Employee Management System

PROJECT REPORT

18CSC202J/ 18AIC203J - OBJECT ORIENTED DESIGN AND PROGRAMMING LABORATORY

(2018 Regulation)

II Year/ III Semester

Academic Year: 2022 -2023

By

Potta Varun Kumar (RA2111026010255)

Kollipara Rohan (RA2111026010263)

Under the guidance of

Dr. M. Uma

Assistant Professor

Department of Computational Intelligence

FACULTY OF ENGINEERING AND TECHNOLOGY SCHOOL OF COMPUTING

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

Kattankulathur

NOVEMBER 2022

BONAFIDE

This is to certify that **18CSC202J - OBJECT ORIENTED DESIGN AND PROGRAMMING LABORATORY project report** titled "**Employee Management System"** is the bonafide work of **Potta Varun Kumar (RA2111026010255) Kollipara Rohan (RA2111026010263)**who undertook the task of completing the project within the allotted time.

Signature of the Guide	**Signature of the II Year Academic Advisor**
Dr. M. Uma	------------------------- **Assistant**
Professor	**Professor and Head**
Department of CINTEL,	Department of CINTEL
SRM Institute of Science and Technology	SRM Institute of Science and Technology

About the course:-

18CSC202J/ 8AIC203J - Object Oriented Design and Programming are 4 credit courses with **L T P C as 3-0-2-4** (Tutorial modified as Practical from 2018 Curriculum onwards)

Objectives:

The student should be made to:

- Learn the basics of OOP concepts in C++
- Learn the basics of OOP analysis and design skills.
- Be exposed to the UML design diagrams.
- Be familiar with the various testing techniques

Course Learning Rationale (CLR): The purpose of learning this course is to:

1. Utilize class and build domain model for real-time programs
2. Utilize method overloading and operator overloading for real-time application development programs
3. Utilize inline, friend and virtual functions and create application development programs
4. Utilize exceptional handling and collections for real-time object-oriented programming applications
5. Construct UML component diagram and deployment diagram for design of applications
6. Create programs using object-oriented approach and design methodologies for real-time application development

Course Learning Outcomes (CLO): At the end of this course, learners will be able to:

1. Identify the class and build domain model
2. Construct programs using method overloading and operator overloading
3. Create programs using inline, friend and virtual functions, construct programs using standard templates
4. Construct programs using exceptional handling and collections
5. Create UML component diagram and deployment diagram
6. Create programs using object-oriented approach and design methodologies

Table 1: Rubrics for Laboratory Exercises

(Internal Mark Split-up :- As per Curriculum)

CLAP-1	5=(2(E-lab Completion) + 2(Simple Exercises)(from CodeZinger, and any other coding platform) + 1(HackerRank/Code chef/LeetCode Weekend Challenge)	Elab test
CLAP-2	7.5=(2.0(E-lab Completion)+ 2.0 (Simple Exercises)(from CodeZinger, and any other coding platform) + 3.5 (HackerRank/Code chef/LeetCode Weekend Challenge)	Elab test
CLAP-3	7.5=(2.0(E-lab Completion(80 Pgms)+ 2.0 (Simple Exercises)(from CodeZinger, and any other coding platform) + 3.5 (HackerRank/Code chef/LeetCode Weekend Challenge)	**2 Mark -** E-lab Completion **80 Program** Completion from 10 Session (Each session min 8 program) **2 Mark -** Code to UML conversion GCR Exercises **3.5 Mark - Hacker Rank** Coding challenge completion
CLAP-4	5= 3 (Model Practical) + 2(Oral Viva)	• **3 Mark** – Model Test • **2 Mark** – Oral Viva
Total	25	

COURSE ASSESSMENT PLAN FOR OODP LAB

S.No	List of Experiments	Course Learning Outcomes (CLO)	Blooms Level	PI	No of Programs in each session
1.	Implementation of I/O Operations in C++	CLO-1	Understand	2.8.1	10
2.	Implementation of Classes and Objects in C++	CLO-1	Apply	2.6.1	10
3,	To develop a problem statement. 1. From the problem statement, Identify Use Cases and develop the Use Case model. 2. From the problem statement, Identify the conceptual classes and develop a domain model with a UML Class diagram.	CLO-1	Analysis	4.6.1	Mini Project Given
4.	Implementation of Constructor Overloading and Method Overloading in C++	CLO-2	Apply	2.6.1	10
5.	Implementation of Operator Overloading in C++	CLO-2	Apply	2.6.1	10
6.	Using the identified scenarios, find the interaction between objects and represent them using UML Sequence diagrams and Collaboration diagrams	CLO-2	Analysis	4.6.1	Mini Project Given
7.	Implementation of Inheritance concepts in C++	CLO-3	Apply	2.6.1	10
8.	Implementation of Virtual function & interface concepts in C++	CLO-3	Apply	2.6.1	10
9.	Using the identified scenarios in your project, draw relevant state charts and activity diagrams.	CLO-3	Analysis	4.6.1	Mini Project Given
10.	Implementation of Templates in C++	CLO-3	Apply	2.6.1	10
11.	Implementation of Exception of Handling in C++	CLO-4	Apply	2.6.1	10
12.	Identify the User Interface, Domain objects, and Technical Services. Draw the partial layered, logical architecture diagram with UML package diagram notation such as Component	CLO-5	Analysis	4.6.1	Mini Project Given
13.	Implementation of STL Containers in C++	CLO-6	Apply	2.6.1	10
14.	Implementation of STL associate containers and algorithms in C++	CLO-6	Apply	2.6.1	10
15.	Implementation of Streams and File Handling in C++	CLO-6	Apply	2.6.1	10

LIST OF EXPERIMNENTS FOR UML DESIGN AND MODELLING:

To develop a mini-project by following the exercises listed below.

1. To develop a problem statement.
2. Identify Use Cases and develop the Use Case model.
3. Identify the conceptual classes and develop a domain model with UML Class diagram. 4. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence diagrams.
5. Draw relevant state charts and activity diagrams.
6. Identify the User Interface, Domain objects, and Technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation.

Suggested Software Tools for UML:

StarUML, Rational Suite, Argo UML (or) equivalent, Eclipse IDE and Junit

INDEX

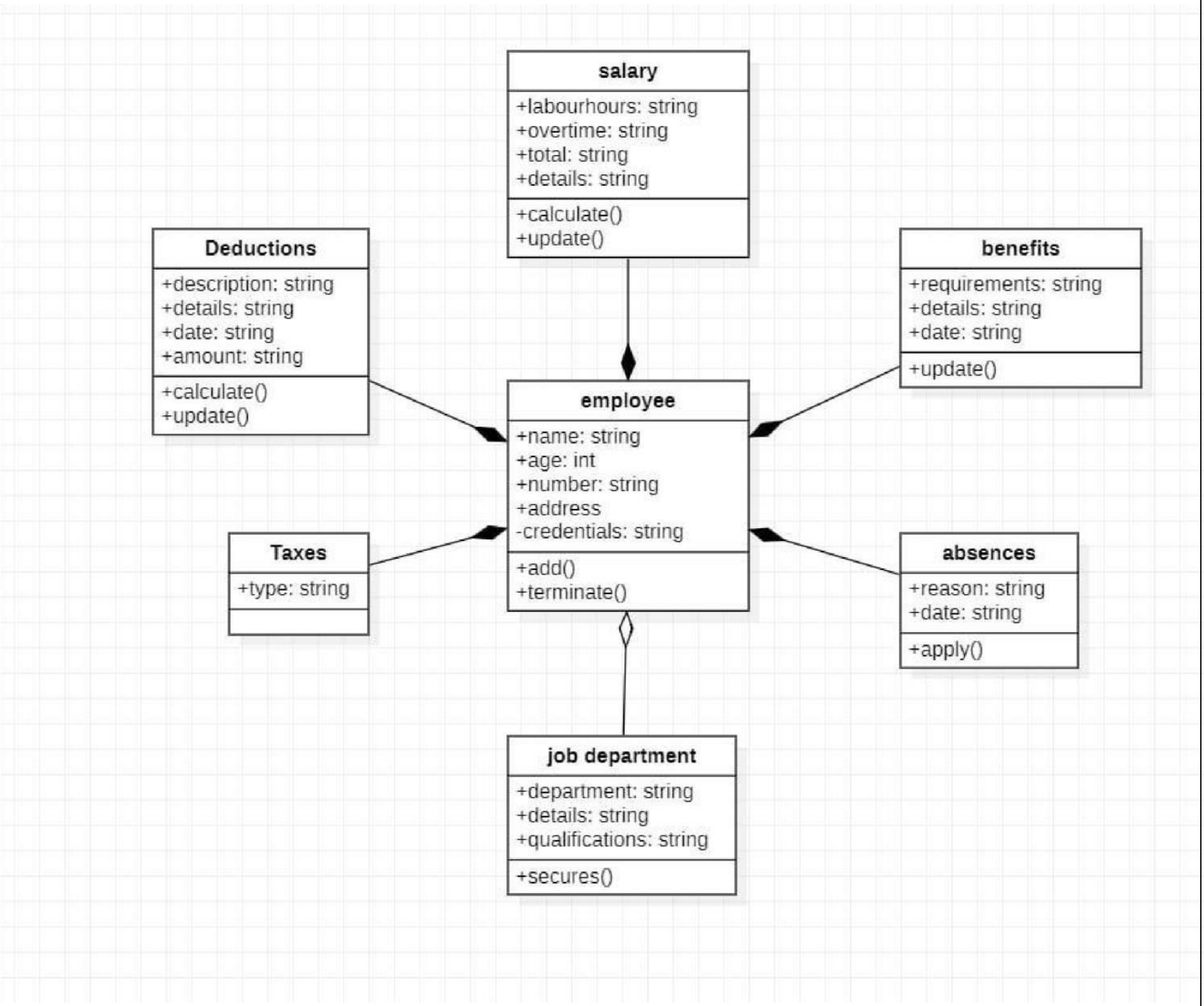
Project Description :

The "EMPLOYEE MANAGEMENT SYSTEM" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides an error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is userfriendly.

This system enables employees to perform their own profile. It enables the automation of workflow notifications and leave requests. Workflow notifications from administrators are stored in the backend and notified to employees, once employees log in to the system. Leave request made by the employee is placed for administrator approval, the administrator module checks up with the leave availability and approves or rejects the request.

Class Diagram :



:

Employee Management System Class Diagram describes the structure of an Employee Management System classes, their attributes, operations (or methods), and the relationships among objects. The main classes of the Employee Management System are Employee, Salary, Leaves, Experience. Login, Attendance.

Classes of Employee Management System Class Diagram:

- Employee Class Manage all the operations of Employee
- Salary Class Manage all the operations of Salary
- Leaves Class: Manage all the operations of Leaves
- Experience Class Manage all the operations of Experience
- Login Class Manage all the operations of Login
- Attendance Class: Manage all the operations of Attendance

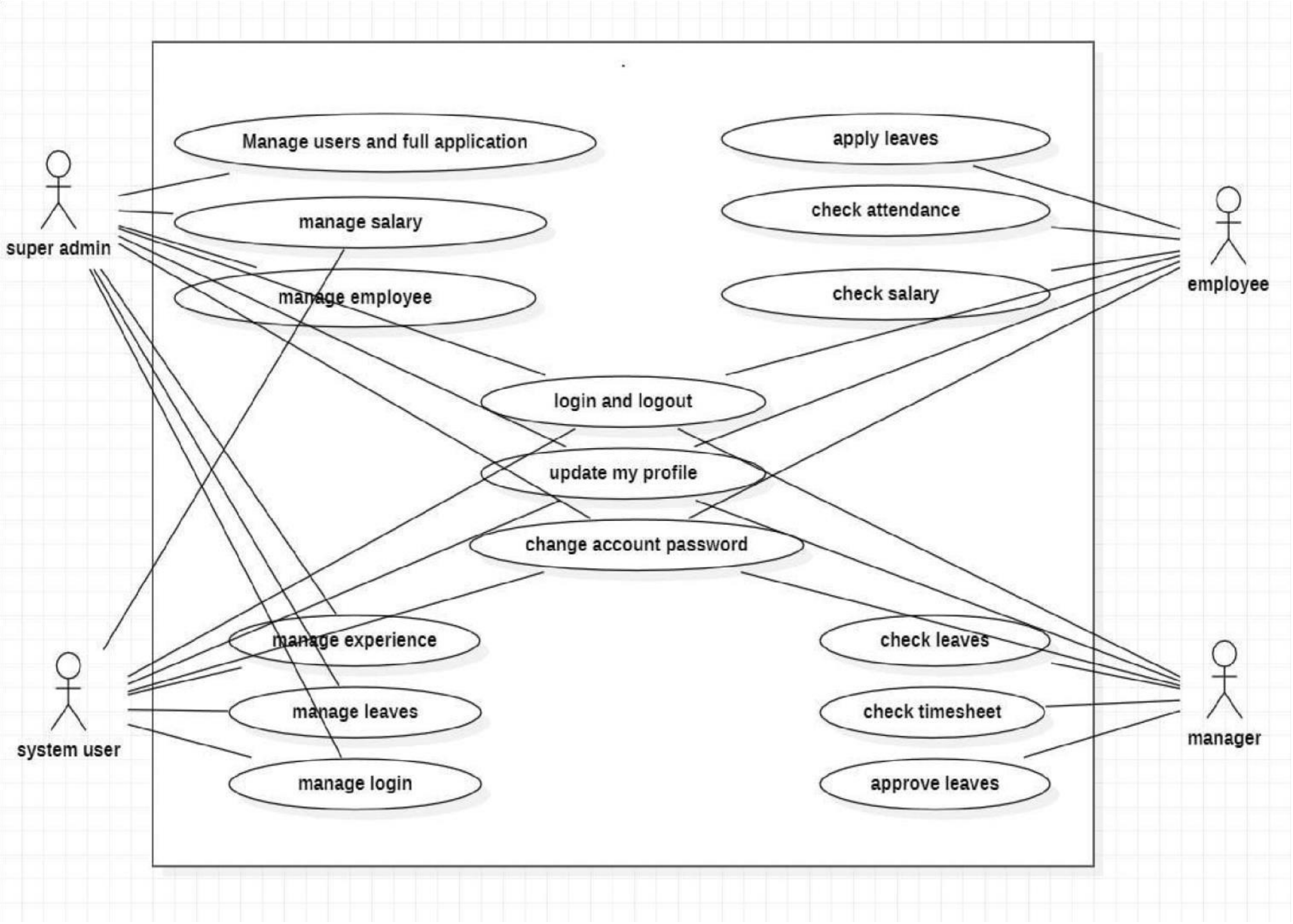
Classes and their attributes of Employee Management System Class Diagram:

- Employee Attributes employee_id, employee_name, employee_mobile, employee_email, employee_username, employee_password, employee_address
- Salary Attributes salary id, salary_employee_id, salary_amount, salary_total, salary_type, salary_description
- Leaves Attributes leave_id, leave employee_id, leave_type, leave_status, leave_to, leave_from, leave_description

Experience Attributes experience_id, experience_employee_id, experience_year, experience_type, experience_description

Login Attributes login_id, login_user_id, login_role_id, login_usemame, login_password, login_lastlogin • Attendance Attributes attendance_id, attendance_employee_id, attendance_student_id, attendance_type, attendance_description

Use case Diagram



This Use Case Diagram is a graphic depiction of the interactions among the elements of Employee Management System. It represents the methodol ogy used in system analysis to identify, clarify, and organize system requirements of Employee Management System. The main actors of Employee Management

System in this Use Case Diagram are: Super Admin, System User,

Employee, Manages, who perform the different type of use cases such as Manage Employee, Manage Salary, Manage Leaves, Manage Experience, Manage Login, Manage Users and Full Employee Management System Operations Major elements of the UML use case diagram of Employee Management System are shown on the picture above.

The relationships between and among the actors and the use cases of Employee Management System:

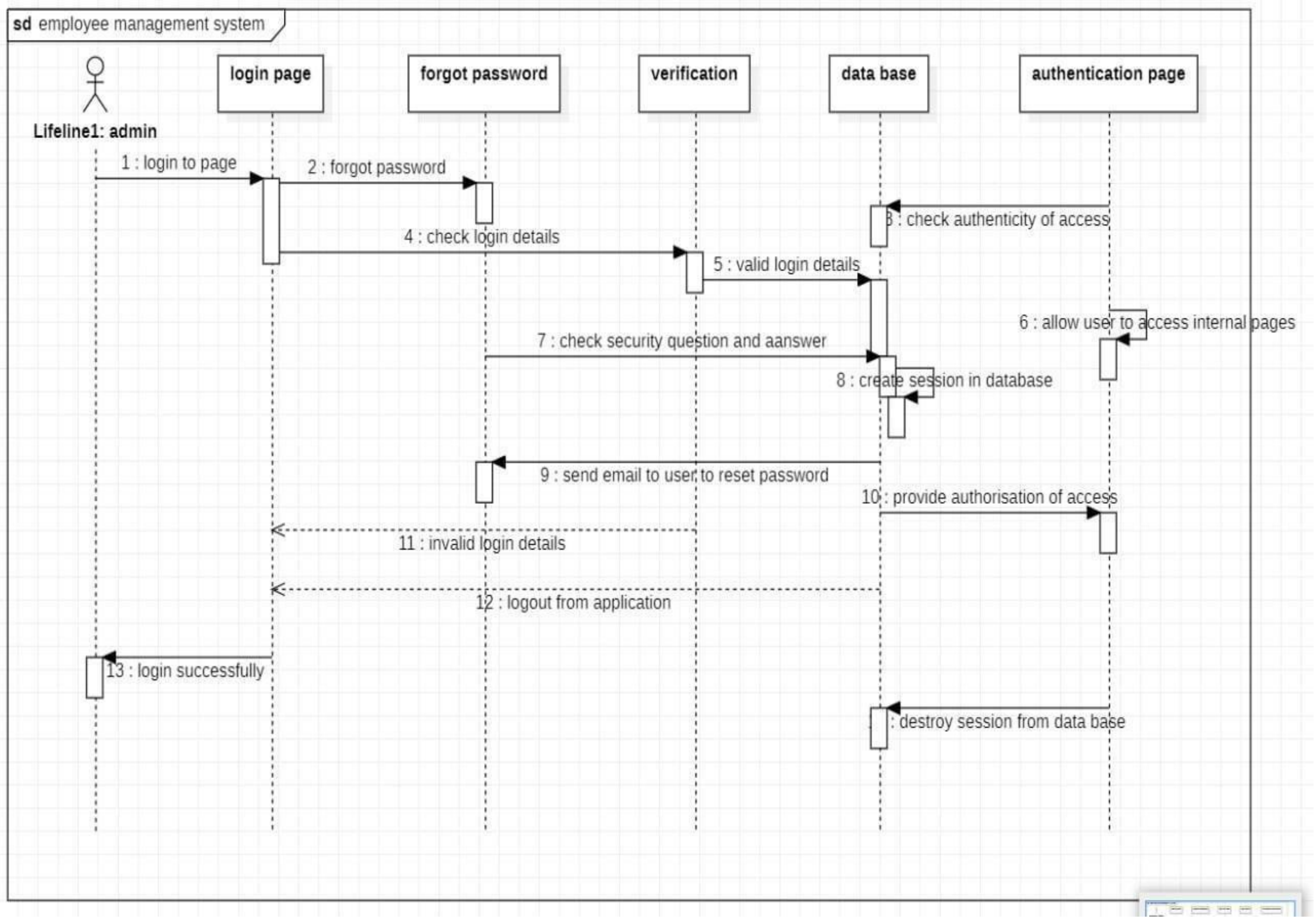
•Super Admin Entity: Use cases of Super Admin are Manage Employee, Manage Salary, Manage Leaves, Manage Experience, Manage Login, Manage Users and Full Employee Management System Operations

•System User Entity: Use cases of System User are Manage Employee, Manage Salary, Manage Leaves, Manage Experience, Manage Login

•Employee Entity: Use cases of Employee are Apply Leaves, Check Salary,
Check Attandance

•Manages Entity: Use cases of Manages are Check Leaves, Approve Leaves, Check Timesheet, Approve Timesheet

Sequence diagram :

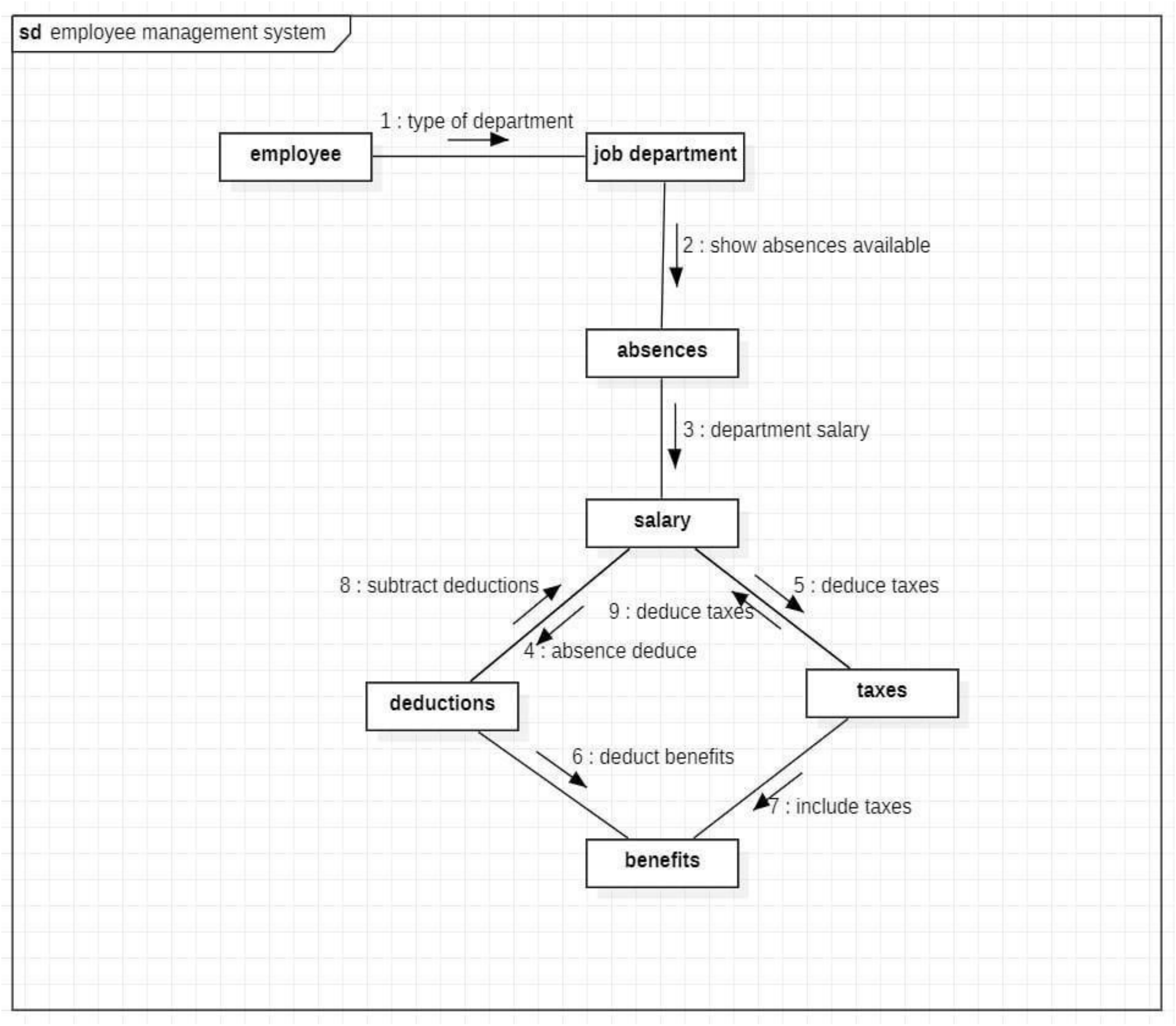

sd employee management system
Lifeline1: admin
login page
forgot password
verification
data base
authentication page
1 : login to page
2 : forgot password
3 : check authenticity of access
4 : check login details
5 : valid login details
6 : allow user to access internal pages
7 : check security question and aanswer
8 : create session in database
9 : send email to user to reset password
10 : provide authorisation of access
11 : invalid login details
12 : logout from application
13 : login successfully
: destroy session from data base

This is the Login Sequence Diagram of Employee Management System, where admin will be able to login in their account using their credentials. After login user can manage all the operations on Salary, Employee, Experience, Leaves, All the pages such as Experience, Leaves, are secure and user can access these page after login. The diagram below helps demonstrate how the login page works in a Employee Management System. The vari ous objects in the Leaves, Salary, Employee, Experience, and page-interact over the course of the sequence, and user will not be able to access this page without verifying their identity. All the pages such as Experience, Leaves, are secure and user can access these page after login. The diagram below helps demonstrate how the login page works in a Employee Management System.

This ER (Entity Relationship) Diagram represents the model of Employee Management System Entity. The entityrelationship diagram of Employee Management System shows all the visual instrument of database tables and the relations between Salary, Experience, Employee, Attendance etc.

It used structure data and to define the relationships between structured data groups of Employee Management System functionalities. The main entities of the Employee
Management System are Employee, Salary, Leaves, Experience, Login and Attendance. All the pages such as Experience, Leaves, are secure and user can access these page after login. The diagram below helps demonstrate how the login page works in a Employee Management System.

Collaboration diagram

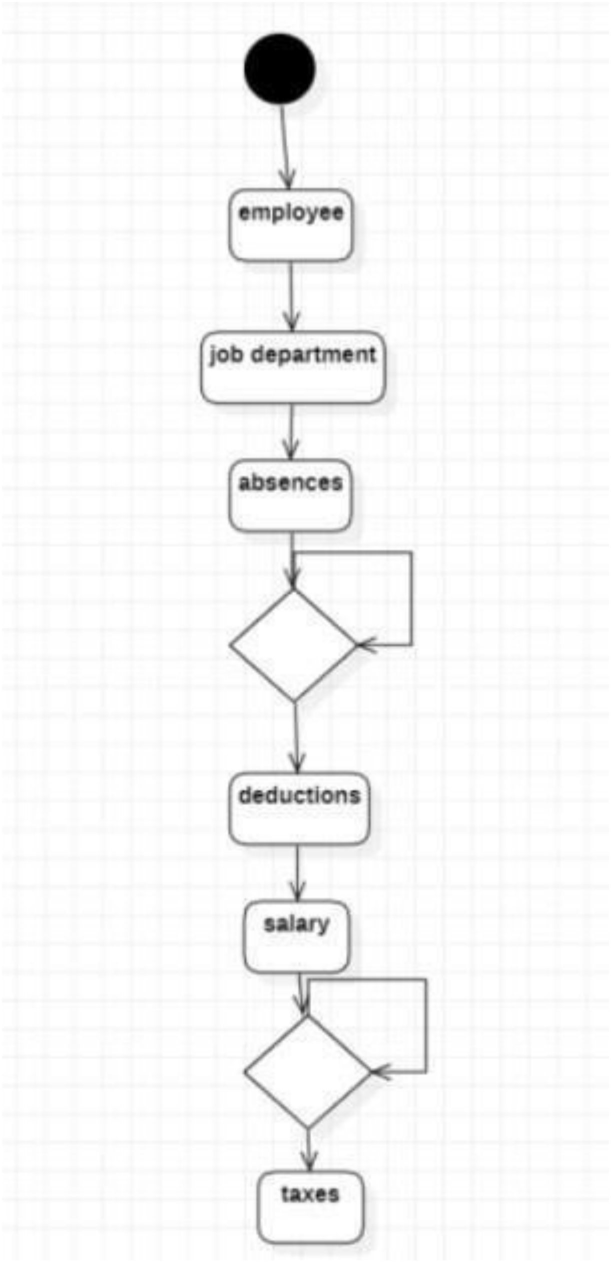


Employee management system for online workforce management in your organization. It does tasks such as keeping track of the workforce data to create. Thich is a very crucial task for payroll, employee experience, performance management and more. As with the exact work hour employee database you can do better workforce management for the firm.

Cloud-based employee management system with this software you can allow your staff and HR team to work remotely without any hurdles. The benefit of having software for staff management is an automated tracking of working hours. It is one of the advanced employee management systems that also identifies the location so you can online keep track of onfield employees effortlessly.

This also tends to save a lot of your HR efforts and time. As the portal allows your staff to service themselves and get their queries attended immediately. HR team do not need to spend time answering these questions. The application also makes it easier for human resources to get attendance and other accurate data

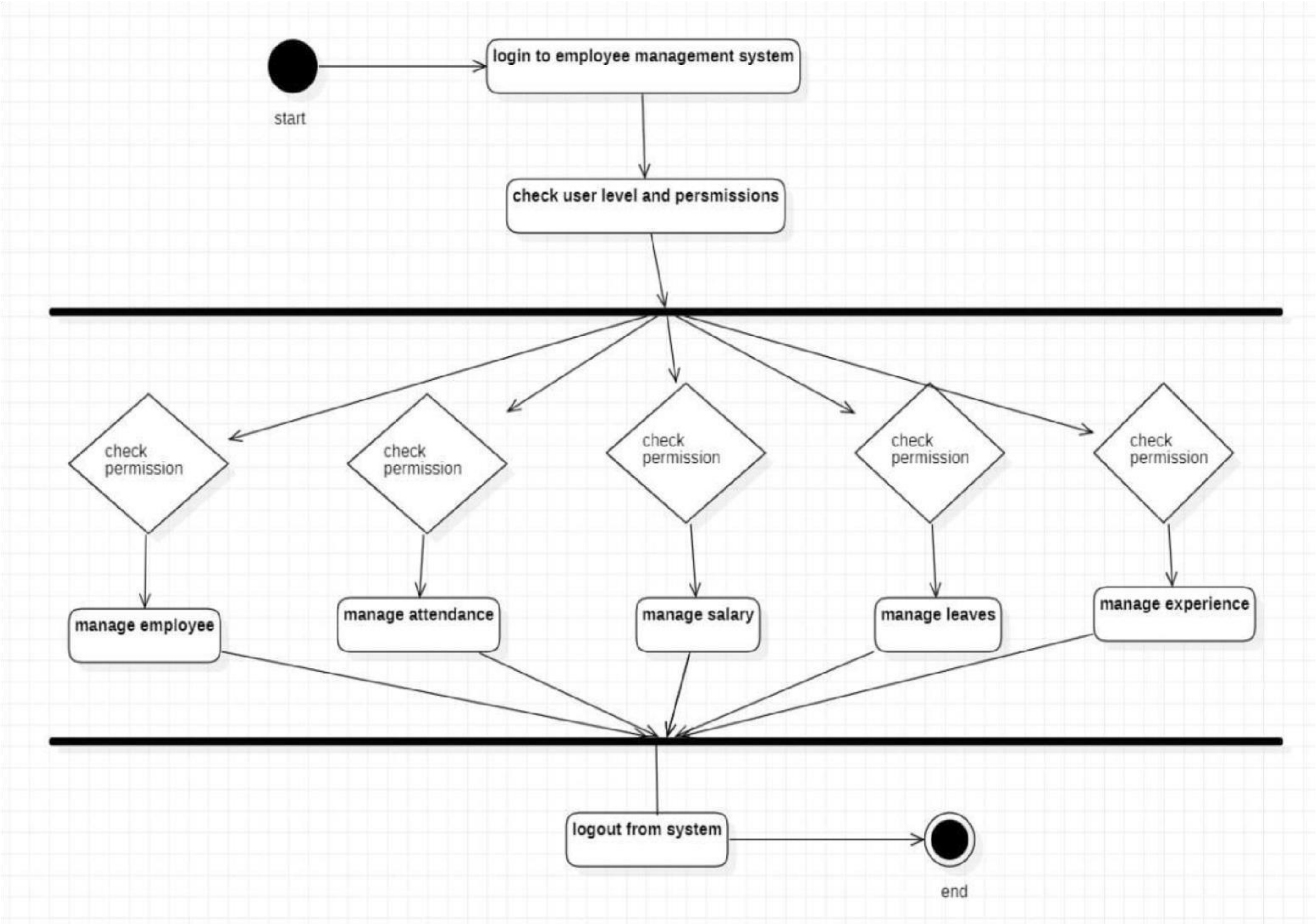
State chart diagram



The state chart for Employee Management System provides a broader overview of the context diagram. It widens the processes from the context diagram and determines the sub processes that complete the employee management system.

Cloud-based employee management system with this software you can allow your staff and HR team to work remotely without any hurdles. The benefit of having software for staff management is an automated tracking of working hours. It is one of the advanced employee management systems that also identifies the location so you can online keep track of onfield employees effortlessly.Leaves, All the pages such as Experience, Leaves, are secure and user can access these page after login. The diagram below helps demonstrate how the login page works in a Employee Management System. The vari ous objects in the Leaves, Salary, Employee, Experience, and page-interact over the course of the sequence, and user will not be able to access this page without verifying their identity.

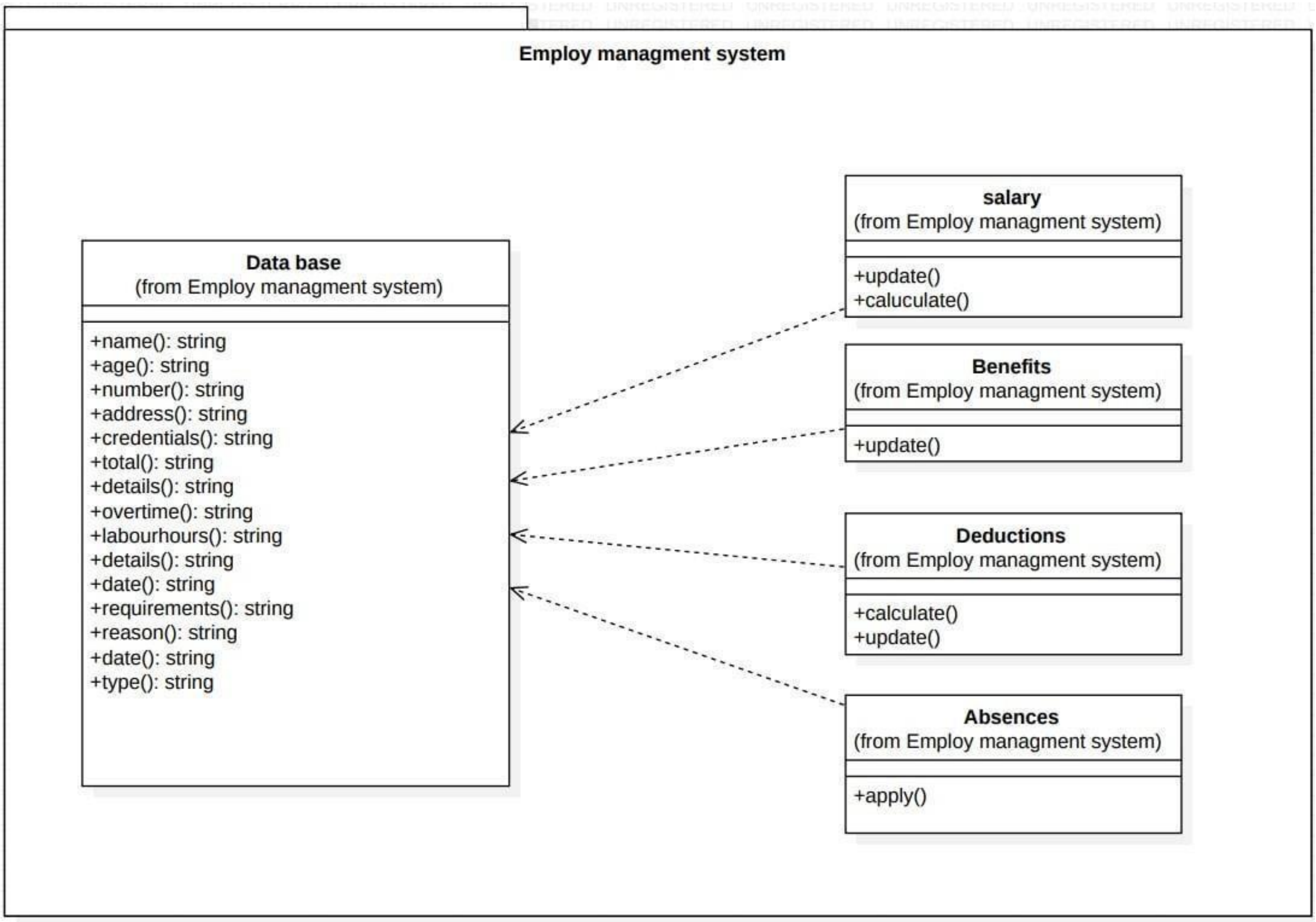
Activity diagram



Admin User can search Employee, view description of a selected Employee, add Employee, update Employee and delete Employee Its shows the activity flow of editing, adding and updating of Experience User will be able to search and generate report of, Salary, Leaves All objects such as (Employee, Experience, Leaves) are interlinked Its shows the full description and flow of Employee, Salary, Leaves, Experience Its shows the activity flow of editing, adding and updating of Experience User will be able to search and generate report of, Salary, Leaves.

Leaves, All the pages such as Experience, Leaves, are secure and user can access these page after login. The diagram below helps demonstrate how the login page works in a Employee Management System. The vari ous objects in the Leaves, Salary, Employee, Experience, and page-interact over the course of the sequence, and user will not be able to access this page without verifying their identity.

Package diagram

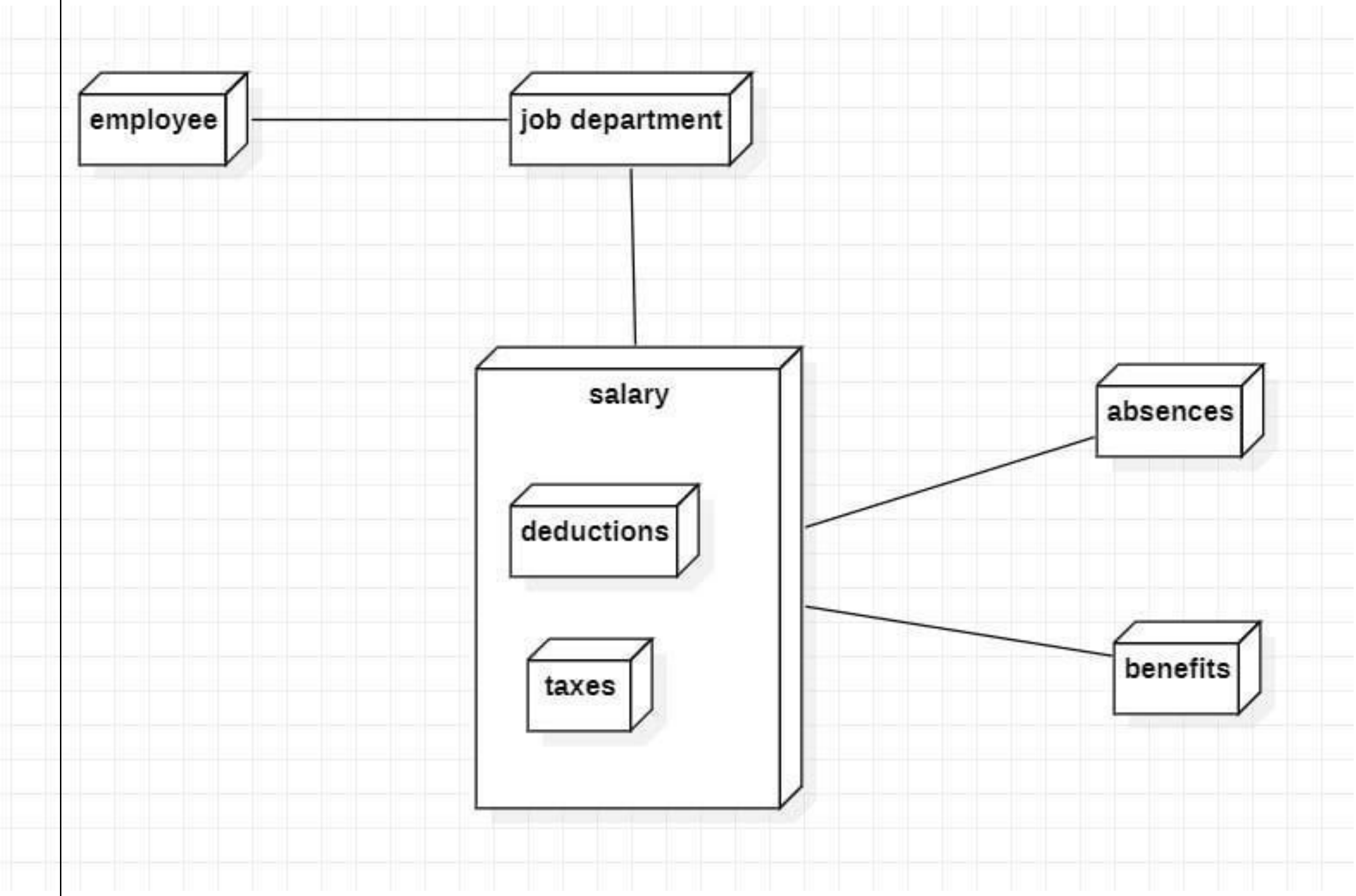


In this world of growing technologies everything has been computerized. With large number of work opportunities the Human workforce has increased. Thus there is a need of a system which can handle the data of such a large number of Employees in an organization. This project simplifies the task of maintain records because of its user friendly nature.

The main actors of Employee Management System in this Use
Case Diagram are: Super Admin, System User, Employee, Manages, who perform the different type of use cases such as
Manage Employee, Manage Salary, Manage Leaves, Manage
Experience, Manage Login, Manage Users and Full Employee Management System Operations Major elements of the UML use case diagram of Employee

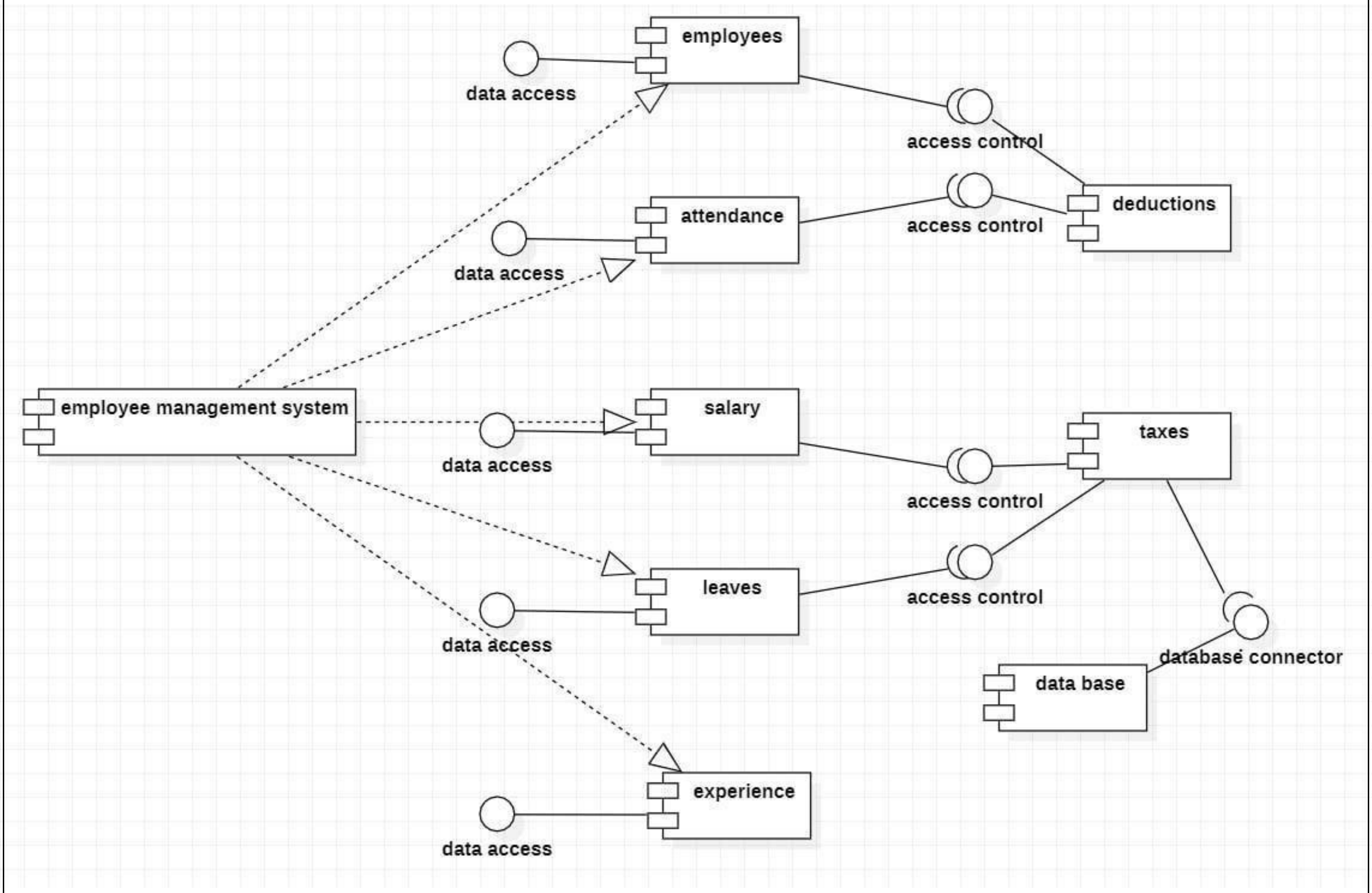
To calculate the payroll correctly it is also important to have access to the employee database easily. This is where the best employee management system, Pocket HRMS comes to your rescue. With our cloud-based employee salary management software you can also automate accurate payroll. This is to help you build employee trust. You can also keep legal troubles away with Employee management systems.

Deployment diagram :



The deployment diagram for the employee management system is used to represent the system's physical architecture showing the included nodes and their associations. UML deployment diagram nodes are composed of hardware and software components that serve as primary requirements to carry out the employee management.

Component diagram :



This is a Component diagram of Employee Management System which shows components, provided and required interfaces, ports, and relationships between the Salary, Employee, Leaves and Experience. This type of diagram is used in Component-Based Development (CBD) to describe systems with Service-Oriented Architecture (SOA). Employee Management System UML component diagram, describes the organization and wiring of the physical components in a system. Components of UML Component Diagram of Employee Management System.

To calculate the payroll correctly it is also important to have access to the employee database easily. This is where the best employee management system, Pocket HRMS comes to your rescue. With our cloud-based employee salary management software you can also automate accurate payroll. This is to help you build employee trust. You can also keep legal troubles away
with Employee management systems.

Conclusion :

By completing all the given Diagrams, the Employee Management Project System development would be much easier and attainable. So those UML diagrams were given to teach you and guide you through your project development journey. You can use all of the given UML diagrams as your reference, or have them for your project development. The ideas presented in UML Diagrams were all based on Employee Management System requirements.

References :

www.youtube.com
https://docs.staruml.io/workingwithumldiagrams/classdiagram