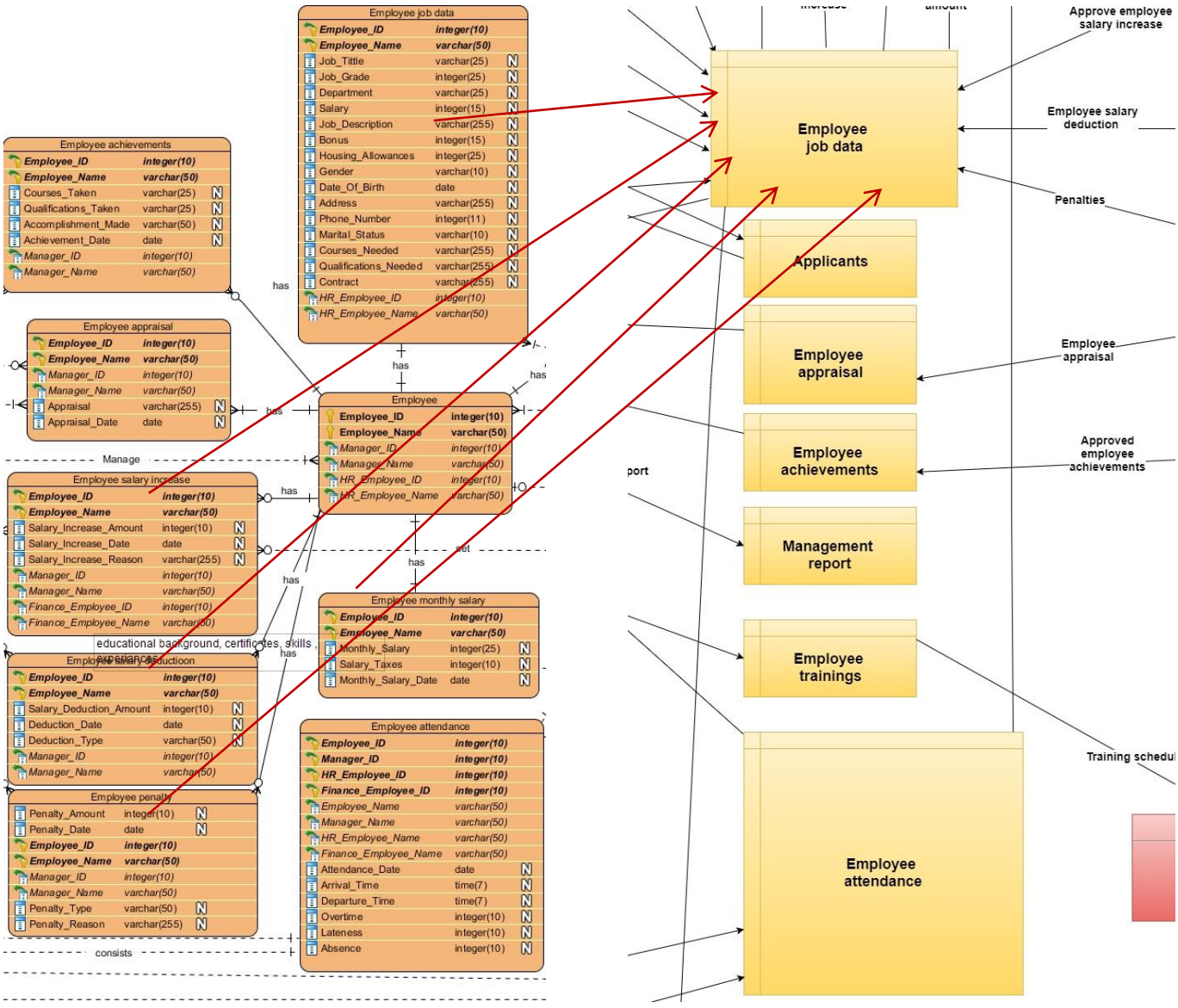


DFD level-0 represent the processes required to accomplish those requirement which means each process in the DFD level-0 above such as generate management report & design training should response to a use case such as generate management report & design training shown in the above use case diagram.

ENTITY RELATIONSHIP DIAGRAM VS. DATA FLOW DIAGRAM LEVEL-0

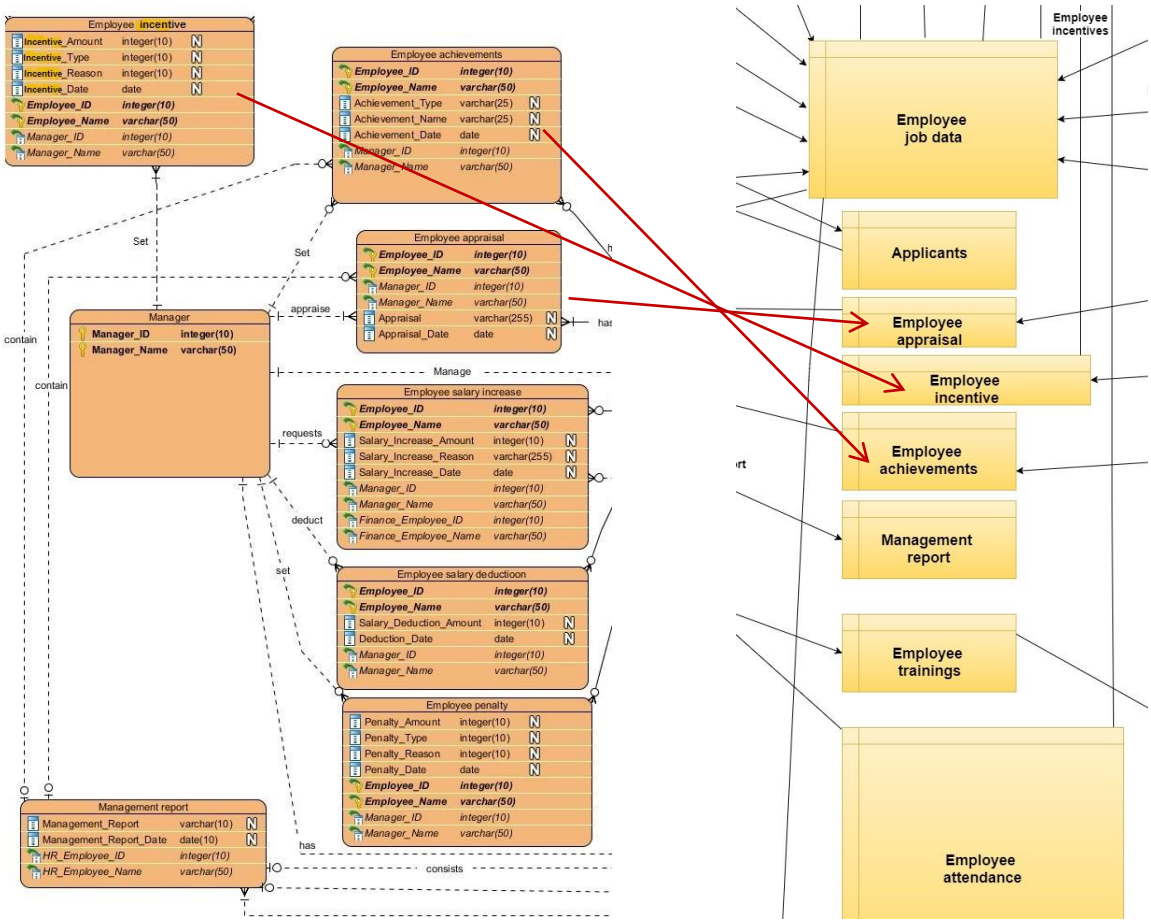
The data stores in the data flow diagram level-0 are used to create the entities of entity relationship diagram

ERD **DFD**



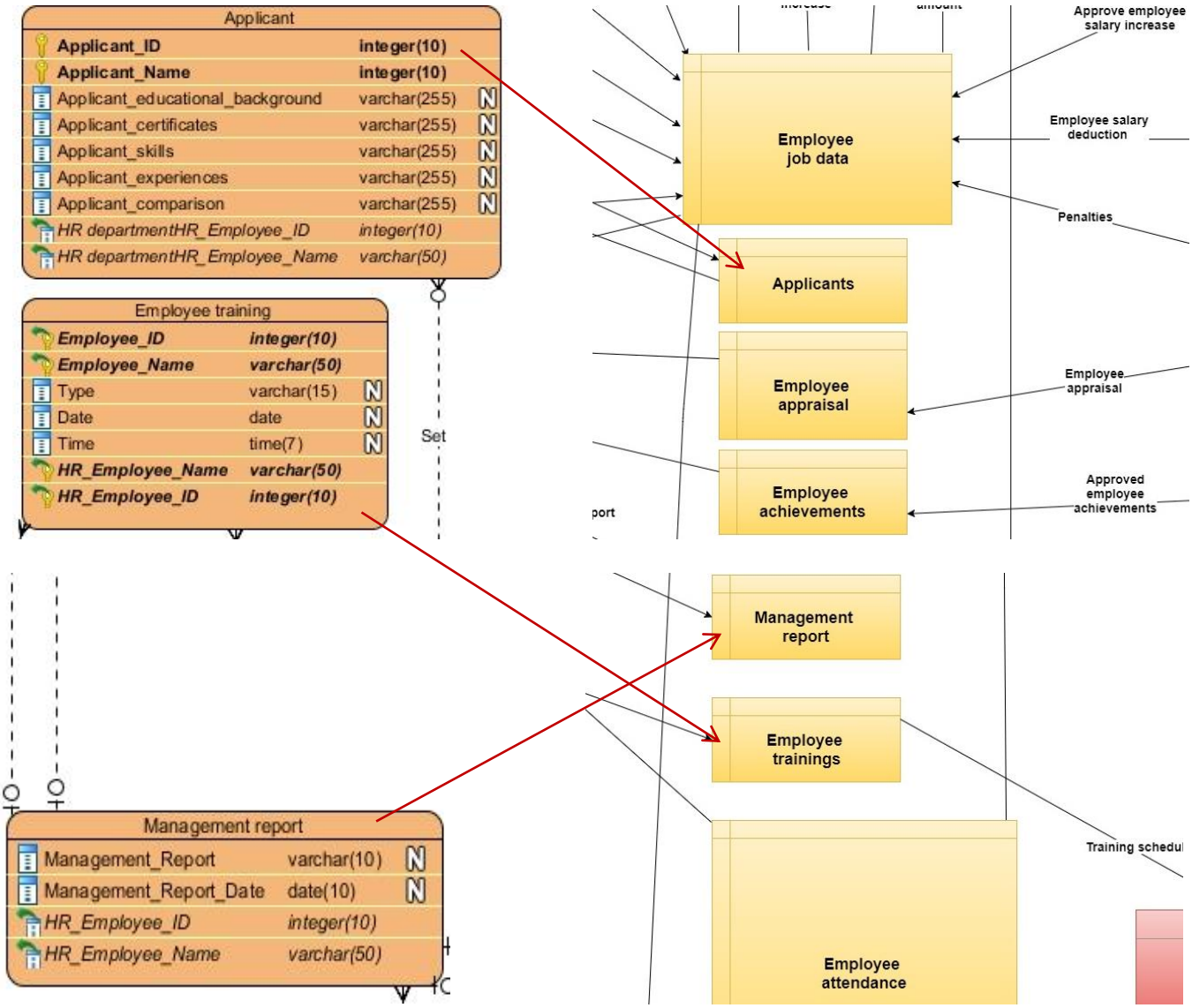
The data stores in the DFD level-0 are used to create the entities of ERD in which Employee job data store in DFD level-0 is used to create five entities which are employee job data, employee monthly salary, employee salary increase, employee penalty, and employee salary deduction

ERD **DFD**



The data stores in the DFD level-0 are used to create the entities of ERD in which Employee achievement data store in DFD level-0 is used to an entity which is employee achievement entity, Employee appraisal data store in DFD level-0 is used to an entity which is employee appraisal entity, and Employee incentive data store in DFD level-0 is used to an entity which is employee incentive entity.

ERD **DFD**



The data stores in the DFD level-0 are used to create the entities of ERD in which Applicant data store in DFD level-0 is used to an entity which is applicant entity, Employee training data store in DFD level-0 is used to an entity which is employee training entity, and Management report data store in DFD level-0 is used to an entity which is management report entity.

15.0 Verification test of the sequence diagram

			Employee attendance class								
Use Case	Operation	Check with activity	ID	Name	Date	Arrival time	Departure time	Absence	Overtime	Lateness	Early leave
Log in	Log in	✓	44	Omar Mamdouh							
	Check account validity	✓			3/4/2017	8:00:00 AM		0		1 hour	

			Employee attendance class								
Use Case	Operation	Check with activity	ID	Name	Date	Arrival time	Departure time	Absence	Overtime	Lateness	Early leave
Log out	Log out	✓	44	Omar Mamdouh	3/4/2017		4:00:00 PM		2 hours		0.00

			Applicants class							
Use Case	Operation	Check with activity	Name	Date of birth	Phone number	Address	Objectives	Experiances	Skills	CV
Fill applicants form	Fill applicant form	✓	Islam Heikal	3/4/1994	01014403332	6 of october, Giza	Fullfiling the company's role needs	System Analyst	Presentation, computer, communication skills	

			Applicants class							
Use Case	Operation	Check with activity	Name	Date of birth	Phone number	Address	Objectives	Experiances	Skills	CV
Upload CV	Upload CV	✓								

			Employee job data class							
Use Case	Operation	Check with activity	ID	Name	Job tittle	Job grade	Department	Salary	Job description	Bonus
Hire new employee	Hire new employee	✓	186	Ahmad Amgad	Finance Manager	Grade-6	Finance Department	EGP 15,000.00	Finance manager responsible for all of Africa	EGP 100.00

			Employee job data class							
Use Case	Operation	Check with activity	ID	Name	Job tittle	Job grade	Department	Salary	Job description	Bonus
Upload Contract	Upload contract	✓								

Housing allowance	Gender	Date of birth	Address	Phone number	Marital status	Courses needed	Qualifications needed	Contract
EGP 1,000.00	Male	3/4/1994	6 of october, Giza	01014403332	Finished	ICDL	CMA	

Housing allowance	Gender	Date of birth	Address	Phone number	Marital status	Courses needed	Qualifications needed	Contract

First, the log in use case which is implemented by the log in operation and is applied in the log in activity diagram in which it use some attributes as shown above.

Second, the log out use case which is implemented by the log out operation and is applied in the log out activity diagram in which it use some attributes as shown above.

Third, the fill applicant form use case which is implemented by the fill applicant form operation and is applied in the recruitment activity diagram in which it use some attributes as shown above.

Forth, the upload CV use case which is implemented by the upload CV operation and is applied in the recruitment activity diagram in which it use some attributes as shown above.

Fifth, the hire new employee use case which is implemented by the hire new employee operation and is applied in the recruitment activity diagram in which it use some attributes as shown above.

Sixthth, the upload contract use case which is implemented by the upload contract operation and is applied in the recruitment activity diagram in which it use some attributes as shown above.

			Appraisal class					
Use Case	Operation	Check with activity	ID	Name	Appraisal	Date		
Appraisal employee	Set employee appraisal	✓	164	Omar Adel		8/7/2017		
			Salary deduction class					
Use Case	Operation	Check with activity	ID	Name	Deduction type	Deduction Amount	Date	
Deduct from employee salary	Set salary deduction	✓	142	Youssef Mourshed	Medical Insurance	EGP 500.00	6/7/2017	
			Penalty class					
Use Case	Operation	Check with activity	ID	Name	Penalty type	Penalty reason	Penalty amount	Date
Set penalties	Set penalty	✓	94	Omar Essam	Automatic Penalty type	Incompliance to manager order	EGP 300.00	8/8/2017
			Salary increase class					
Use Case	Operation	Check with activity	ID	Name	Amount	Reason for salary increase	Date	
Send proposed salary increase for employee	Send email (proposed salary increase)	✓	143	Youssef Mourshed	EGP 1,000.00	Due to achieving the previous six months sales targets	6/7/2017	
			Salary increase class					
Use Case	Operation	Check with activity	ID	Name	Amount	Reason for salary increase	Date	
Set salary increase	Set salary increase	✓	143	Youssef Mourshed	EGP 1,000.00	Due to achieving the previous six months sales targets	14/7/2017	
			Achievement class					
Use Case	Operation	Check with activity	ID	Name	Courses taken	Qualifications taken	Achievement made	Date
Send proposed achievement	Send email (proposed achievement)	✓	89	Omar Essam		SAP Accociate Consultant		8/8/2017
			Achievement class					
Use Case	Operation	Check with activity	ID	Name	Courses taken	Qualifications taken	Achievement made	Date
Set achievement	Set salary increase	✓	89	Omar Essam		SAP Accociate Consultant		18/8/2017

Seventh, the appraise employee use case which is implemented by the appraise employee operation and is applied in the appraisal activity diagram in which it use some attributes as shown above.

Eighth, the deduct from employee salary use case which is implemented by the deduct from employee salary operation and is applied in the salary deductions activity diagram in which it use some attributes as shown above.

Ninth, the set penalty use case which is implemented by the set penalty operation and is applied in the penalty activity diagram in which it use some attributes as shown above.

Tenth, the send proposed salary increase use case which is implemented by the send proposed salary increase operation and is applied in the salary increase activity diagram in which it use some attributes as shown above.

Eleventh, the set salary increase use case which is implemented by the set salary increase operation and is applied in the salary increase activity diagram in which it use some attributes as shown above.

Twelve, the send proposed achievement use case which is implemented by the send proposed achievement operation and is applied in the achievement activity diagram in which it use some attributes as shown above.

Thirteen, the set achievement use case which is implemented by the set achievement operation and is applied in the achievement activity diagram in which it use some attributes as shown above.

			Employee job data class									
Use Case	Operation	Check with activity	ID	Name	Job tittle	Job grade	Department	Salary	Job description	Bonus	Housing allowance	Gender
Update employee job data	Update employee job data	✓				Grade-7					EGP 5,000	

Date of birth	Address	Phone number	Marital status	Courses needed	Qualifications needed	Contract

			Employee job data class									
Use Case	Operation	Check with activity	ID	Name	Job tittle	Job grade	Department	Salary	Job description	Bonus	Housing allowance	Gender
View monthly salary	Set employee job data	✓										

Date of birth	Address	Phone number	Marital status	Courses needed	Qualifications needed	Contract

			Monthly salary class					Salary increase class				
Use Case	Operation	Check with activity	ID	Name	Taxes	Monthly salary	Date	ID	Name	Amount	Reason	Date
Generate	Generate monthly salary	✓	186	Ahmad Amgad	10%	EGP 5,000	1/1/2018					
	Send salary increase amount	✓								EGP 500		
monthly salary	Send salary deduction amount	✓										
	Send penalty amount	✓										
calculation	Send attendance	✓										
	Send salary data	✓										
	Calculate monthly salary	✓										

	Salary deduction class					Penalty class				
	ID	Name	Amount	Type	Date	ID	Name	Amount	Type	Reason
			EGP 300							
								EGP 200		

Fourteen, the update employee job data use case which is implemented by the update employee job data operation and is applied in the update employee job data activity diagram in which it use some attributes as shown above.

Fifteen, the view monthly salary use case which is implemented by the view monthly salary operation and is applied in the view monthly salary activity diagram in which it use some attributes as shown above.

Sixteen, the generate monthly salary use case which is implemented by the generate monthly salary, send salary increase amount, send salary deduction amount, send penalty amount, send incentive amount, send attendance, send salary data, calculate monthly salary and is applied in the monthly salary calculations activity diagram in which it use some attributes as shown above.

Attendance class					
ID	Name	Amount	Type	Reason	Date
		EGP 200			

			Management report class				Monthly salary class				
Use Case	Operation	Check with activity	ID	Name	Report	Date	ID	Name	Taxes	Monthly salary	Date
Generate	Generate managment report	✓	186	Ahmad Amgad							
	Send monthly salary	✓								EGP 5,000	
management	Send appraisal	✓									
	Send achievements	✓									
report	Send attendance	✓									
	Send acheivement needed data	✓									
	Prepare management report	✓			2	۷/۱/2018					

Appraisal class				Achievement class					
ID	Name	Appraisal	Date	ID	Name	Courses taken	Qualifications taken	Achievement made	Date
		Document							
							SAP certificate		

Attendance class								
ID	Name	Date	Arrival time	Departure time	Absence	Overtime	Lateness	Early leave
					2	1	1	3

Employee job data								
ID	Name	Job tittle	Job grade	Department	Salary	Job description	Bonus	Housing allowance

			Gender	Date of birth	Address	Phone number	Marital status	Courses needed	Qualifications needed	Contract
								Excel Expert	SAP Certificate	

			Training class				
Use Case	Operation	Check with activity	ID	Name	Type	Date	Time
Design training	Set training	✓	186	Islam Heikal	Course	6/11/2017	6:00pm - 9:00pm

			Training class				
Use Case	Operation	Check with activity	ID	Name	Type	Date	Time
View scheduled training	Display training schedules	✓	186	Islam Heikal	Course	6/11/2017	6:00pm - 9:00pm

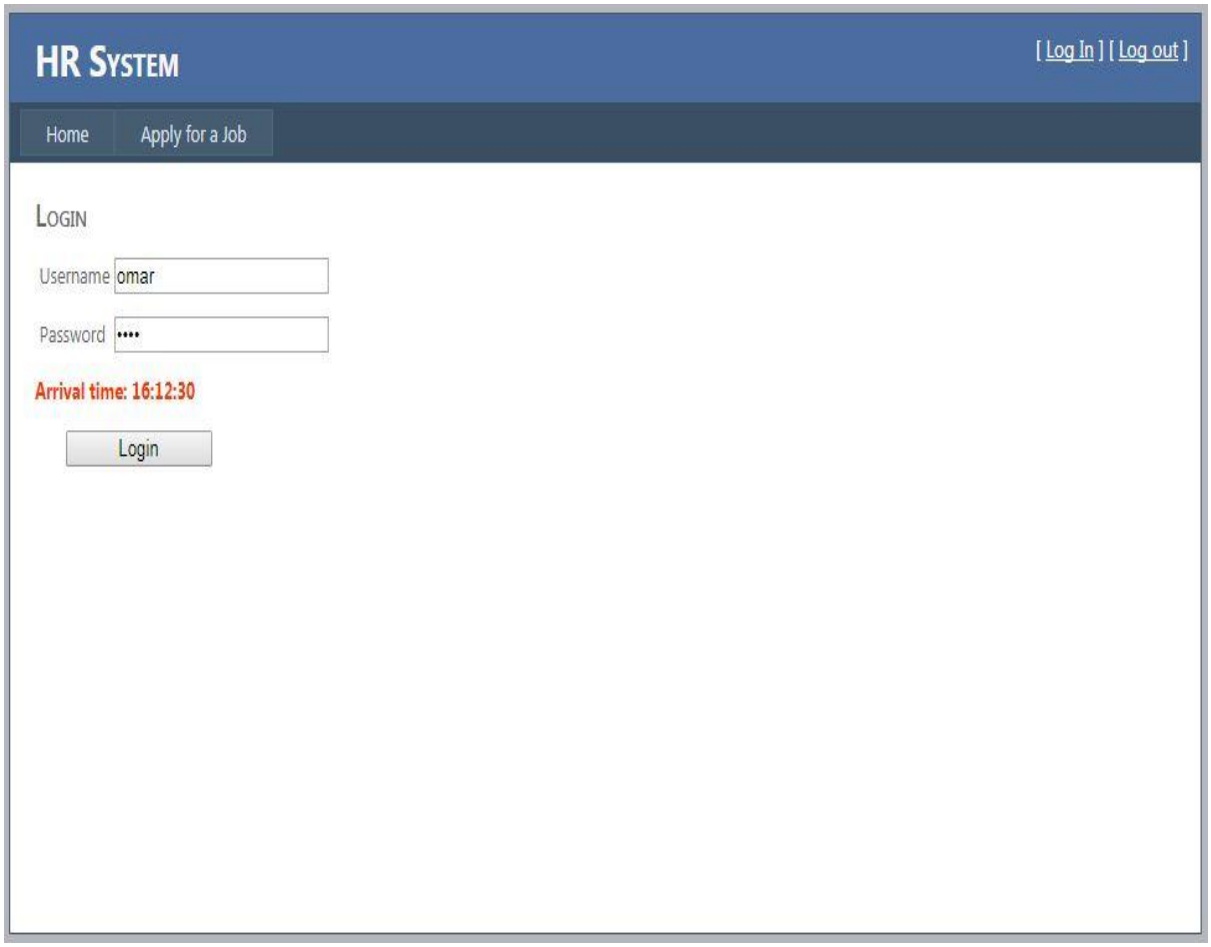
Seventeen, the generate management report use case which is implemented by the generate management report, send salary appraisal, send salary monthly salary, send attendance, send achievement needed data, send achievements made, prepare management report and is applied in the management report activity diagram in which it use some attributes as shown above.

Eighteen, the set training use case which is implemented by the set training operation and is applied in the set training activity diagram in which it use some attributes as shown above.

Nineteen, the view training use case which is implemented by the view training operation and is applied in the view training activity diagram in which it use some attributes as shown above.

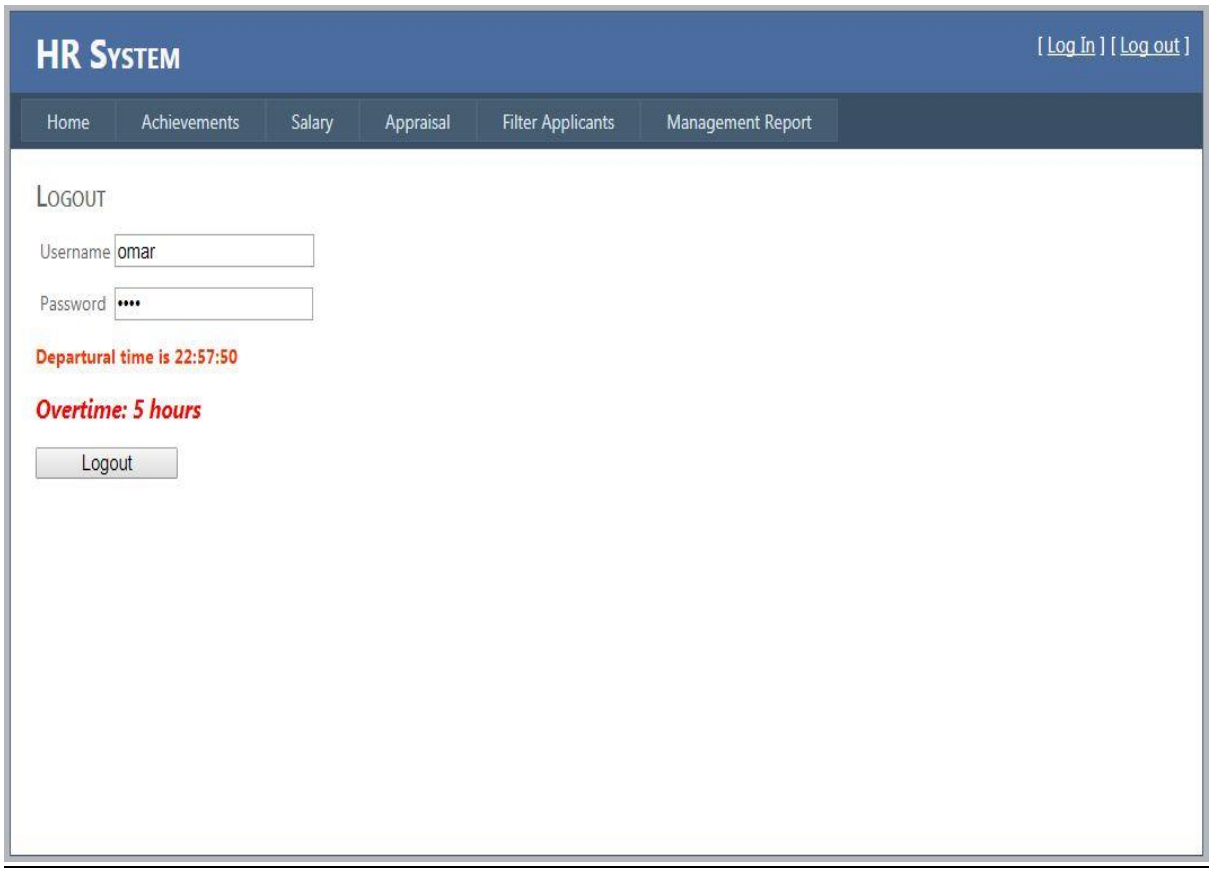
16.0 Screen format design

Log in screen:



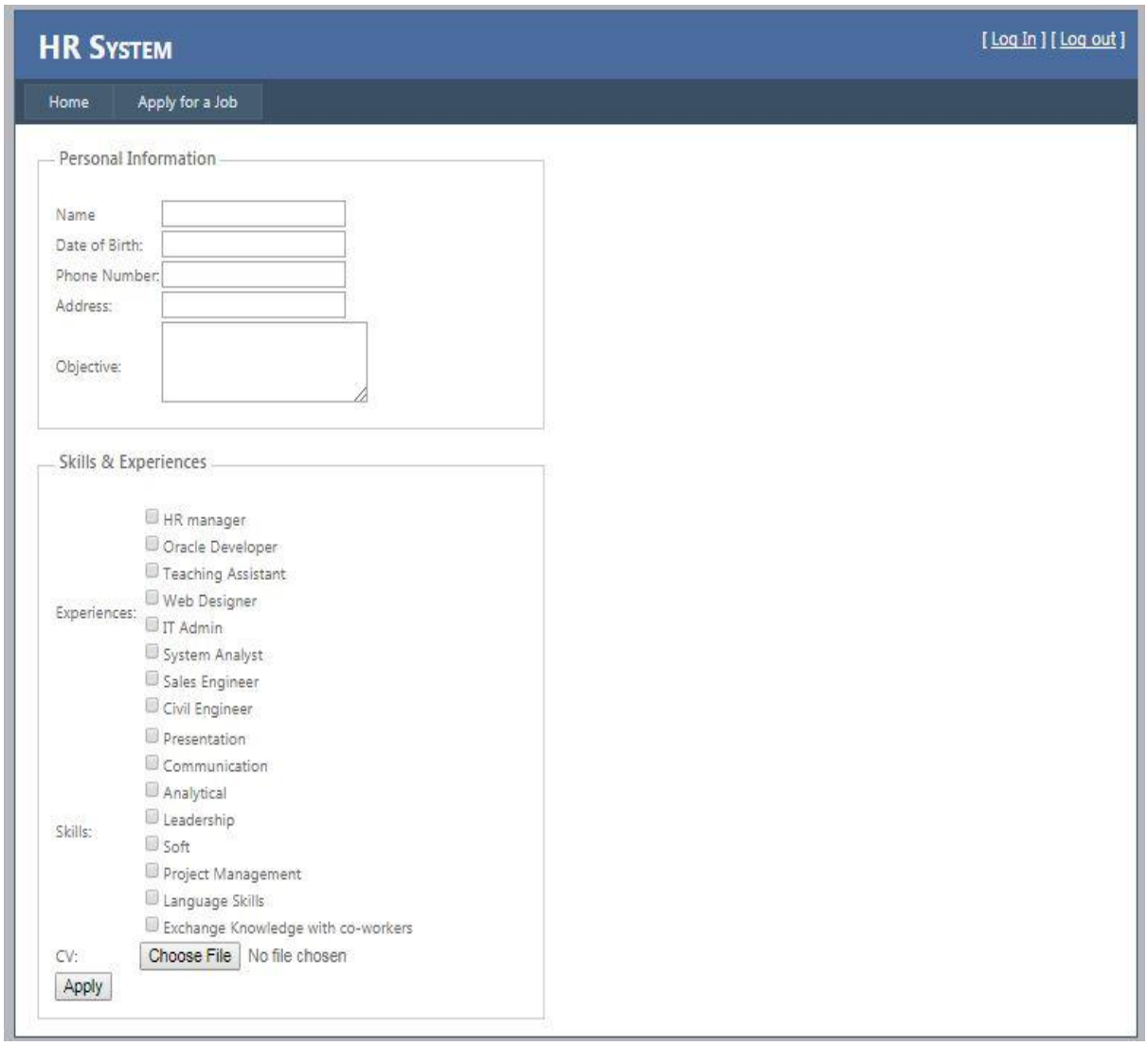
Any normal employee, manager, HR employee or finance employee must login to the system with their username and password in order to access the system features. Therefore, the system set the arrival time which is when he/she logged in the system calculates his/her absence or lateness based on his/her arrival time then displays the arrival time and the hours of lateness.

Log out screen:



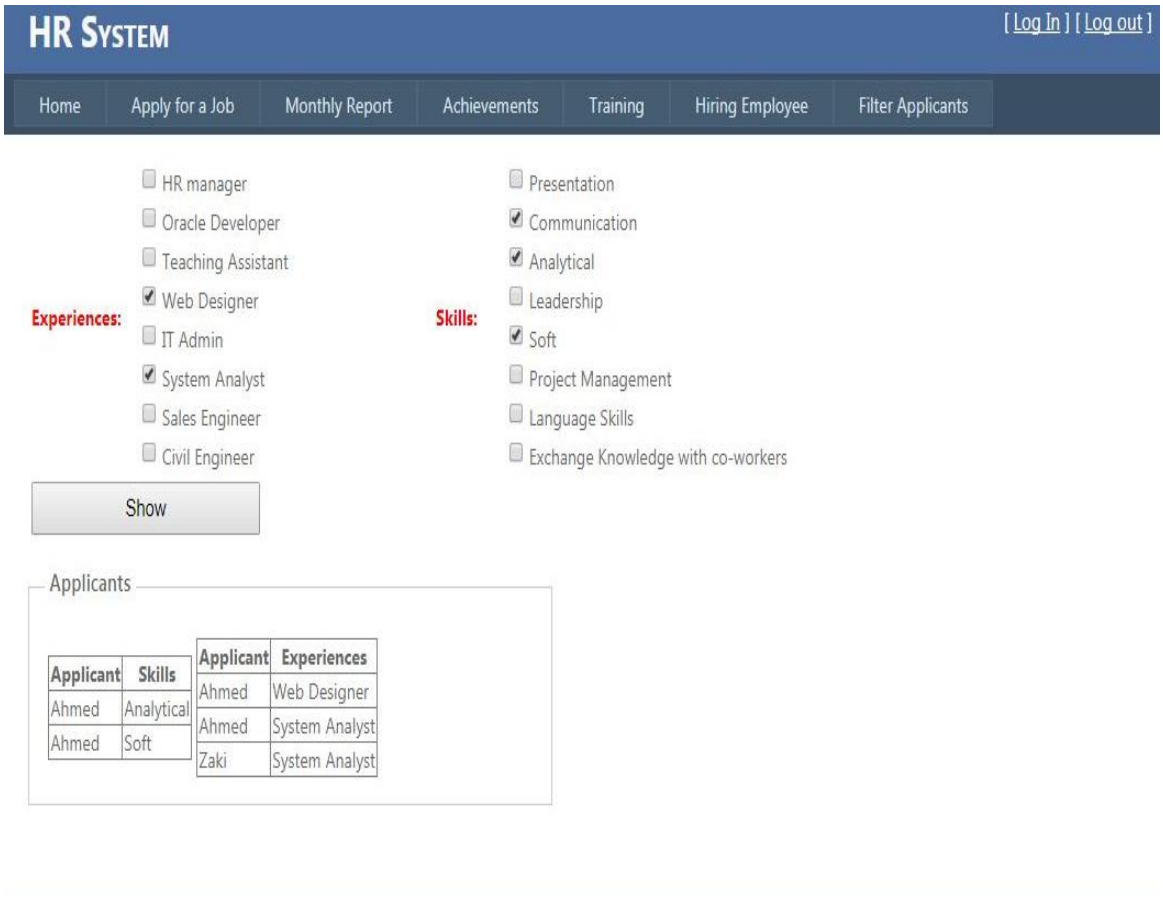
Any normal employee, manager, HR employee or finance employee must logout from the system. Therefore, the system set the departure time which is when he/she logged out the system calculates his/her overtime or early leave based on his/her departure time then displays the departure time and the hours of overtime or early leave.

Fill the applicant form & upload CV screen:



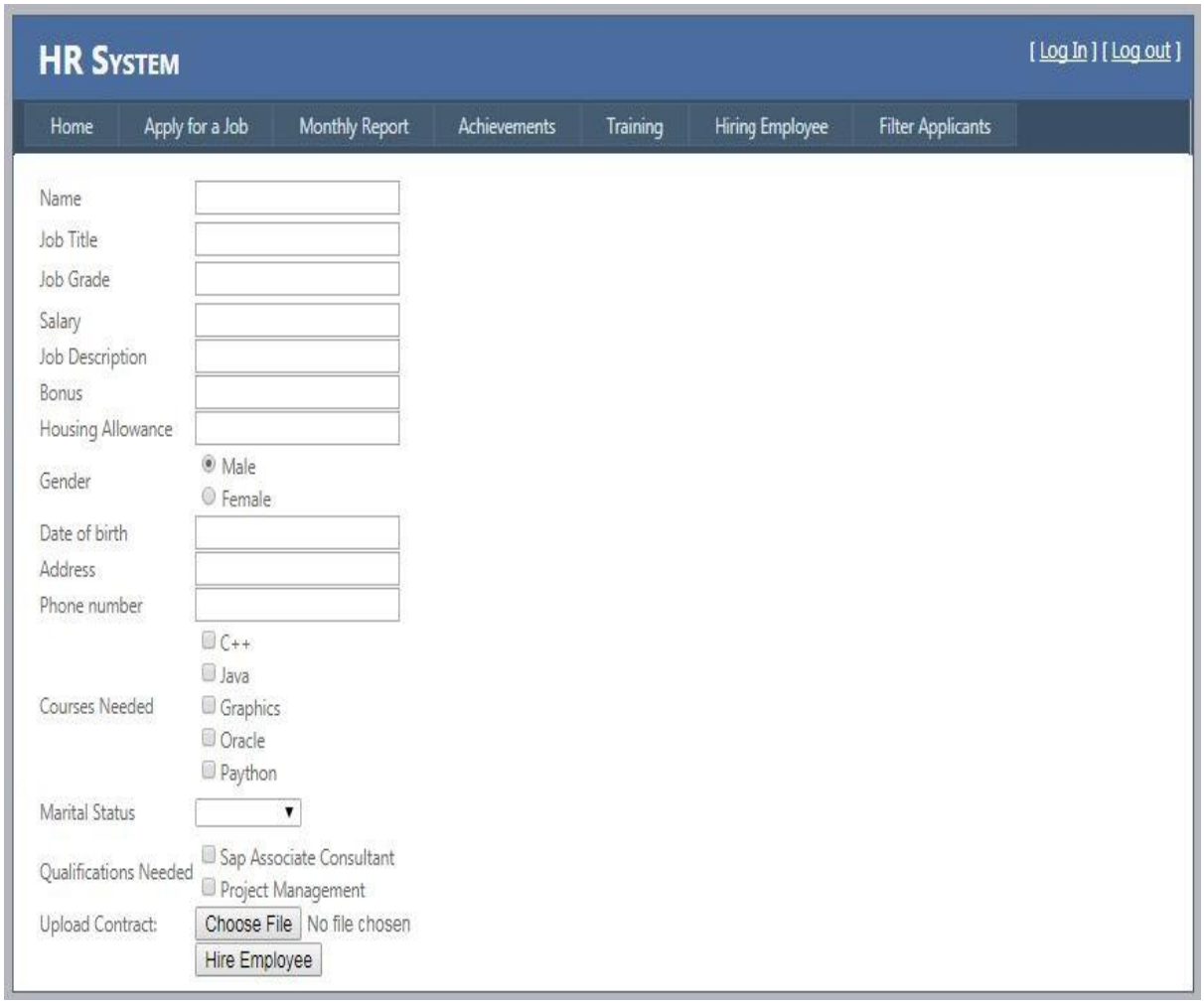
This screen serves the process of filling the application form and uploading the CV by the applicant in which the applicant fills his/her name, date of birth, phone number, objectives, experiences & skills then upload his/her CV in order to use these data further for the HR department to filter applicants to hire the best employee for a specific job or role with the needed experiences and skills.

Filter applicant screen:



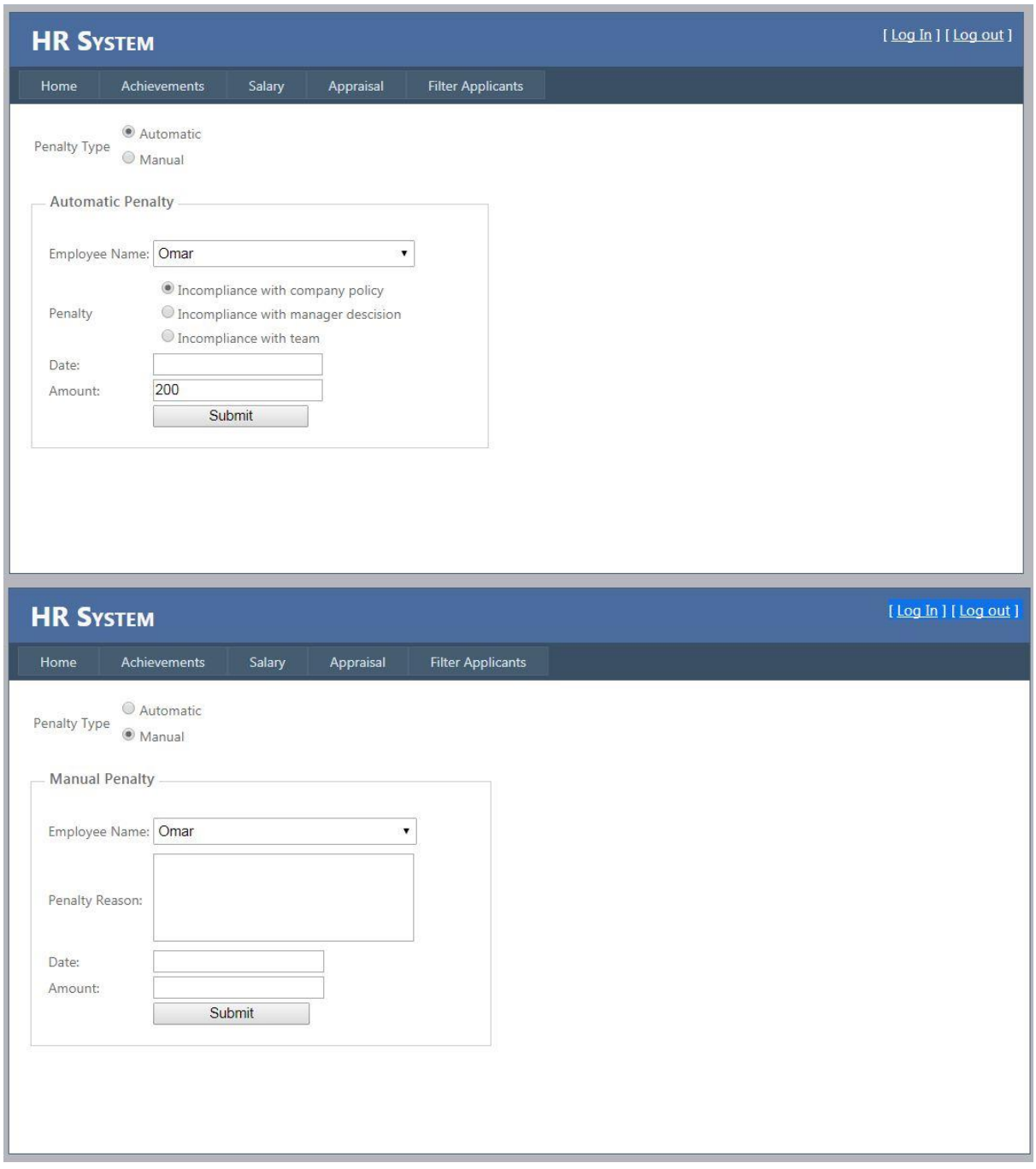
This screen serves the filtering of applicants by the HR department in which HR department must filter all applicants' experiences and skills in order to find the best applicant to hire for a specific job or role. Therefore, the system search for applicants who have skills and experiences that meets those who the HR department needs.

Hire new employee and upload his/her contract screen:



This screen serves hiring the new employee process in which HR department must set his/her data to hire the new employee such as employee ID, name, job tittle, job grade, salary, job description, department, bonus, housing allowance, gender, date of birth, address, phone number, marital status, courses needed & qualifications needed then they upload his/her contract to save a soft copy form the contract of the new employee in the system.

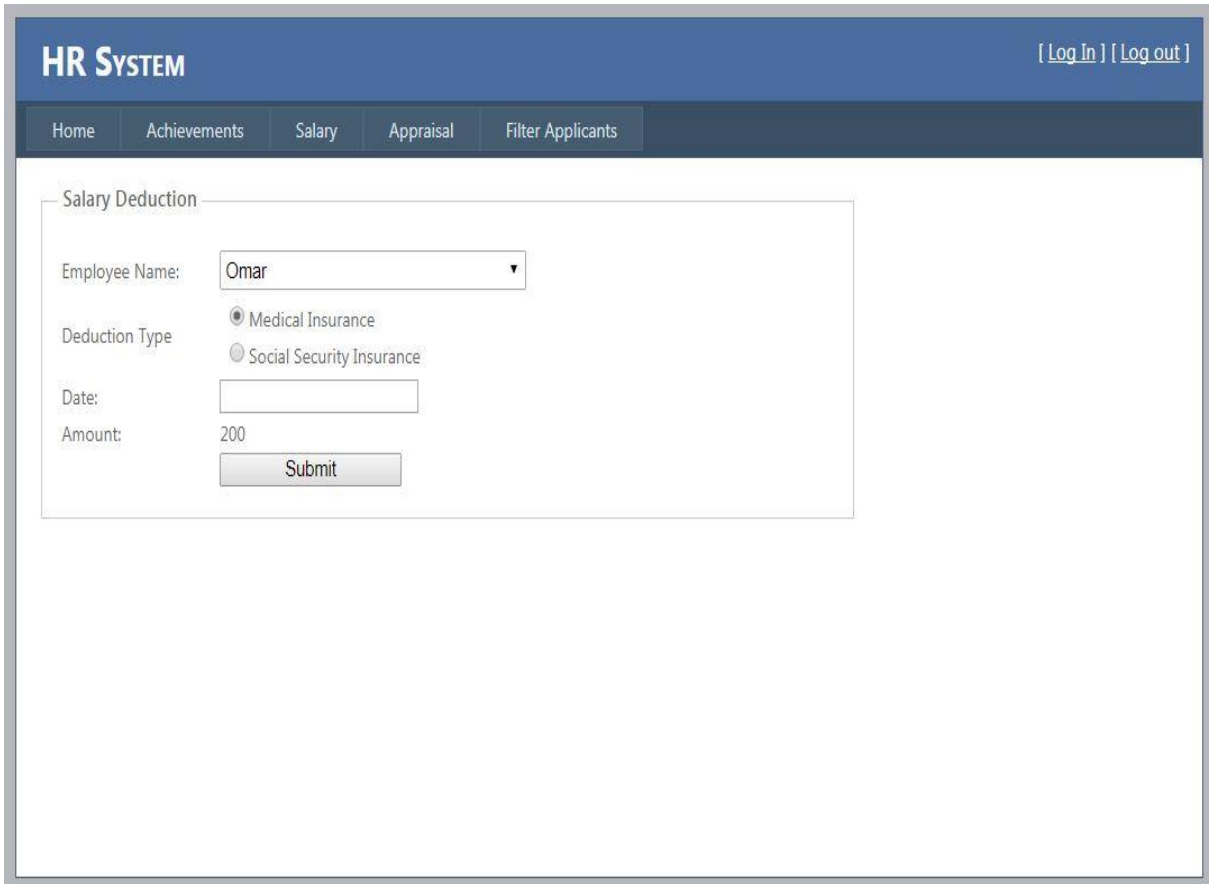
Setting penalty screens:



This screen serves **setting a penalty** on an employee in which the manager must **choose the penalty type** (Automatic penalty or Manual penalty) for which if he/she choose either **automatic penalty** the system **displays automatic penalty format** in order to **enter the automatic penalty** by choosing the employee name (the employee whom the manager want to penalize), date and type such as penalty reason due to incompliance with the company's

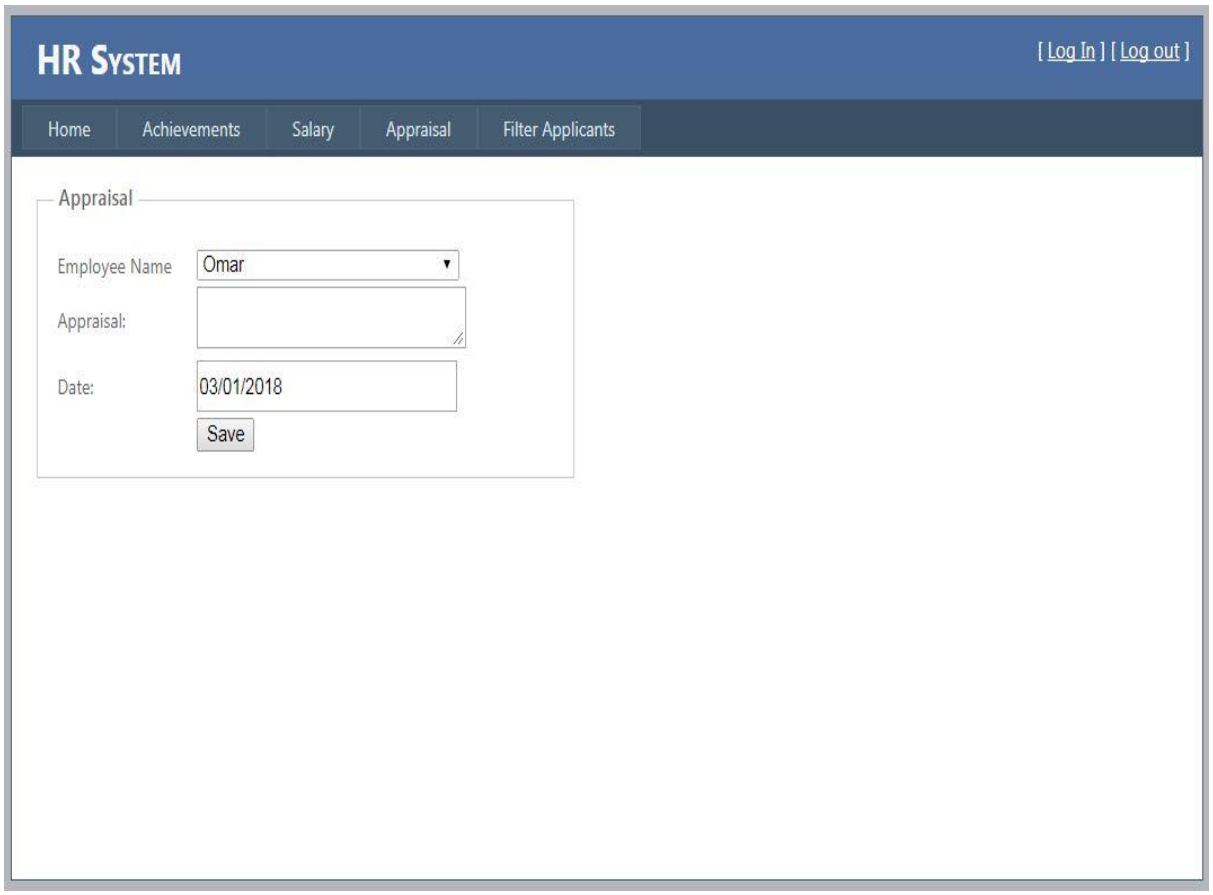
policy, penalty due to incompliance with the manager's decision, penalty due to incompliance with his/her team, etc. that *manager selects the type of penalty the system will automatically deduct the amount stated for this penalty type*, or **manual penalty** the system **displays manual penalty format** in order to **enter the manual penalty** by choosing the employee name (the employee whom the manager want to penalize), date, entering the penalty amount to be deducted and entering the reason for imposing penalty on this employee.

Salary deduction screen:



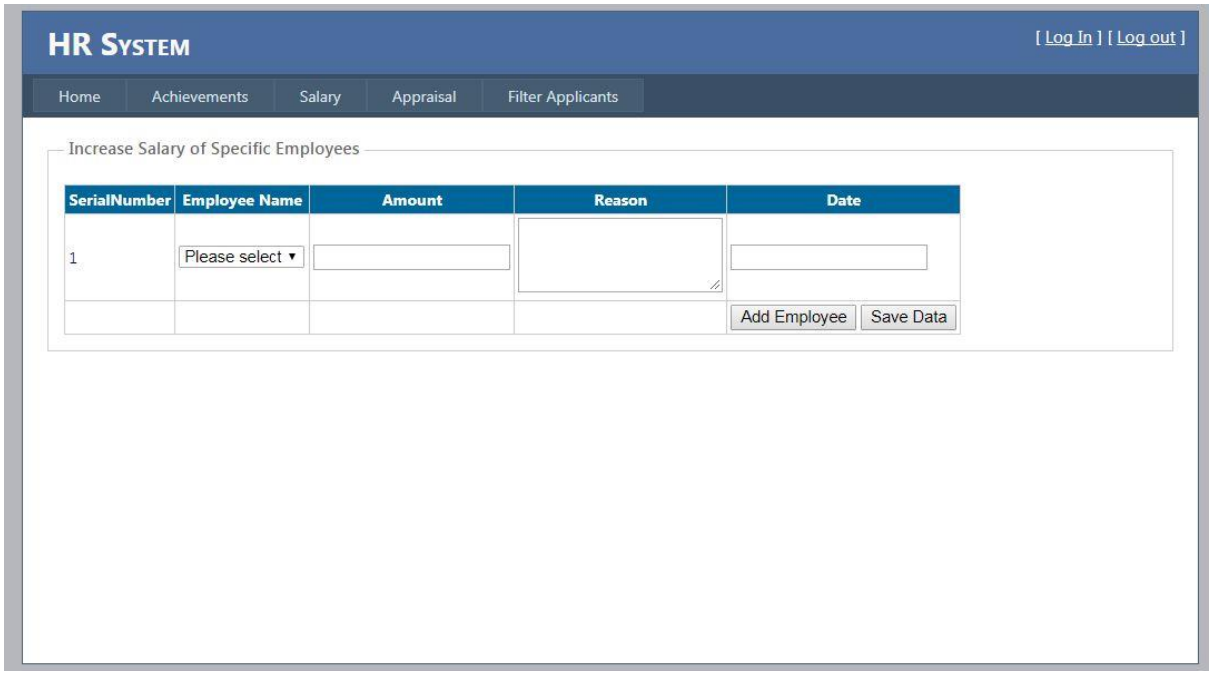
This screen serves the salary deduction imposed by the manager on the employee in which managers can deduct from employee salary for example due to medical insurance, social security insurance, and the deducted amount will be used when calculating monthly salary.

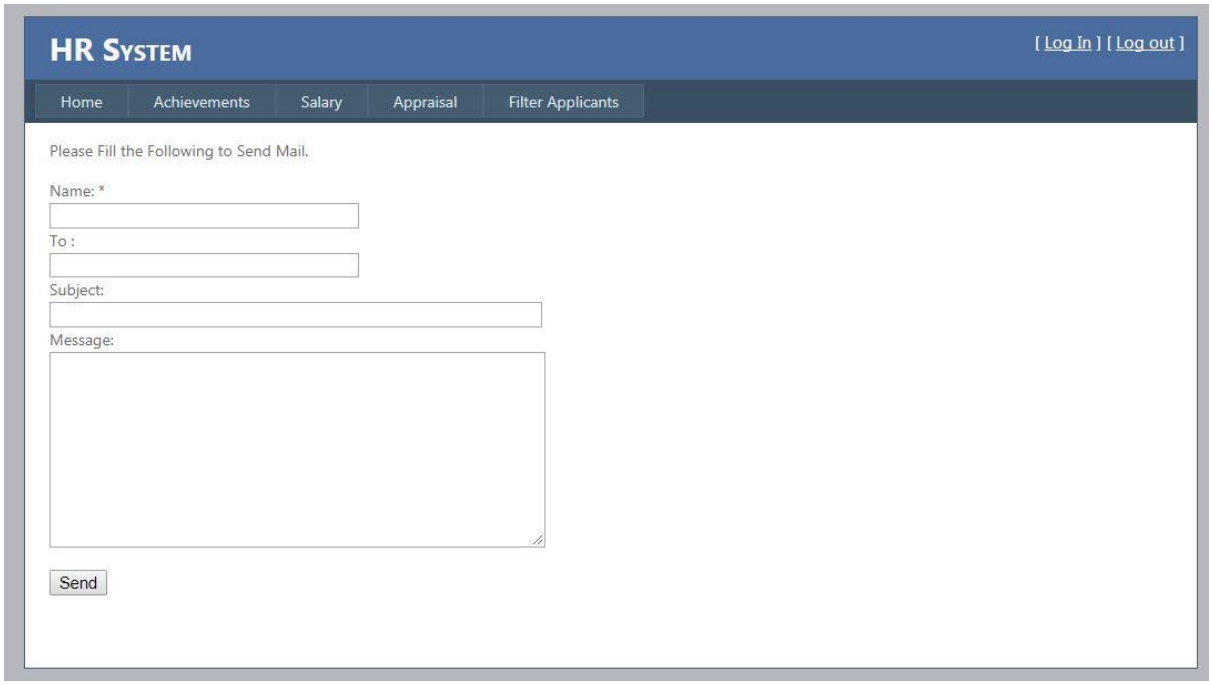
Appraisal screen:



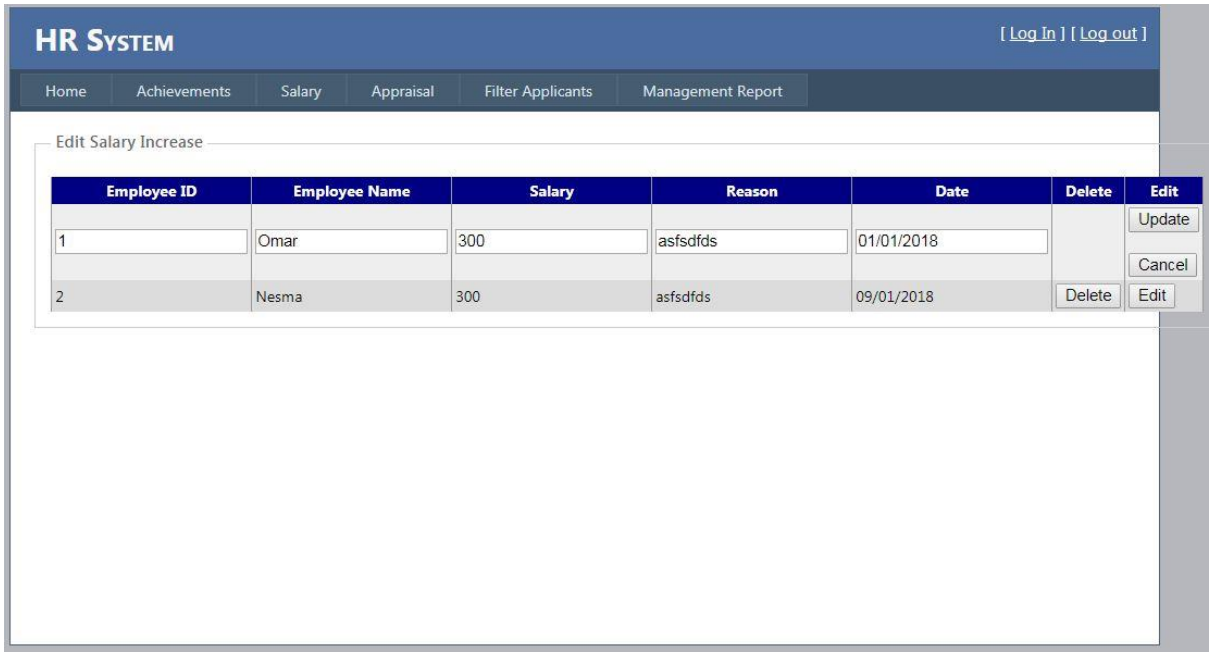
This screen serves the appraisal made by the manager for the employee in which themanager should appraise employee by evaluating and providing feedback on employee job performance that provides the basis for pay increase, bonus and promotions in which it is very important to help employees improve their performance and to help HR to tack employees' performance.

Salary increase screens:



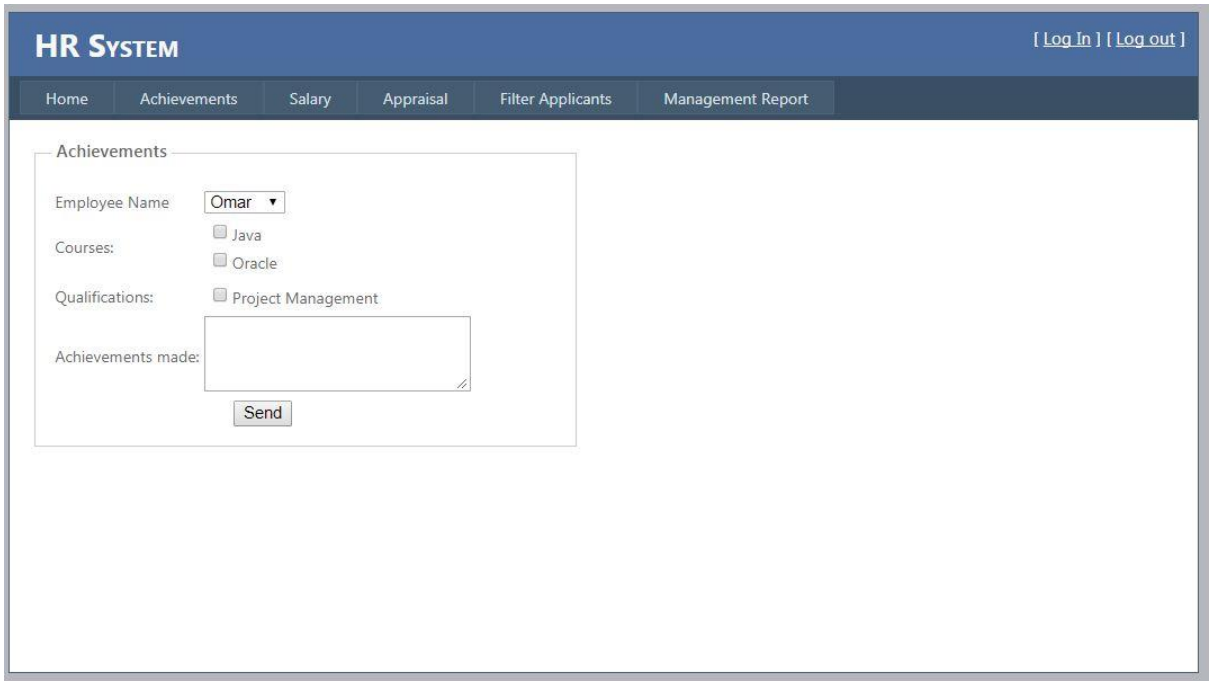


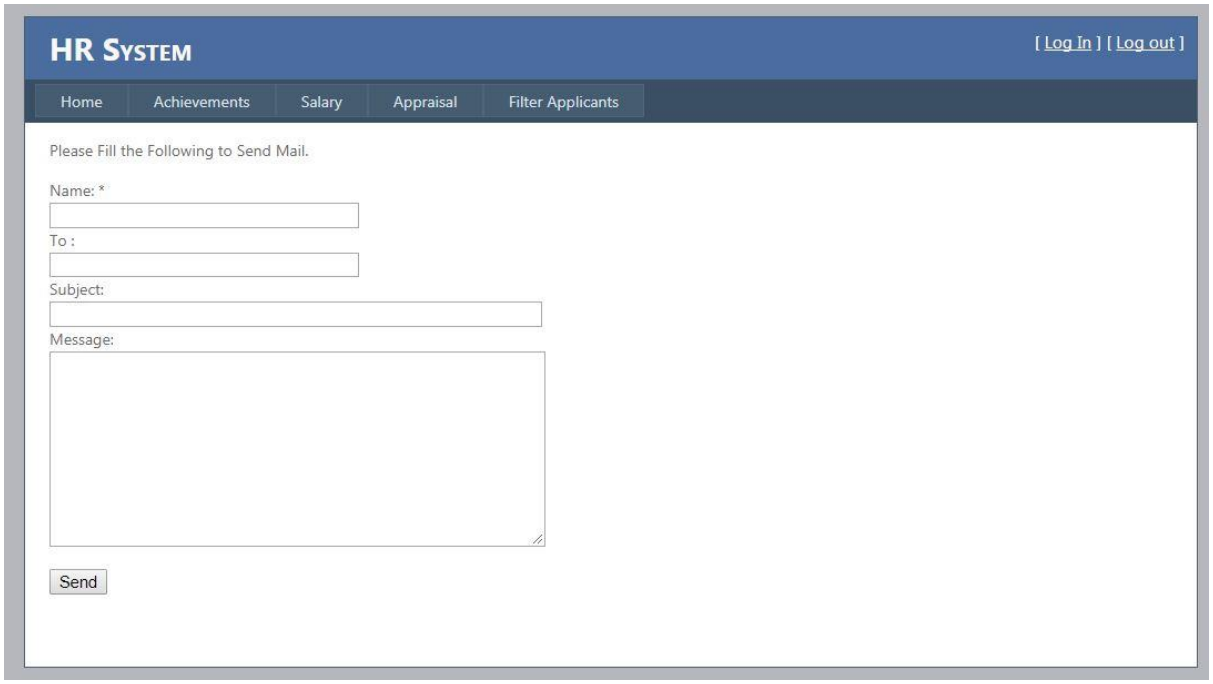
Those screens serve **filling the proposed salary increase for an employee** details containing the employee name, salary increase amount, salary increase reason and date in order to **send these data via email** to the finance department to approve or reject the salary increase.



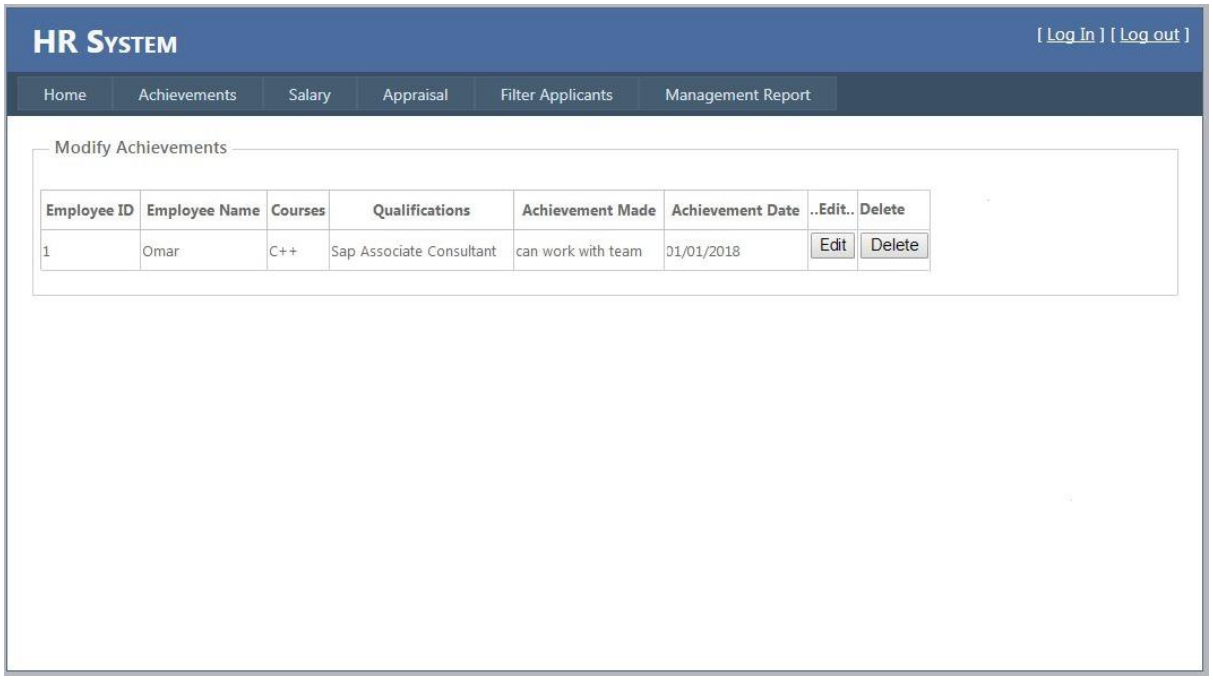
Therefore, the finance department can **approve** the salary increase by either making no change or making some updates on the salary increase or **reject** the salary increase by deleting the salary increase made.

Achievement screens:



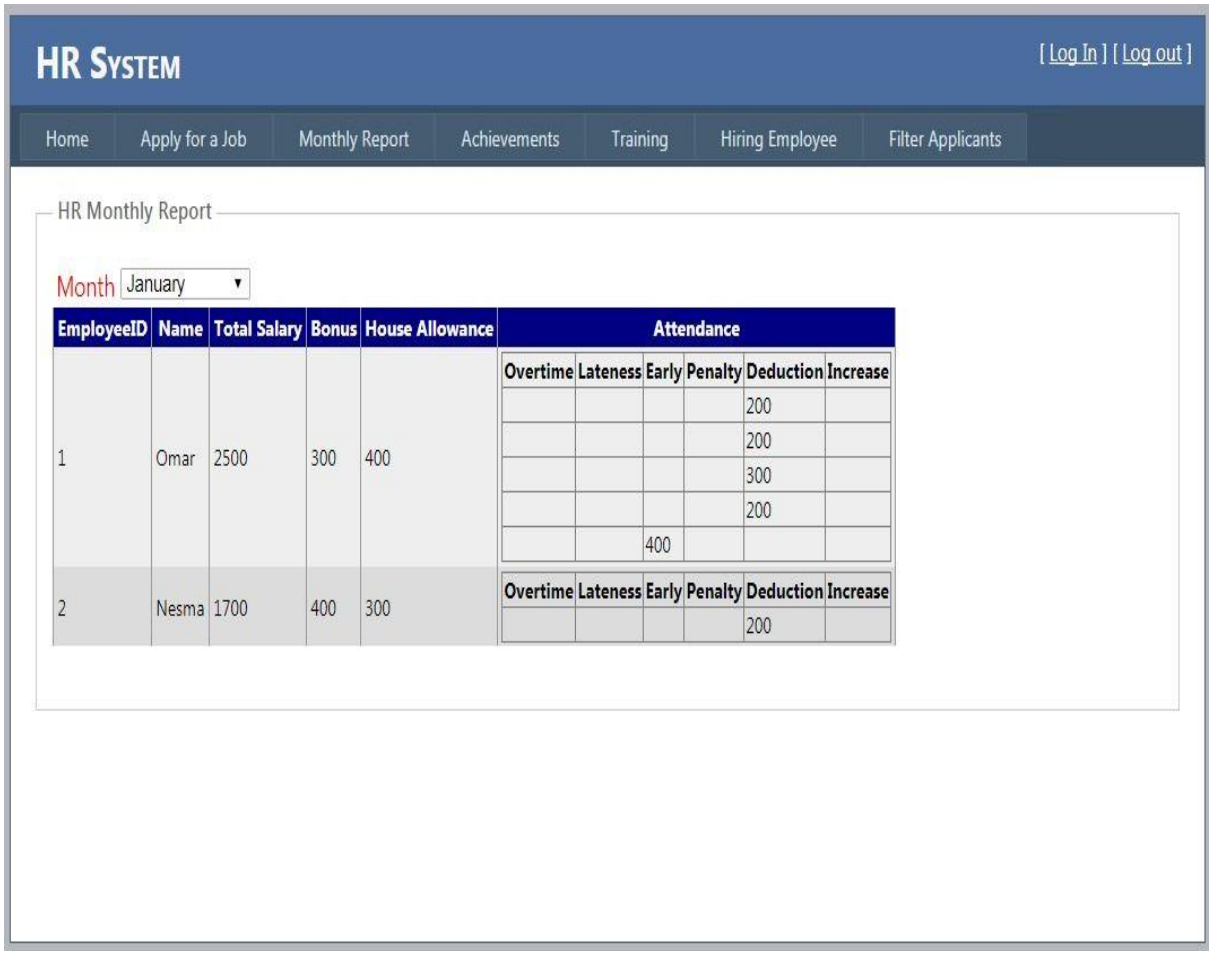


Those screens serve **filling the proposed achievement for an employee** details containing the employee name, course, qualification, accomplishment made and date in order to **send these data via email** to the manager to approve or reject the achievement.



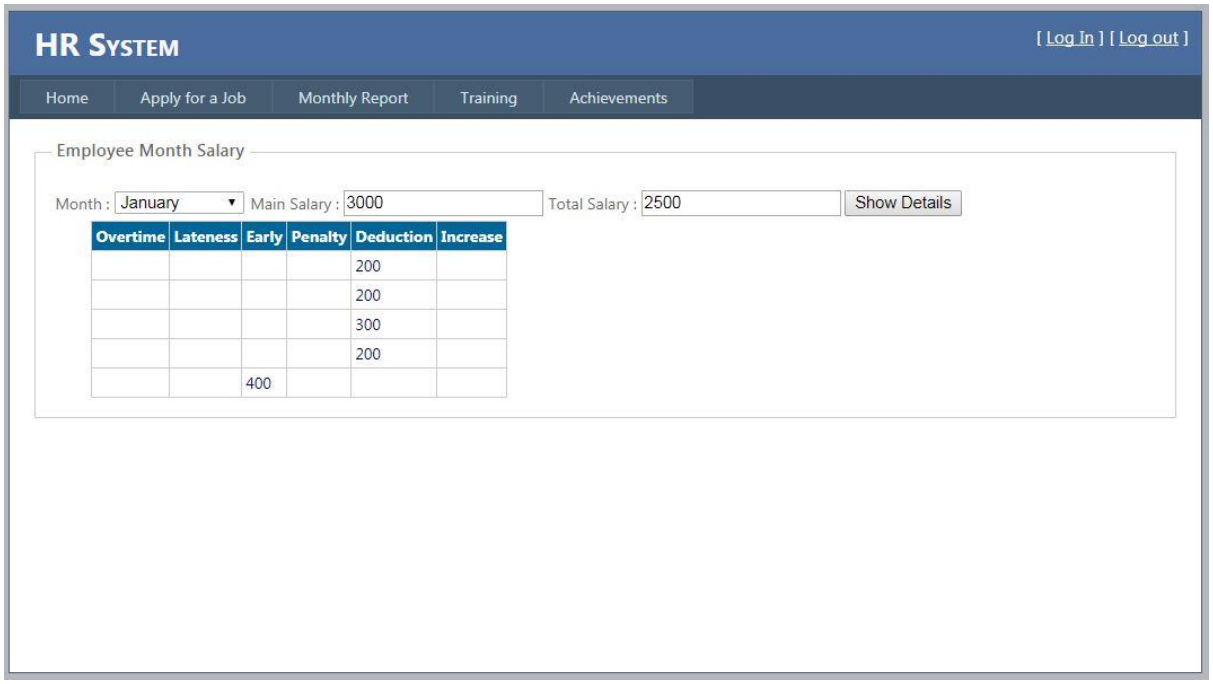
Therefore, the manager can **approve** the achievement by either making no change or making some updates on the achievement or **reject** the achievement by deleting the achievement.

Calculating monthly salary screen:



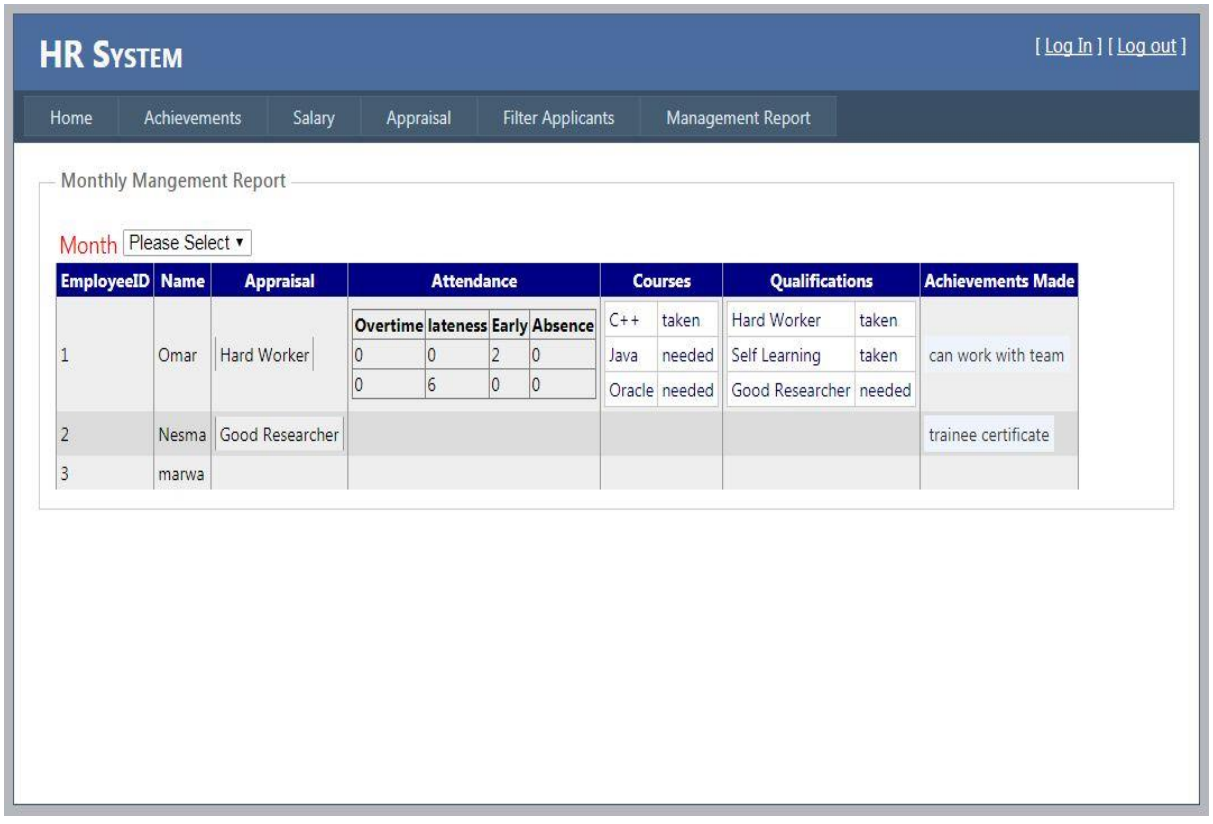
This screen serves calculating monthly salary for employees in which HR department want to **generate the monthly salary** for all employees by selecting the month. After generating monthly salary calculation, the system displays the monthly salary for each employee with the details such as his/her **salary increase (amount)** if there is any new salary increase in this month, **salary deduction (amount)** if there is any new salary deduction in this month, **penalty (amount)** if there is any new penalty in this month, **attendance (lateness, overtime, early leave)** if there is any absence, lateness, overtime and/or early leave in this month **salary data (basic salary, bonus, housing allowance)** if there is any bonus or housing allowance in this month.

View monthly salary screen:



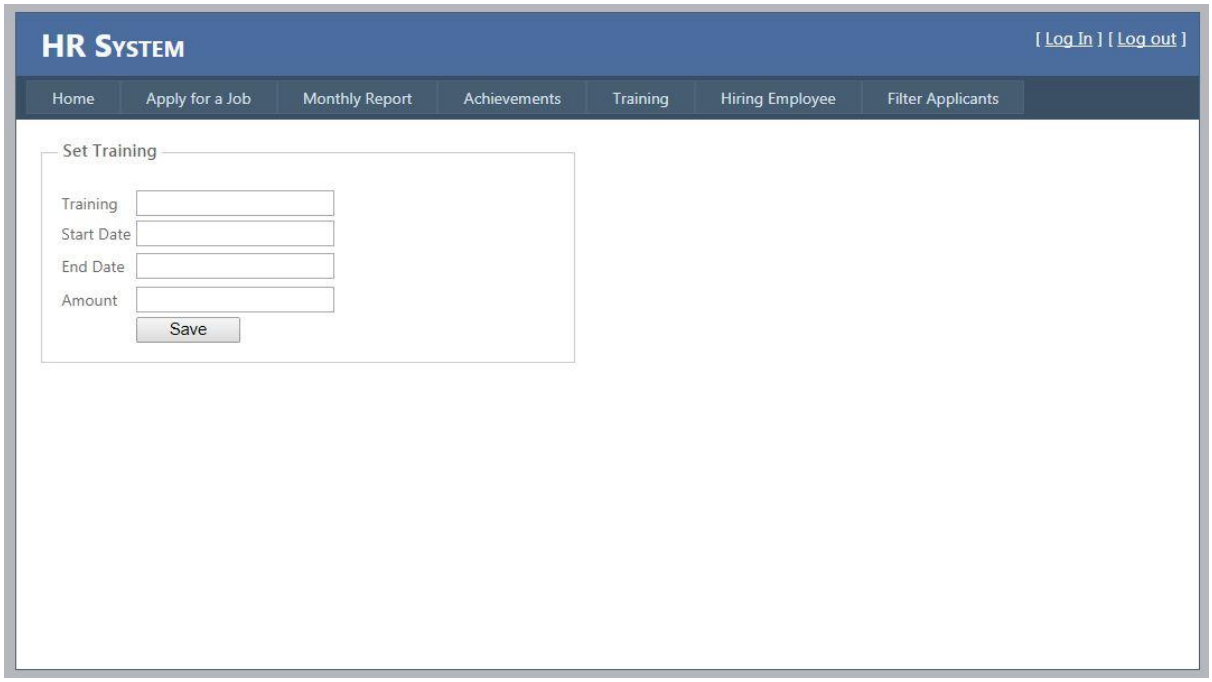
This screen serves viewing the monthly salary by the employee in which the employee selects the month then the system displays his/her main salary, total salary and its details.

Management report screen:

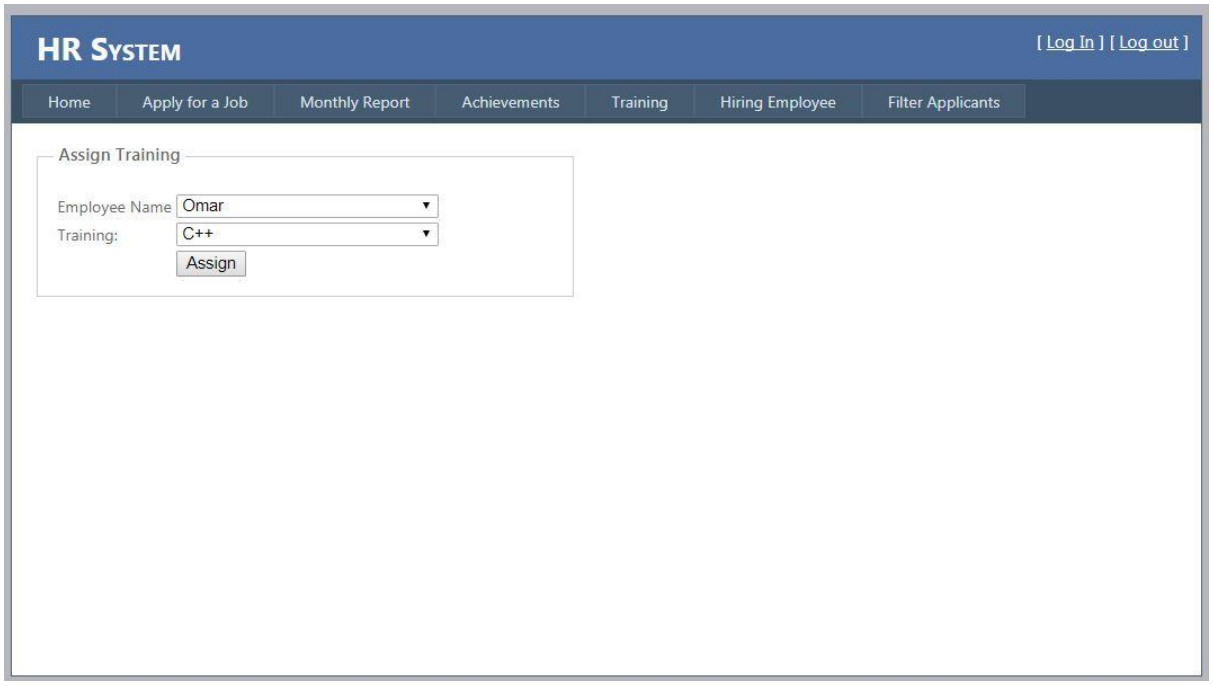


This screen serves generating management report in which HR department want to **generate the management report** for all employees by selecting the month. After generating management report, the system displays the management report for each employee with the details such as his/her **appraisal** if there is any new appraisal in this month, **courses** if there is any new courses taken in this month or still needed to be taken, **accomplishment** if there is any new accomplishment made in this month**, qualifications** if there is any new qualifications taken in this month or still needed to be taken**, and attendance (lateness, overtime, early leave)** if there is any absence, lateness, overtime and/or early leave in this month.

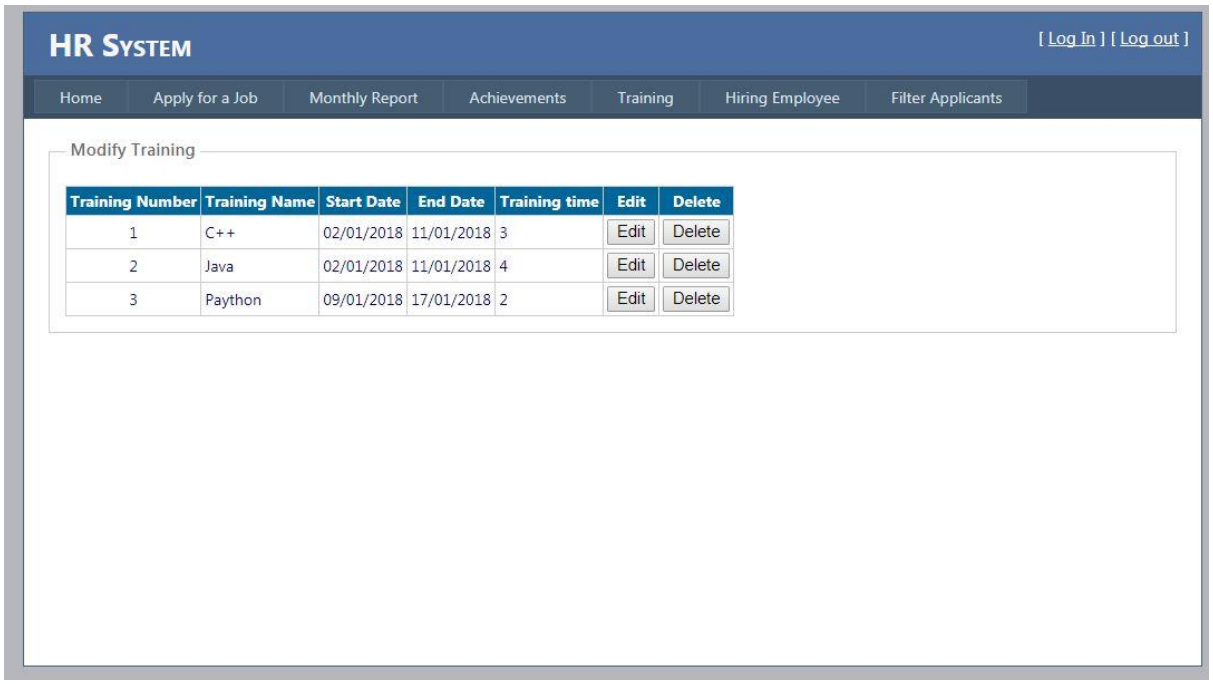
Training screens:



This screen serves setting a new training in which HR department wants to **set new training**, the system **display setting new training format** in order to **enter the new training type, start & end date and the amount of hours.**

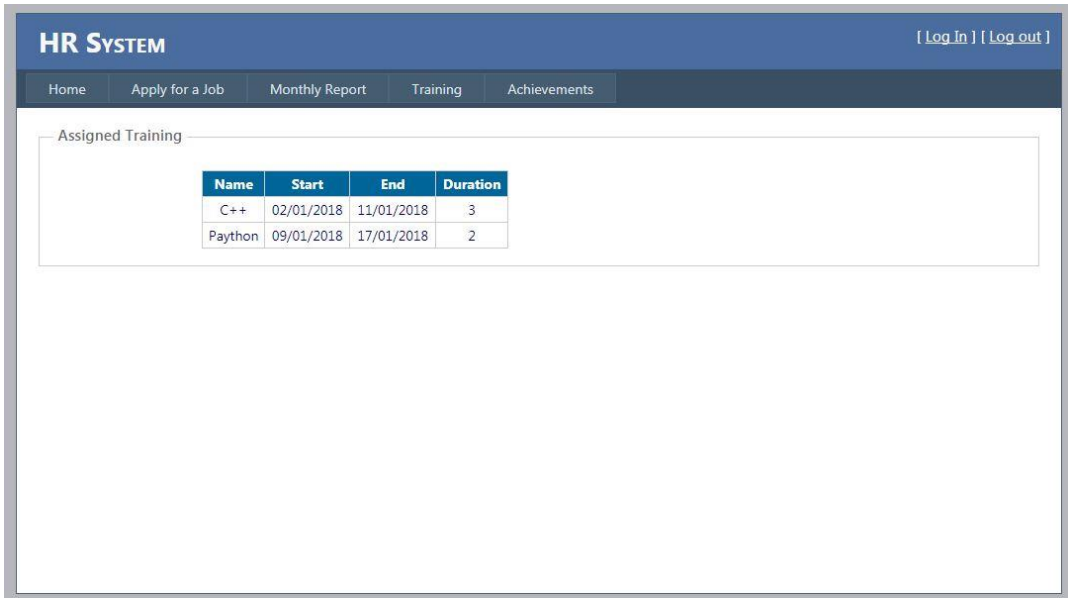


After setting the new training, HR department set the training to the employee in this screen.



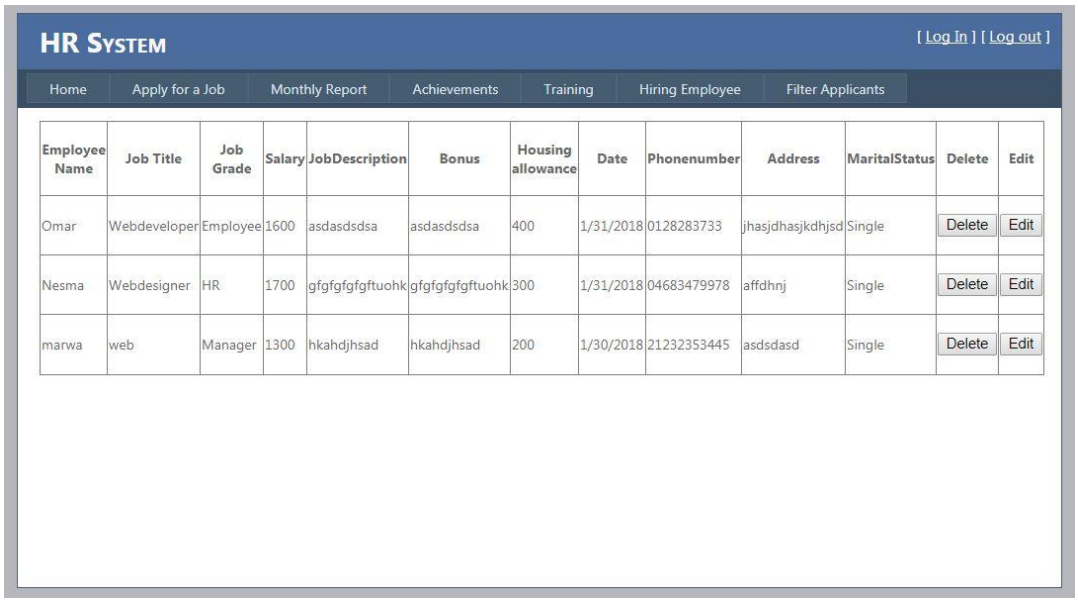
Finally, the HR department can modify or edit training schedules if any change has been happened to the training schedules.

View training screen:



This screen serves viewing the training schedules by the employee in which the employee selects the month then the system displays his/her training schedules.

Updating employee job data screen:



This screen serves updating the employee job data by the HR department in which the HR department edit or delete any attribute for the employee job data.

17.0 Results

Human Resource Management system (HRIS) involves practices used by managers and human resource workers and automated systems to manage people. Examples of Human resource management system practices include performance management, hiring, firing, training, payroll and employee benefits administration. Without organizational coordination through Human resource management system practices, managing personnel is cumbersome, which can leave employees feeling dissatisfied and wanting to work for a more organized firm. **Human Resource Management system** (HRIS) solutions are more than just employee ***data houses***. They also allow employees and managers to take on some of the human resource tasks that directly affect them. These solutions are ***helpful tools*** for performance and attendance tracking, and reporting systems that help with compliance and decision making.

18.0 Future work

The future work will improve the scope the potential of the current human resource management system in which the future work will cover **offering both on-premises & cloud system, improvement of ability to track employees' performance** to support managers & identify top talent by comparing & rating employees across the same dimensions to identify high performers & potential future leaders, **creating career succession planning for employees** in which Succession Planning is one of those areas which can make a big difference inside an organization; put the right employees into the appropriate career paths to reduce the risk of poor business execution, **improving the process of recruitment** (how the applicants apply for a job or a role and how the HR filter applicants in order to hire the best new employee for a specific role or a job in which online assessment such as leadership assessment, problem solving skills assessment, corporate communication skills assessment, technical assessment, and other types of assessments will be also provided in order to filter applicants), **requesting school coverage, vacations, sick leave & early leave** from the system & receive response, **shift planning**, **travel management** in which there is a specific flight & hotel reservation category depending on his/her job grade or title, **storing data about competitors' employees**, and **making a daily online newspaper** in the HR system to retain employees by sharing expertise & knowledge that can reduce training costs with informal learning where experts can create content or video to share expertise, improving communication & better connection of employees across different geographies, and ease the process of social onboarding.

19.0 References

- Almutawa, Z., Muenjohn, N., & Jiaying, Z. (2016). THE EFFECT OF HUMAN RESOURCE MANAGEMENT SYSTEM ON EMPLOYEES' COMMITMENT: THE MEDIATING ROLE OF THE AMO MODEL. Journal Of Developing Areas, 50(6), 17-29. doi:10.1353/jda.2016.0147
- Almutawa, Z., Muenjohn, N., & Jiaying, Z. (2015). UNLOCKING THE BLACK BOX OF THE CONCEPTUAL RELATIONSHIP BETWEEN HRM SYSTEM AND ORGANIZATIONAL PERFORMANCE. Journal Of Developing Areas, 49(6), 413-420.
- Object-Oriented Analysis and Design with Applications, by Grady Booch ,2007 Pearson education
- Peretz H. (2013) Information Technology Systems in the Human Resource Management Area. In: Parry E., Stavrou E., Lazarova M. (eds) Global Trends in Human Resource Management. Palgrave Macmillan, London
- Ferguson, KL & Reio Jr, TG (2010), 'Human resource management systems and firm performance', Journal of Management Development, vol. 29, no. 5, pp. 471-494.
- Structured Analysis and System Specification, by Tom DeMarco, P. J. Plauger, Prentice hall , 1978