January 2022-June 2022

OBJECT ORIENTED DESIGN LAB CS-315

B.Tech (CSE) Sixth Semester

Department of Computer Science and Engineering

Under the Guidance of:

Dr. Malaya Dutta Borah

Assistant Professor

Submitted By:

Sourabh Shah

1912158

LIST OF EXPERIMENTS

Experiment Number	Experiment Name	Date
1	TO DRAW THE UML DIAGRAM FOR A CONFERENCE MANAGEMENT SYSTEM	25/01/2022
2	TO DRAW THE UML DIAGRAM FOR A DELIVERY AGENT SYSTEM	01/02/2022
3	TO DRAW THE UML DIAGRAM FOR AN ONLINE CAMPUS SECURITY MANAGEMENT SYSTEM	01/02/2022
4	TO DRAW THE UML DIAGRAM FOR A HOSPITAL MANAGEMENT SYSTEM	08/02/2022
5	TO DRAW THE UML DIAGRAM FOR A PAYROLL MANAGEMENT SYSTEM	15/02/2022
6	TO DRAW THE UML DIAGRAM FOR A SMART HOME MANAGEMENT SYSTEM	22/02/2022
7	TO DRAW THE UML DIAGRAM FOR AN ONLINE EXAMINATION SYSTEM	01/03/2022
8	TO DRAW THE UML DIAGRAM FOR AN ONLINE TOLL PLAZA SYSTEM	08/03/2022
9	TO DRAW THE UML DIAGRAM FOR AN ONLINE HEALTH MONITORING SYSTEM	31/03/3033

01: TO DRAW THE UML DIAGRAM FOR A CONFERENCE MANAGEMENT SYSTEM

THEORY:

Scientific conferences publish the latest innovations and outcomes in research work spanning several domains. However, before a research paper is published – or even selected for publication – it must go through a rigorous peer-review process. A Conference Management System (CMS) is software that helps to manage and streamline such review processes. EDAS is an example of a CMS.

Organizing a conference is a huge task, and the responsibilities are shared by many people divided into different committees. In general, there are two large committees – the program committee (PC) and the organizing committee (OC). The PC consists of several people (both from academia and industry) who are experts in their respective areas. The role of PC members includes pre-screening the submissions to ensure that appropriate qualities of papers are considered. Subsequently, the PC members assign reviewers for each paper under consideration. Any given paper must have at least two reviewers assigned. However, if a reviewer declines to review a particular paper, new reviewer(s) must be assigned for it. Once all the reviews for all the papers are available, the PC members select a fraction of them for publication. The PC chair(s) oversee the entire review phase and help in tie-breaking if required.

The OC has 1-2 general chair(s) who oversee the entire conference organization process. The OC consists of smaller subcommittees with 1-5 members such as publication chair(s), publicity chair(s), registration chair(s), web chair(s), and so on.

Once a conference has been planned, the OC chair(s) has to request the CMS to create an entry for the corresponding conference. The CMS provides a special link to make such requests online. After an entry for the conference is created, the OC chairs are given admin privileges, who, in turn, are responsible for adding the other relevant members. The URL linking the conference to the CMS is advertised. Authors access that URL, and upload their papers along with all other relevant inputs. After a pre-published deadline, the decision of a paper (accept or reject) together with reviewers 39; comments are visible to the authors. In case a paper is accepted, the authors should make changes as per the review comments, if relevant, and upload the camera-ready version within a specified deadline.

INPUTS:

Conference information (title, date, venue, topics covered, and deadlines)
OC &amp; PC information (email addresses of the members)
User information (name, password, email address, affiliation, and areas of expertise [optional])
Paper information (title, abstract, authors, affiliations, and PDF file)
Review comments:

- o Rate the paper on a scale of 1-5
- o Detailed comments
- o Reviewer 39 opinion on whether to accept/reject the paper

Final selection decision (paper id, accept/reject)

OUTPUTS:

Details of papers submitted
Review decision (see the format of review comments)
List of papers accepted
Percentage of papers accepted

CONSTRAINTS:

All users must register with the CMS to access the portal
A reviewer assigned for a paper may not be an existing user of the CMS; he/she must register in order to submit the review comments
Login information consists of email address and
password Each submitted paper is assigned a unique ID
It is sufficient if only one author of a paper is registered to the CMS; he/she will be responsible for submitting the paper
An author can view the review decision for his/her paper(s) only
A reviewer can view information only about those papers for which he/she has been assigned a reviewer
Only PC members can view all reviews for all papers

EXPLANATION:

Assigner, OC (organising committee), Author, Reviewer, and External Reviewer are the five actors in this scenario. The document is assigned by the assigner. The organising committee is in responsible of making decisions, collecting papers, submitting final submissions, and doing final reviews, among other things. The author next submits his or her article in accordance with the guidelines provided by the organising committee. The author can still make revisions to their submitted article if the submission hasn't been closed yet. After that, the Reviewer evaluates the paper, approves or rejects it, and assigns a rating to it. The paper can also be rated and reviewed by an external review.

USECASE DIAGRAM:

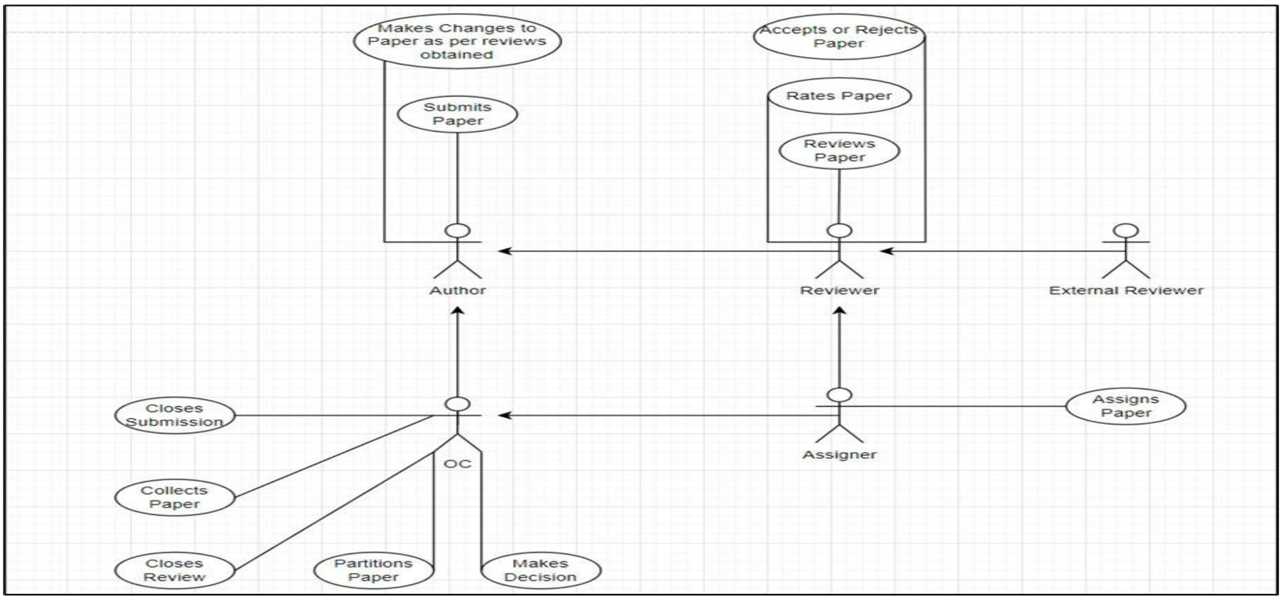


Fig 1.1: Use-Case Diagram for a Conference Management System

CLASS DIAGRAM:

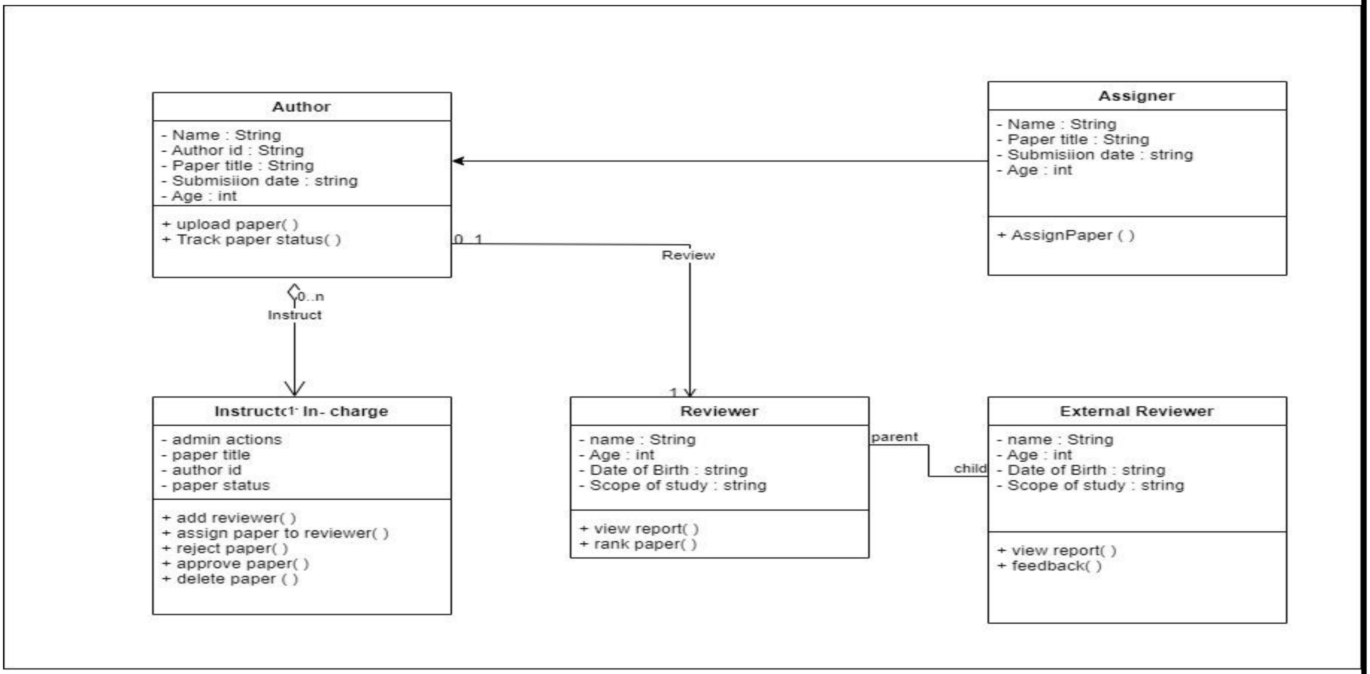


Fig 1.2: Class Diagram for a Conference Management System

SEQUENCE DIAGRAM:

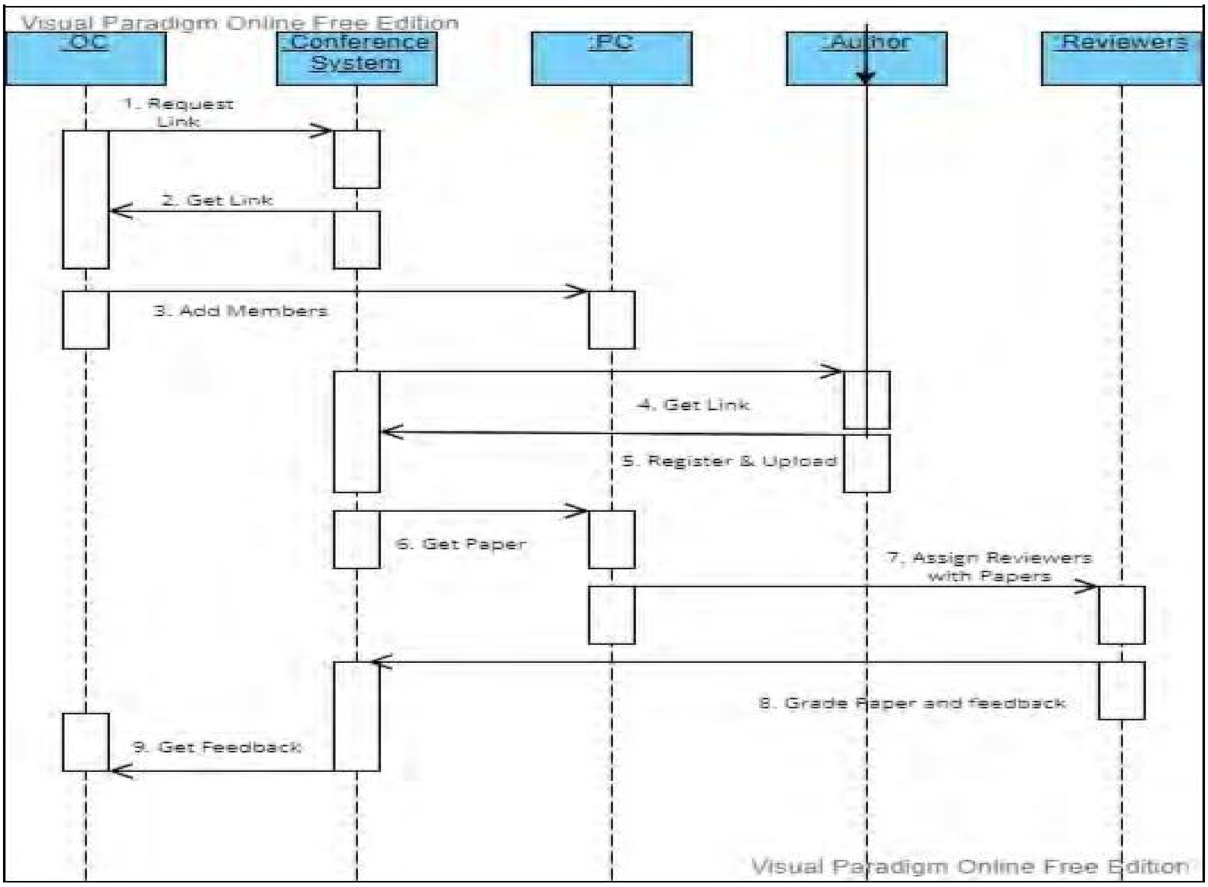


Fig 1.3: Sequence Diagram for a Conference Management System

STATE DIAGRAM:

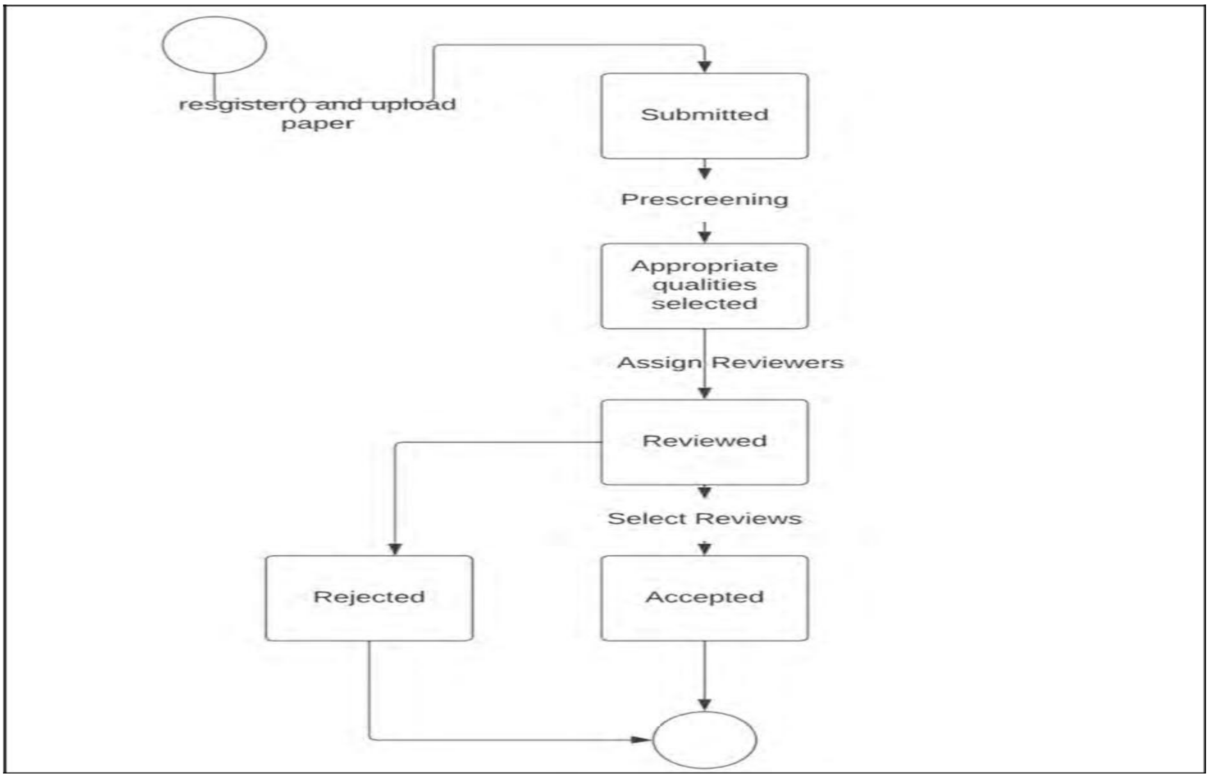


Fig 1.4: State Diagram for a Conference Management System

COLLABORATION DIAGRAM:

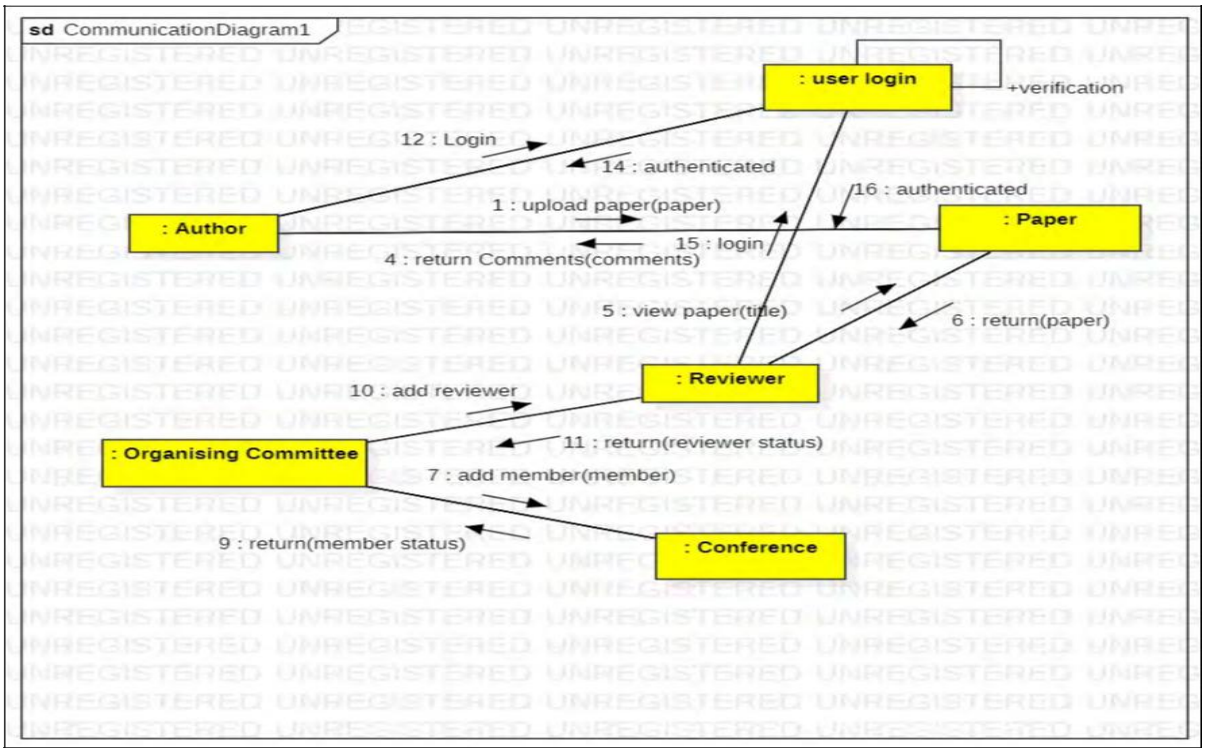


Fig 1.5: Collaboration Diagram for a Conference Management System

PACKAGE DIAGRAM:

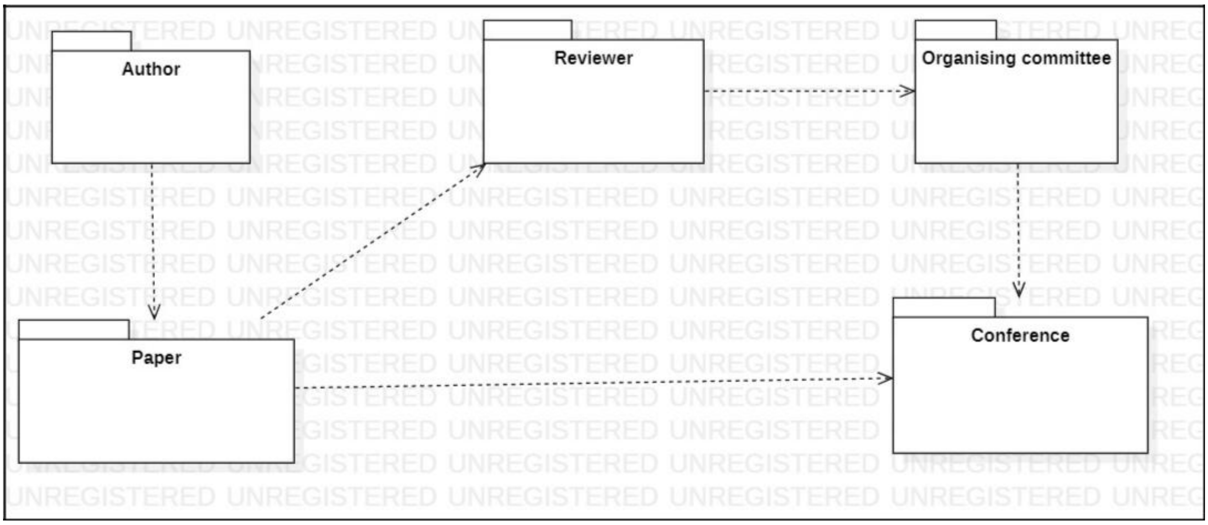


Fig 1.6: Package Diagram for a Conference Management System

COMPONENT DIAGRAM:

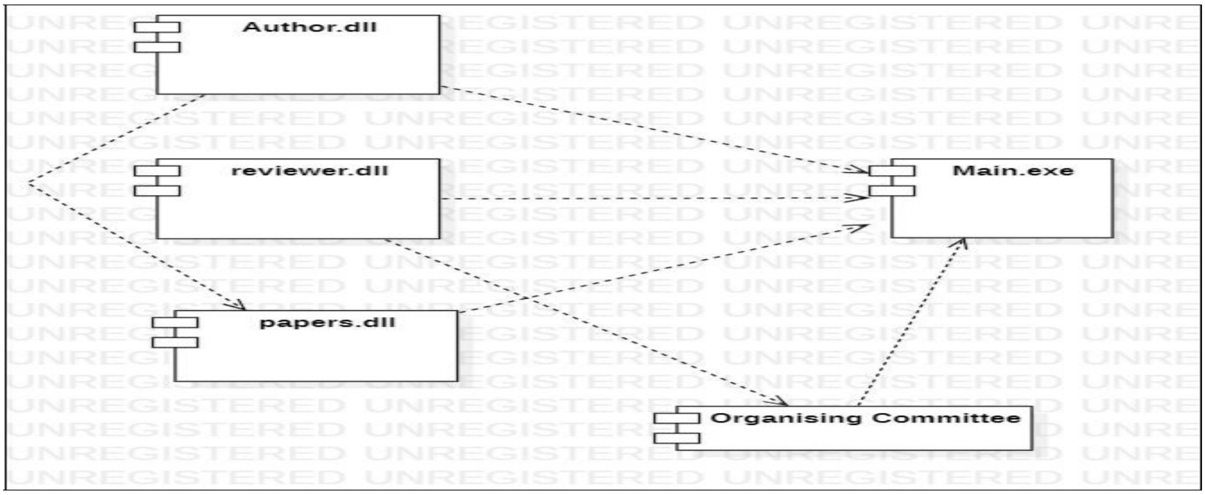


Fig 1.7: Component Diagram for a Conference Management System

DEPLOYMENT DIAGRAM:

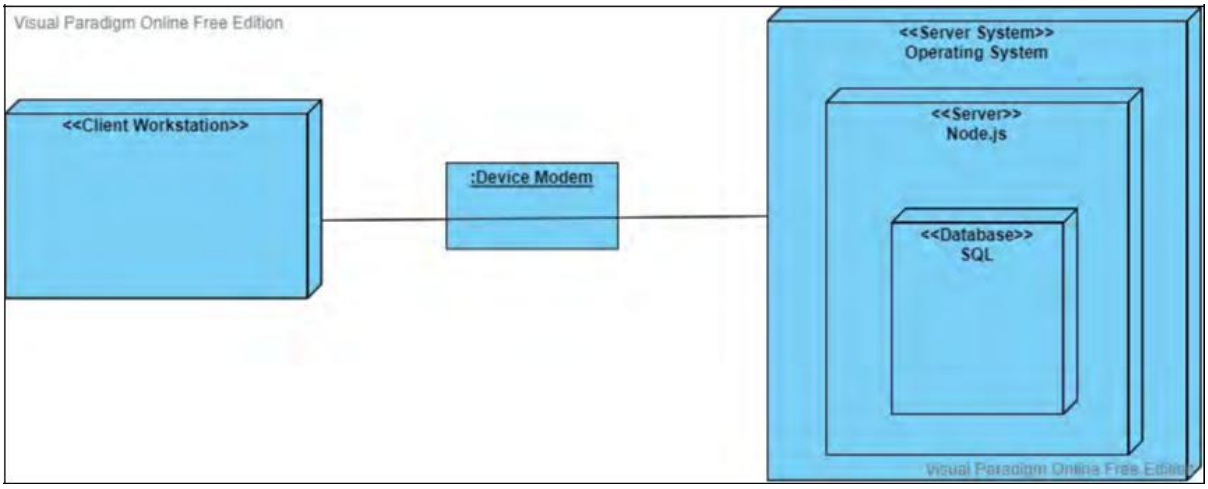


Fig 1.8: Deployment Diagram for a Conference Management System

02: TO DRAW THE UML FOR A DELIVERY AGENT SYSTEM

THEORY:

There are many online shopping portals such as Flipkart, Amazon, Snapdeal, etc. are active in the Indian market. One major task is to deliver online books products to the customers as first as possible in a cost-effective (cheapest) manner. A delivery agent system, which would automatically receive a delivery request from an online portal and identify the couriers, whom the delivery job can be assigned deadline.

INPUTS:

Shipping details (source and destination) locations
Couriers' details in different localities.
Service offering for each courier company.

OUTPUTS:

Booking delivery
Status of delivery
Cancellation of booking
Delivery rescheduling
Reward and penalty calculation
Archiving record of past services

CONSTRAINTS:

Booking confirmation, if booking is successful.
Reporting delivery status
Cancellation of booking confirmation
Update record on delivery rescheduling
Update record and intimation on reward and penalty calculation
Query generation on record of past service, given a courier agency.

EXPLANATION:

Because this is a Delivery Agent System, we have two actors: Customer and Delivery Agent. I've incorporated an online system called "System" that may be either an actor (Admin) or an automated system because it's online (System). The consumer will log in and place an order for delivery. He can also make changes to his delivery or postpone it, thus the Delivery Booking is expanded to reflect these restrictions. The consumer will proceed to payment details, which will contain his shipping address, when the delivery has been scheduled. Once the delivery has been scheduled, the consumer receives a receipt for his transaction or purchase, as well as the ability to monitor his goods. The system will include information about the establishments as well as information about delivery agents in the area. So,When a delivery is required, the system will notify the delivery agent. The delivery agent will collect the package and notify you of the status. When the delivery agent has finished his delivery, he will update the status once more, and the client will be rewarded. The complete order and delivery information will be preserved and saved in the system for future use.

USECASE DIAGRAM:

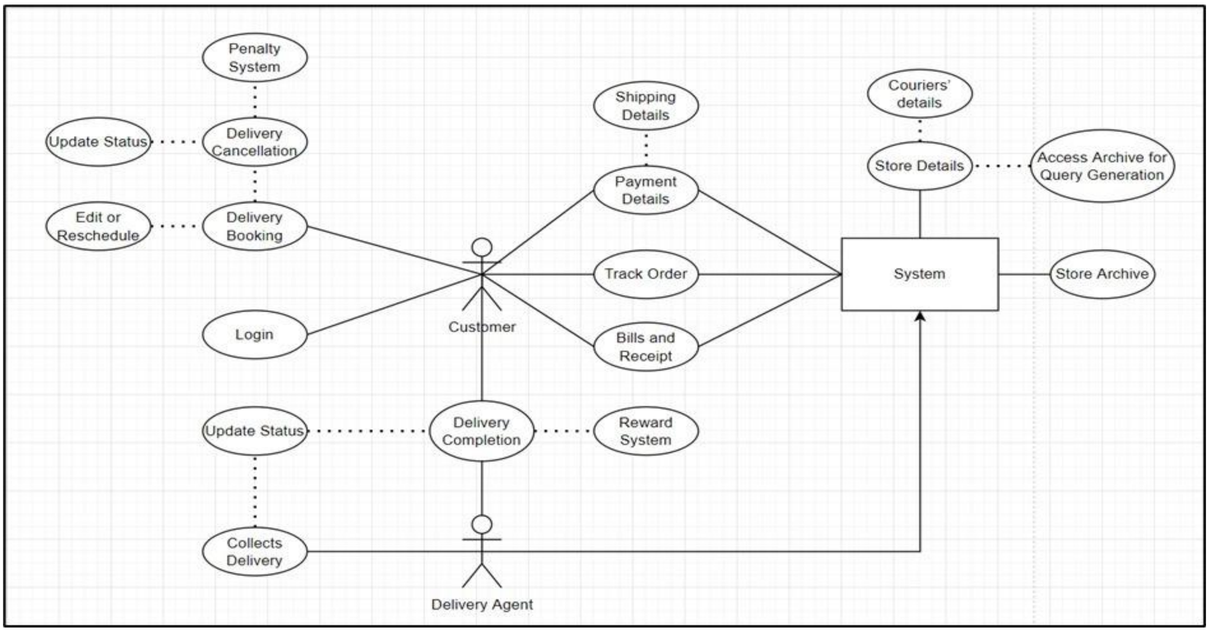


Fig 2.1: UseCase Diagram for a Delivery Agent System

CLASS DIAGRAM:

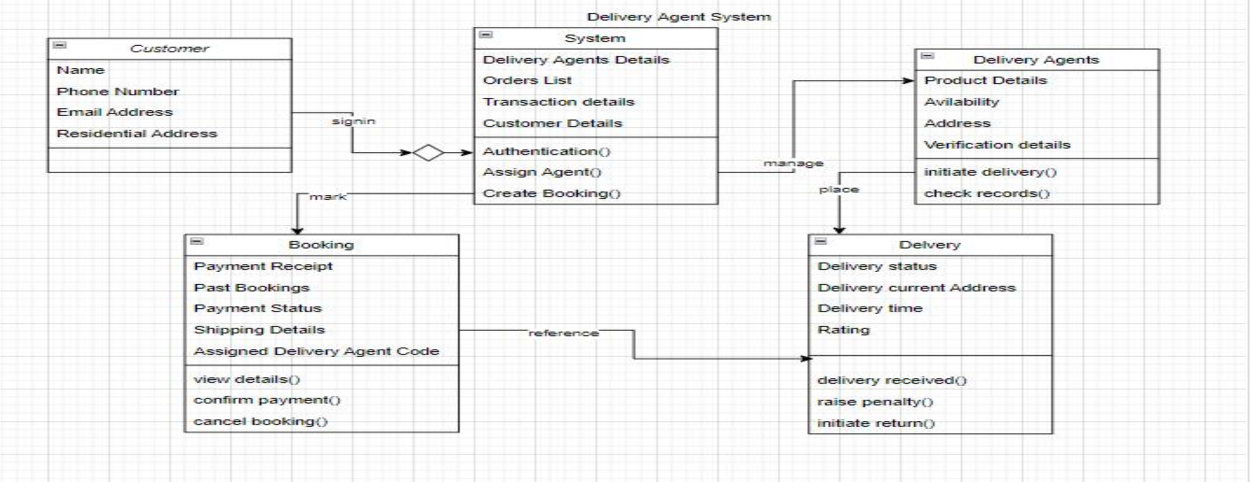


Fig 2.2: Class Diagram for a Delivery Agent System

SEQUENCE DIAGRAM:

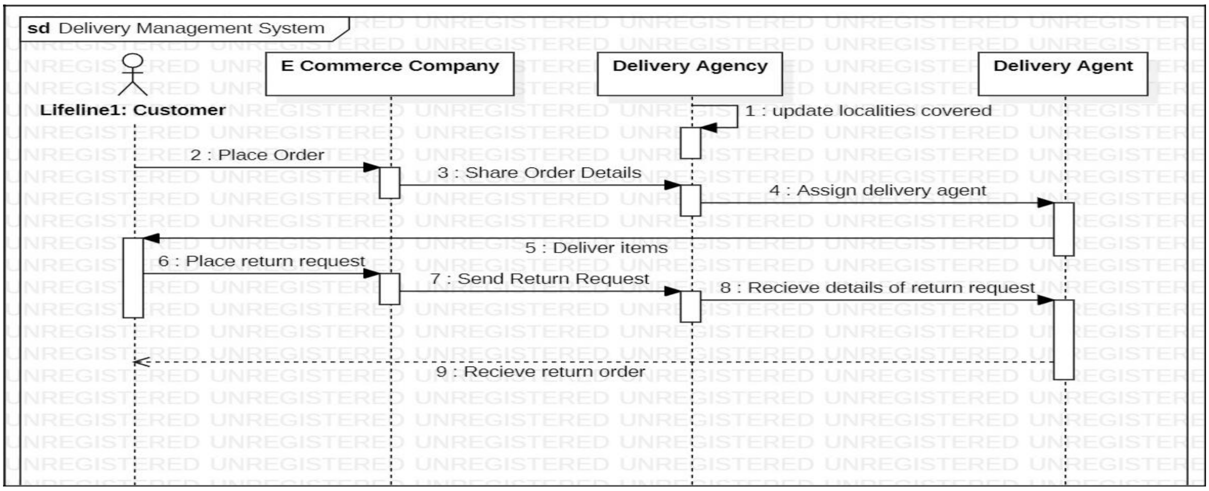
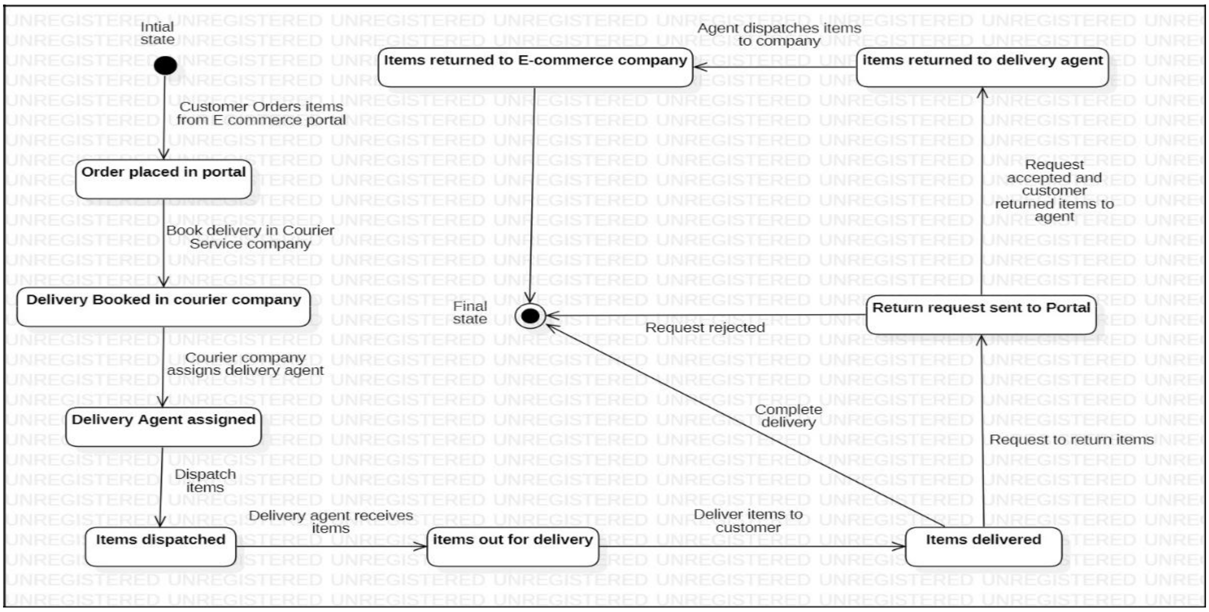


Fig 2.3: Sequence Diagram for a Delivery Agent System

STATE DIAGRAM:



Intial state
Customer Orders items from E commerce portal
Order placed in portal
Book delivery in Courier Service company
Delivery Booked in courier company
Courier company assigns delivery agent
Delivery Agent assigned
Dispatch items
Items dispatched
Delivery agent receives items
items out for delivery
Deliver items to customer
Items delivered
Complete delivery
Final state
Request rejected
Request to return items
Return request sent to Portal
Request accepted and customer returned items to agent
items returned to delivery agent
Agent dispatches items to company
Items returned to E-commerce company

Fig 2.4: State Diagram for a Delivery Agent System

COLLABORATION DIAGRAM:

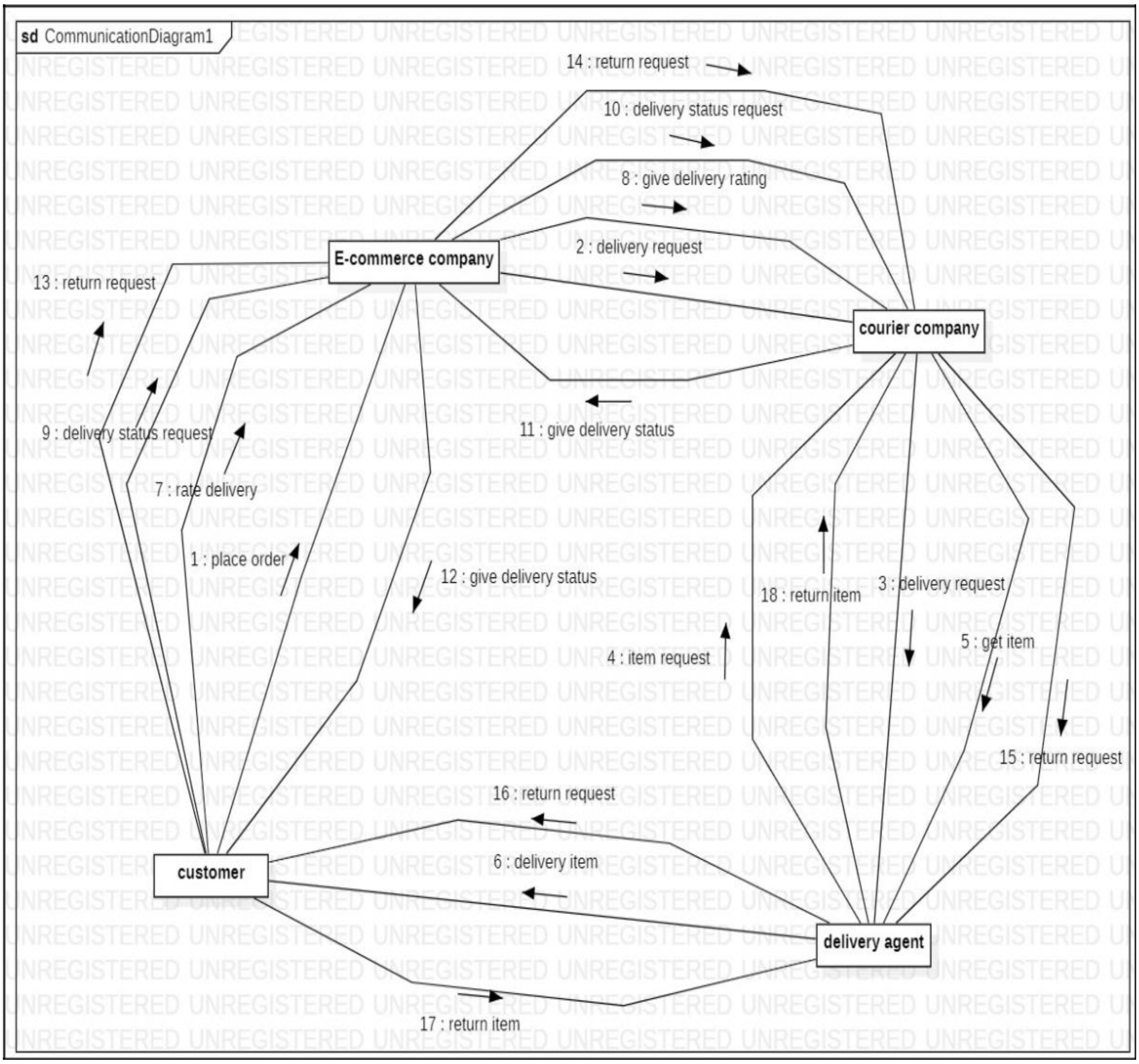


Fig 2.5: Collaboration Diagram for a Delivery Agent System

PACKAGE DIAGRAM:

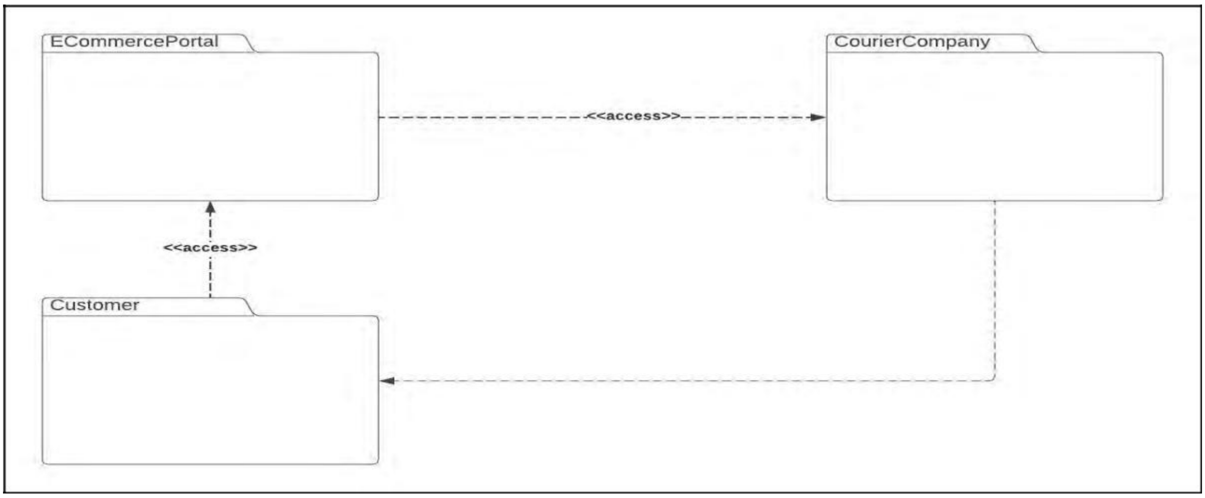


Fig 2.6: Package Diagram for a Delivery Agent System

COMPONENT DIAGRAM:

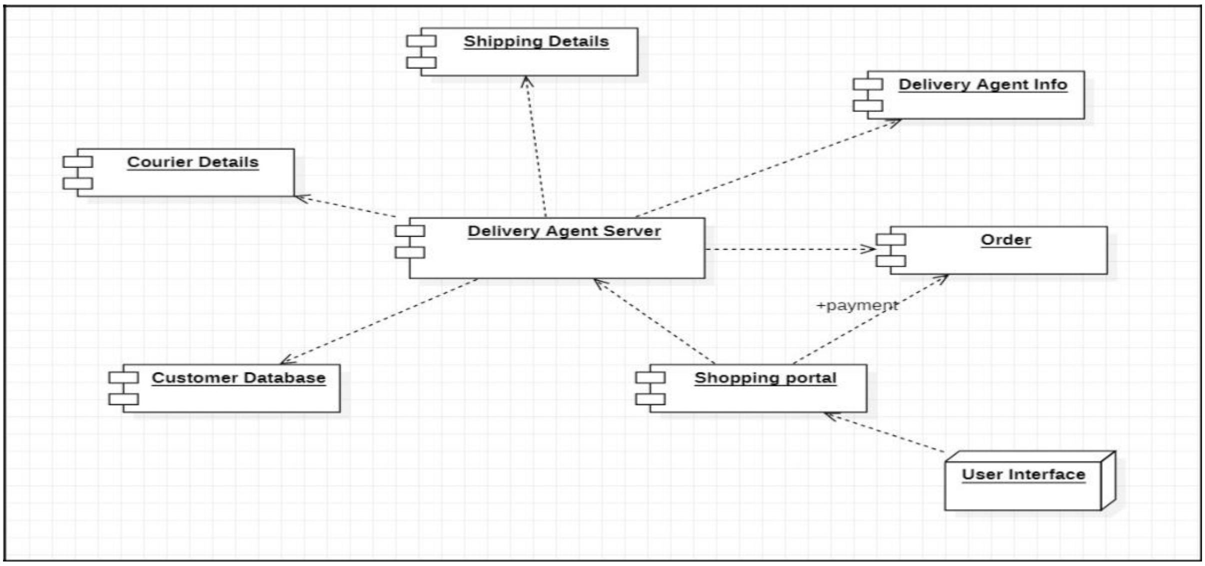


Fig 2.7: Component Diagram for a Delivery Agent System

DEPLOYMENT DIAGRAM:

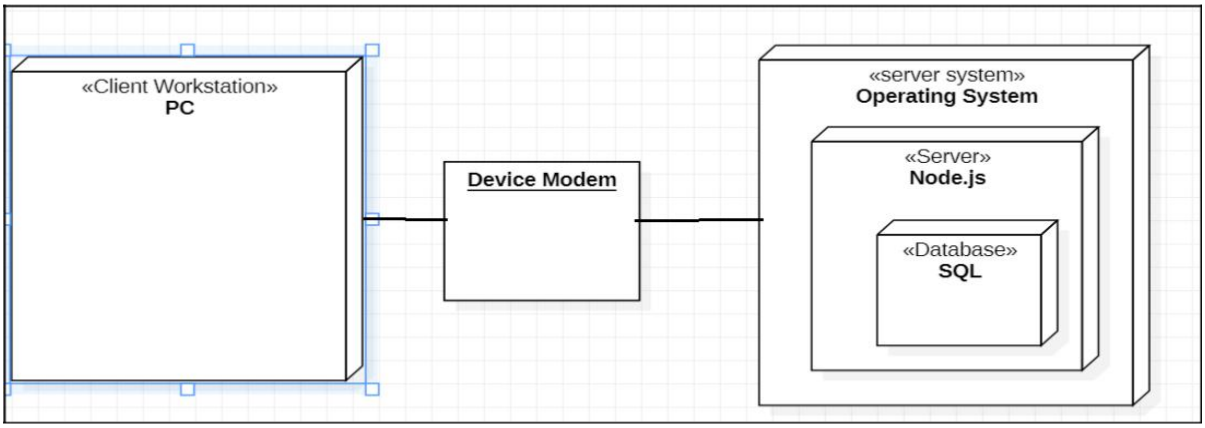


Fig 2.2: Deployment Diagram for a Delivery Agent System

03: TO DRAW THE UML FOR AN ONLINE CAMPUS SECURITY MANAGEMENT SYSTEM

THEORY:

XYZ Company has more than thousand security persons, who are instructed to give duties at different places within the campus. Additionally, they also maintain a routine, which contains all information, such as Date, Duty Start Time, Duty End Time, and Place. Most importantly, all the places are covered by at least one security person. If a security person takes leave, manual entry is done against that person. Finally, at the end of a month, the security persons get paid for their duties, while considering the number of leaves as well. You can see that the manual calculation/operation is a heavy task for the security manager. Therefore, the objective is to build an Online security management system through which entire security system within the campus can be controlled in an efficient manner.

INPUTS:

User Information

- Security – (Name, Identity Number, Password)
- Total number of security persons
- Manager – (Name, Identity, Password)

Place Information

- Number of places identified by unique numbers

OPERATIONS:

Security Person

- Log-In
- View duty date, place, start time, end time (upcoming 7 days schedule can be viewed)
- Request manager to take leave or to do over duty
- Request approved/declined
- Number of leaves taken/ number of allowed leaves remaining
- Log-Out

Manager

- Log-In
- Create routine for upcoming 7 days for all persons considering leave requests
- Approve/decline leave request
- Monitoring
- Log-Out

OUTPUTS:

Salary at the end of the month

View routine

CONSTRAINTS:

All users MUST register themselves into the system.

A security person can only check his/her own routine.

Manager can check the status of all security persons.

A fixed number of leaves are allowed. Beyond that, fixed amount will be deducted as fine.

All security persons get same number of duties in a month.

Adequate number of security persons are there to cover all places considering leave requests. A security person must not have duties in two different places at the same date and time.

EXPLANATION:

There are two characters in this scene: a Security Person and a Manager. The security person's schedule and remuneration are managed by the manager. A security guard has access to his routine, which contains information about his responsibilities. In the event of a special occasion or an emergency, security personnel might also seek time off. All of this is represented in the View Leave section, and the salary is adjusted as necessary.

USECASE DIAGRAM:

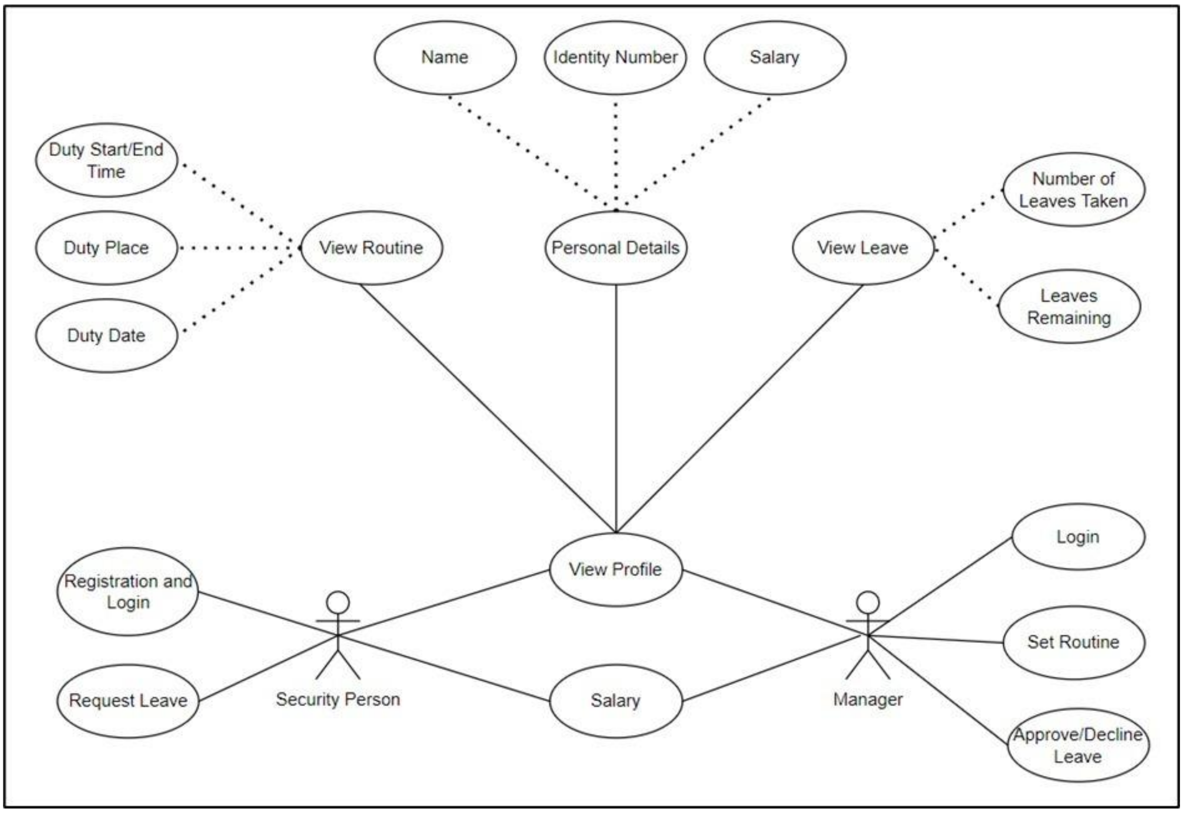


Fig 3.1: USECase Diagram for an Online Campus Security Management System

CLASS DIAGRAM:

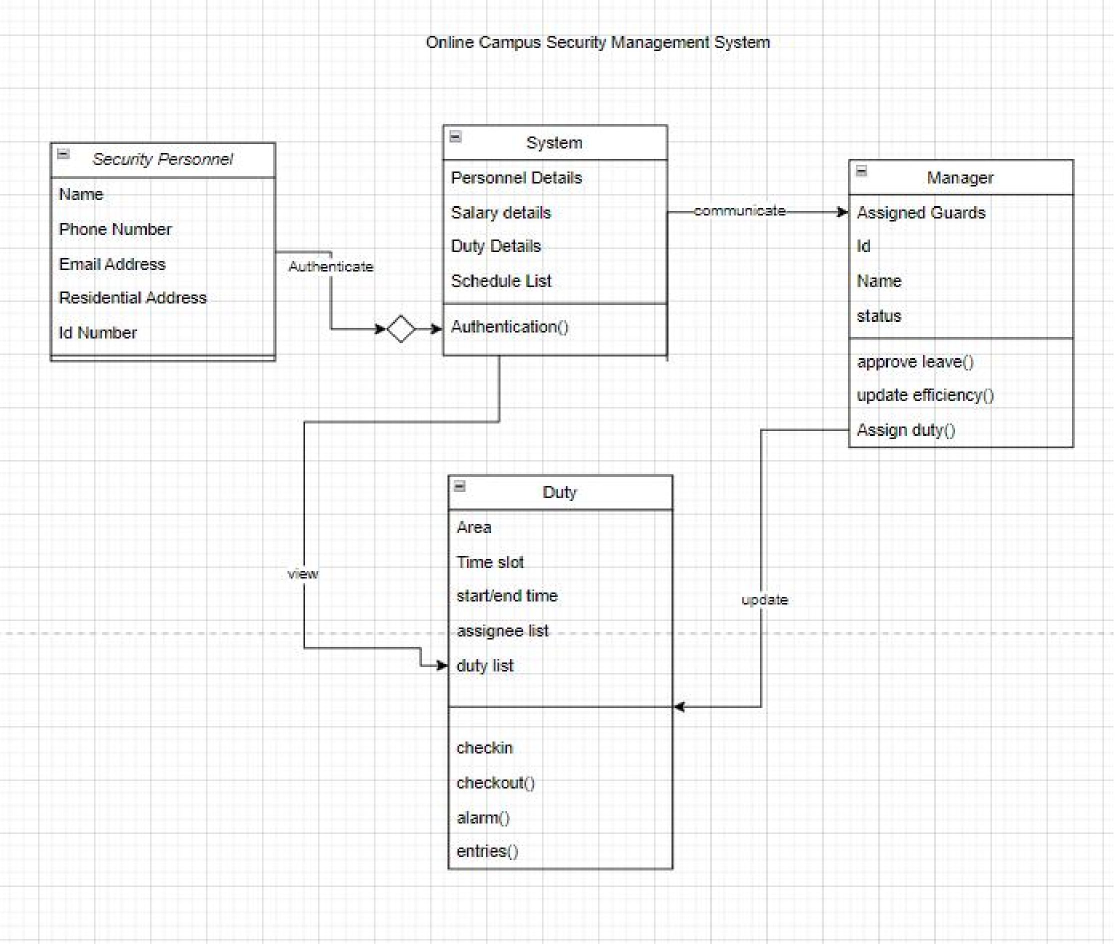


Fig 3.2: Class Diagram for an Online Campus Security Management System

SEQUENCE DIAGRAM:

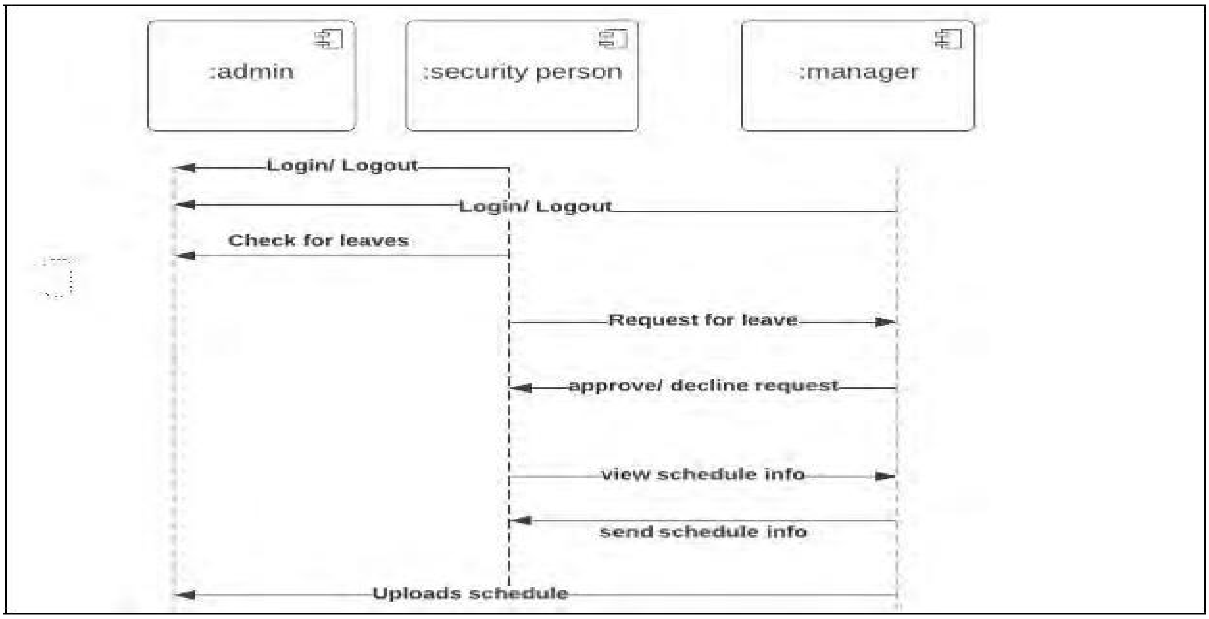


Fig 3.3: Sequence Diagram for an Online Campus Security Management System

STATE DIAGRAM:

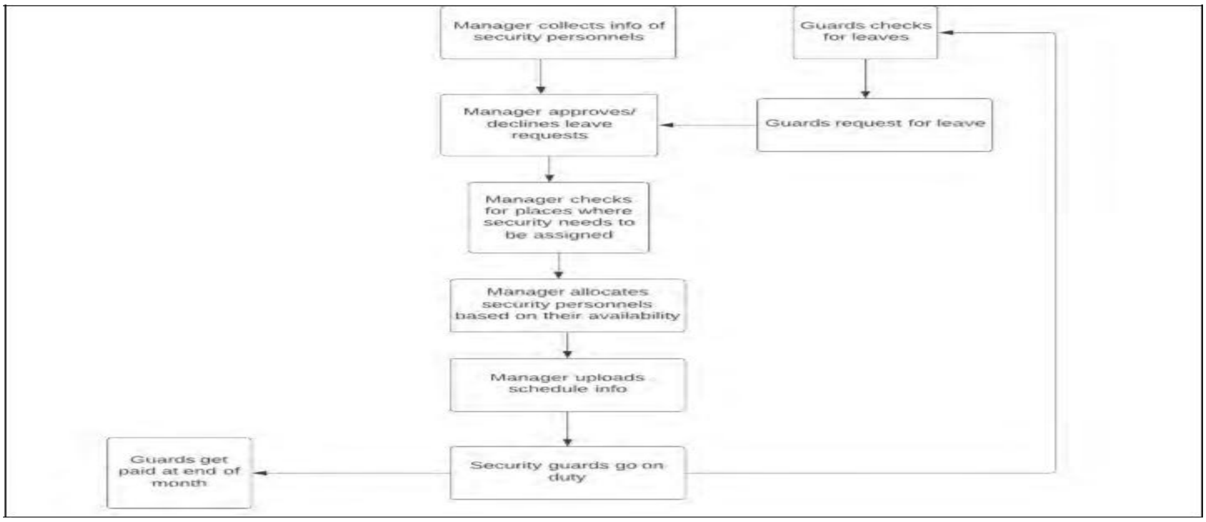

Manager collects info of security personnels
Guards checks for leaves
Manager approves/ declines leave requests
Guards request for leave
Manager checks for places where security needs to be assigned
Manager allocates security personnels based on their availability
Manager uploads schedule info
Guards get paid at end of month
Security guards go on duty

Fig 3.4: State Diagram for a Campus Security Management System

COLLABORATION DIAGRAM:

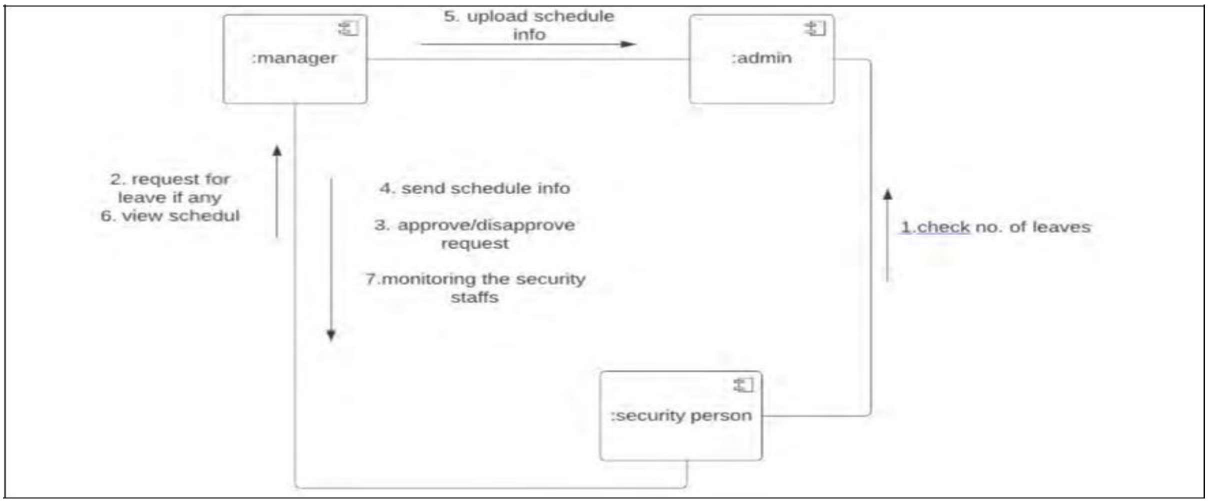


Fig 3.5: Collaboration Diagram for an Online Campus Security Management System

PACKAGE DIAGRAM:

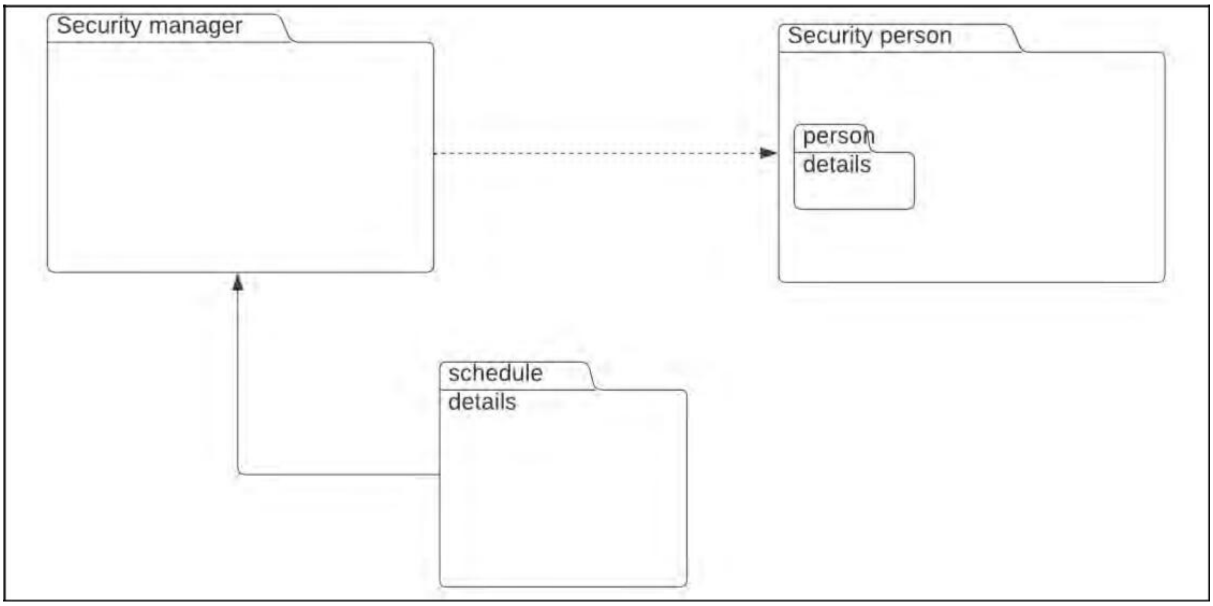


Fig 3.6: Package Diagram for an Online Campus Security Management

System **COMPONENT DIAGRAM:**

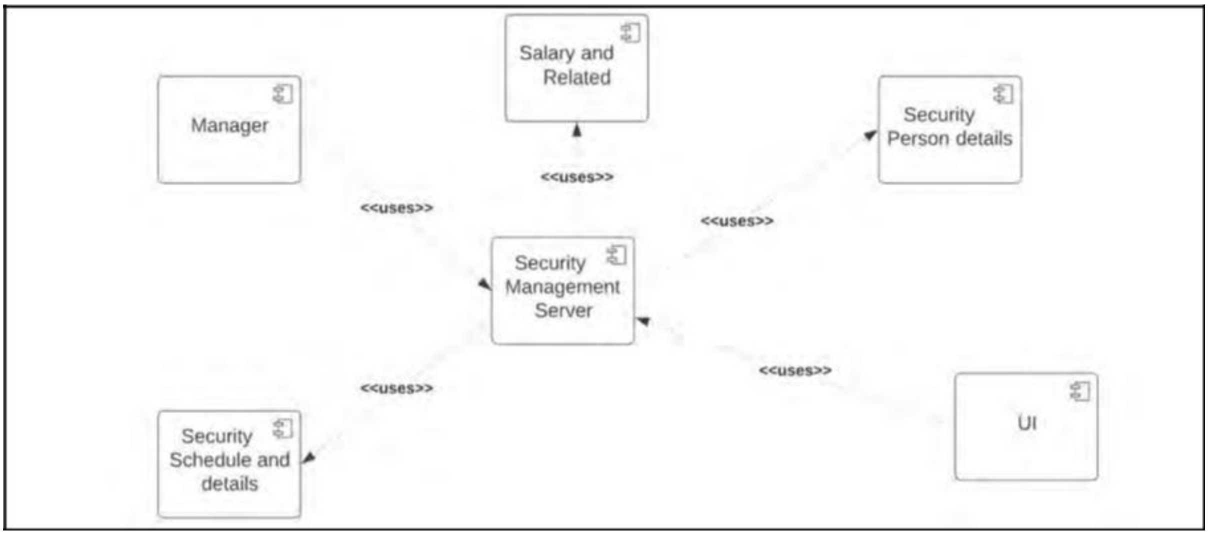


Fig 3.7: Component Diagram for an Online Campus Security Management

System **DEPLOYMENT DIAGRAM:**

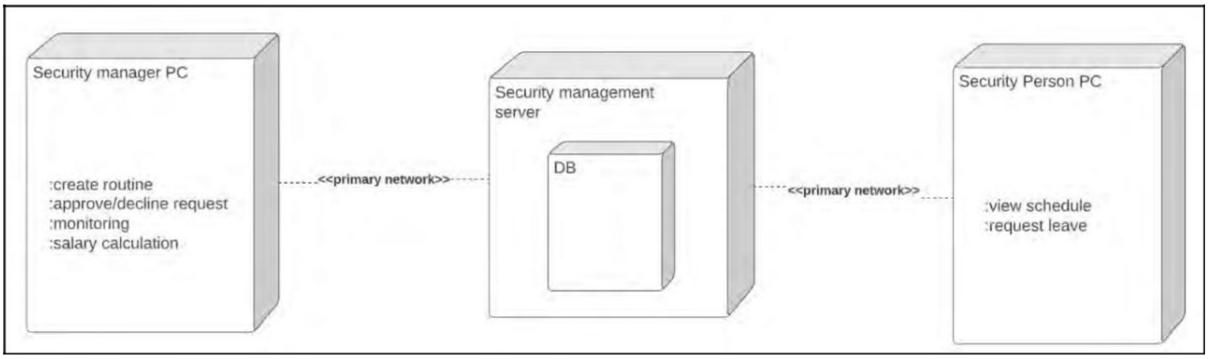


Fig 3.8: Deployment Diagram for an Online Campus Security Management System

04: TO DRAW THE UML DIAGRAM FOR A HOSPITAL MANAGEMENT SYSTEM

THEORY:

Hospital Management System (HMS) is powerful, flexible, and easy to use system, which is very helpful in a hospital environment, in order to maintain a hospital efficiently. Hospital Management System designed for multi-specialty hospitals, to cover a wide range of hospital administration and management processes. It is an integrated end-to-end Hospital Management System (HMS) that provides relevant information across the hospital to support effective decision making for patient care, hospital administration and critical financial accounting, in a seamless flow.

In an HMS need to have entry of all the data related to patients, doctors, staffs, and administrator(s). Each patient admitted in the hospital needs to enter their personal information, which are accessible by authorized doctors and staffs. On the other hand, a patient should have access permission to download any medical report associated with him/her. Billing process should be included in the system, such that it can easily be update after each payment.

INPUTS:

Hospital information (name, address, branch, department, facilities)
Employee information (name, employee id, address, contact details)
Doctor information (name, employee id, address, contact details, specialization, degrees)
Patient information (name, id, address, age, payment, others)

OUTPUTS:

Patient details
Patient’s list
Receipt
generation Report
generation

CONSTRAINTS:

All the users must register into the system though the web interface.
All the users must login into the web interface through userid (must be unique) and password
Patients should allow to get login him/her-self in order to collecting own information
On login into the system, a doctor must be able to see the patients list under his/her observation
One patient must not access the information of some other patients
All the patients must be able to download all the medical report and bill as per his requirement by login into the system
Add, delete, and update must be reflected immediately in the system

EXPLANATION:

We have three actors in our case diagram: a doctor, a patient, and an employee. After making a payment, the patient will see which physicians are available and schedule an appointment. All of these appointments and payment-related actions will be handled by a hospital staffer, who will update the doctor's calendar accordingly. The doctors will have access to their schedules, appointments, and patient information. Following the examination, the doctor will prepare a medical report that the patient can review.

USECASE DIAGRAM:

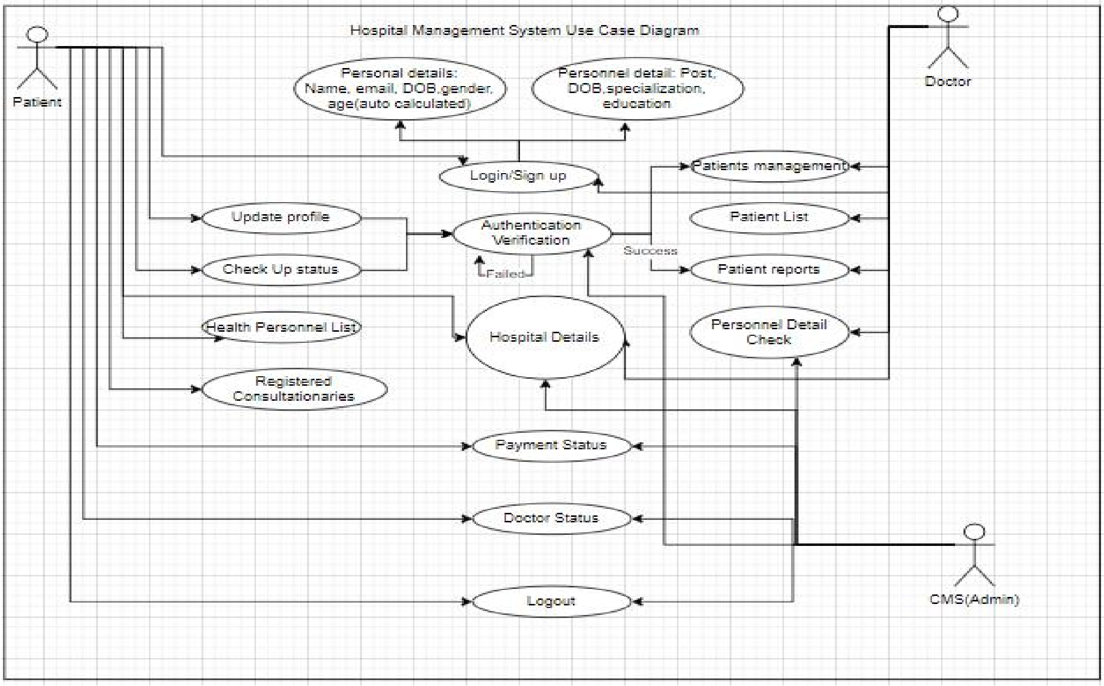


Fig 4.1: UseCase Diagram for a Hospital Management System

CLASS DIAGRAM:

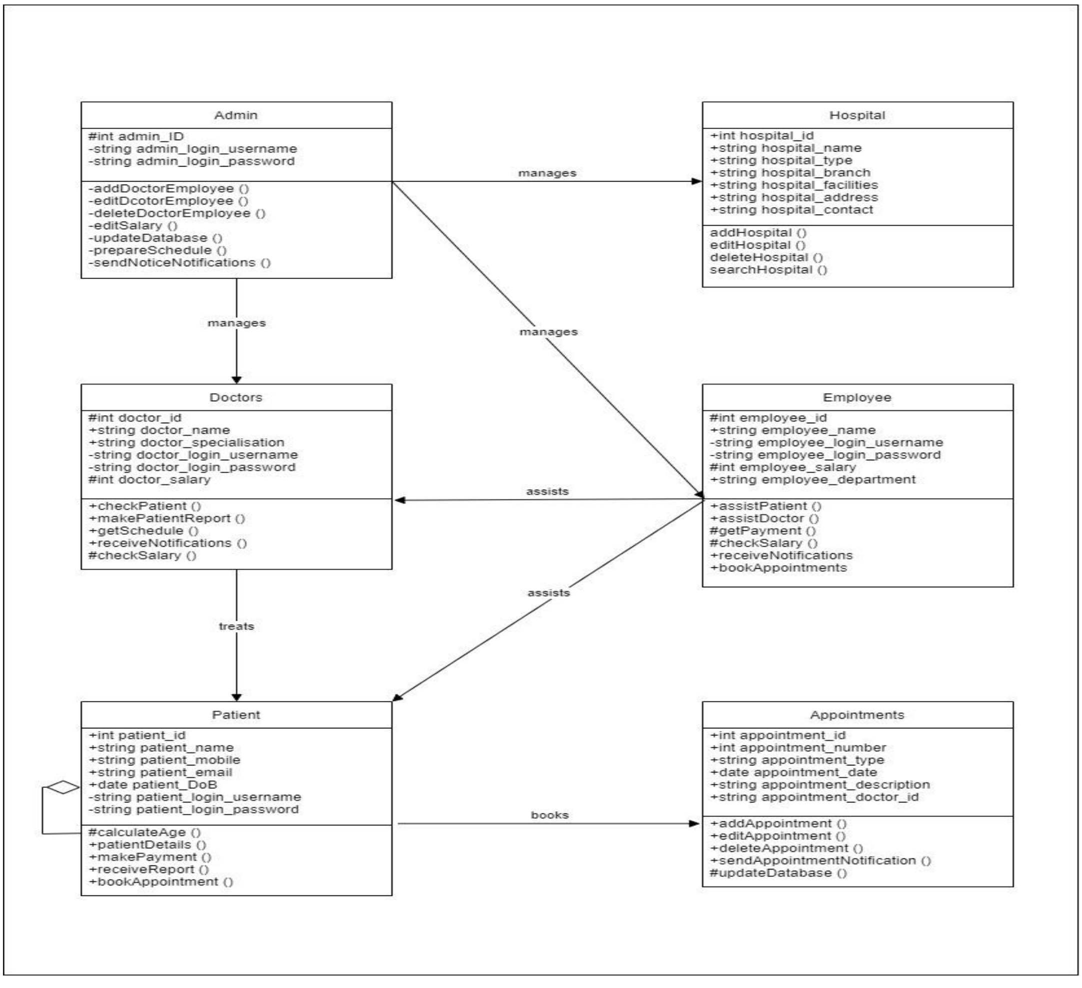


Fig 4.2: Class Diagram for a Hospital Management System

SEQUENCE DIAGRAM:

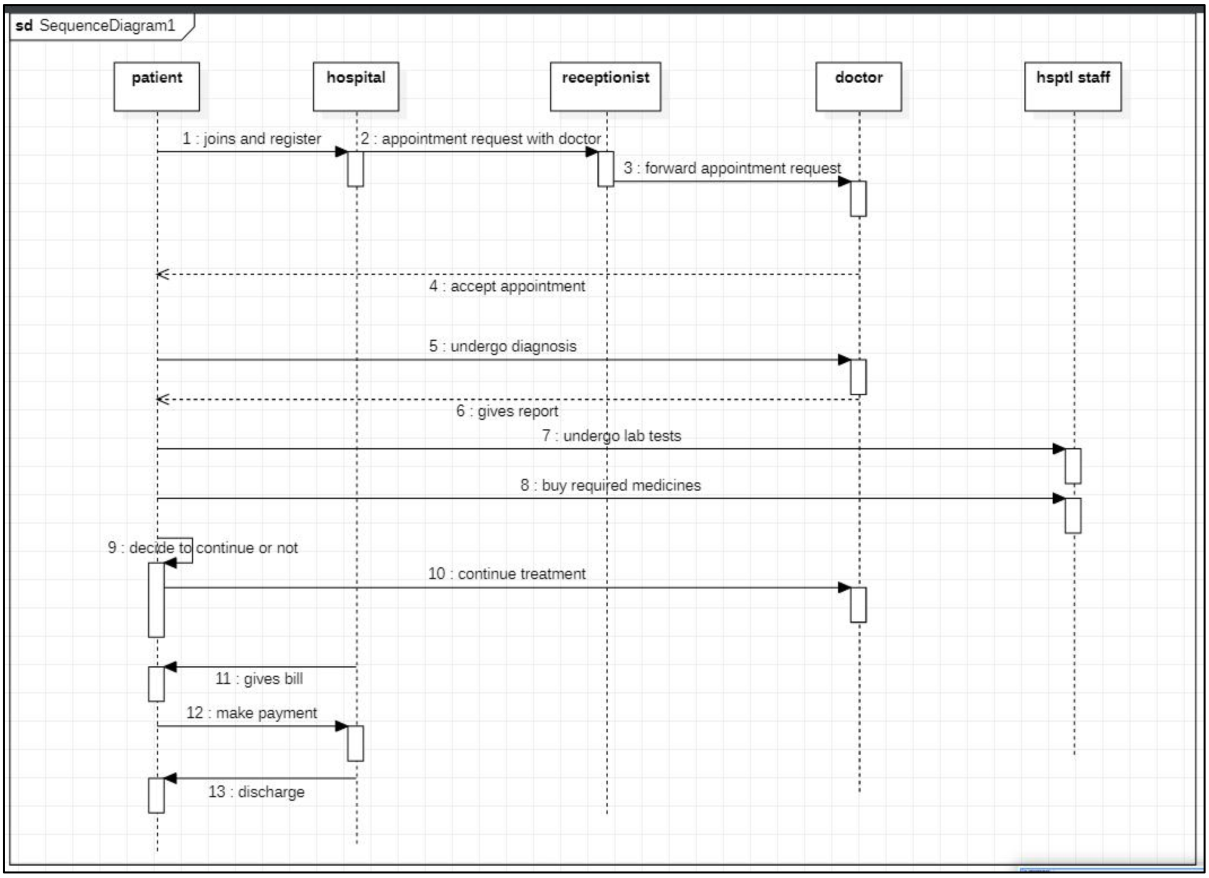


Fig 4.3: Sequence Diagram for a Hospital Management System

PACKAGE DIAGRAM:

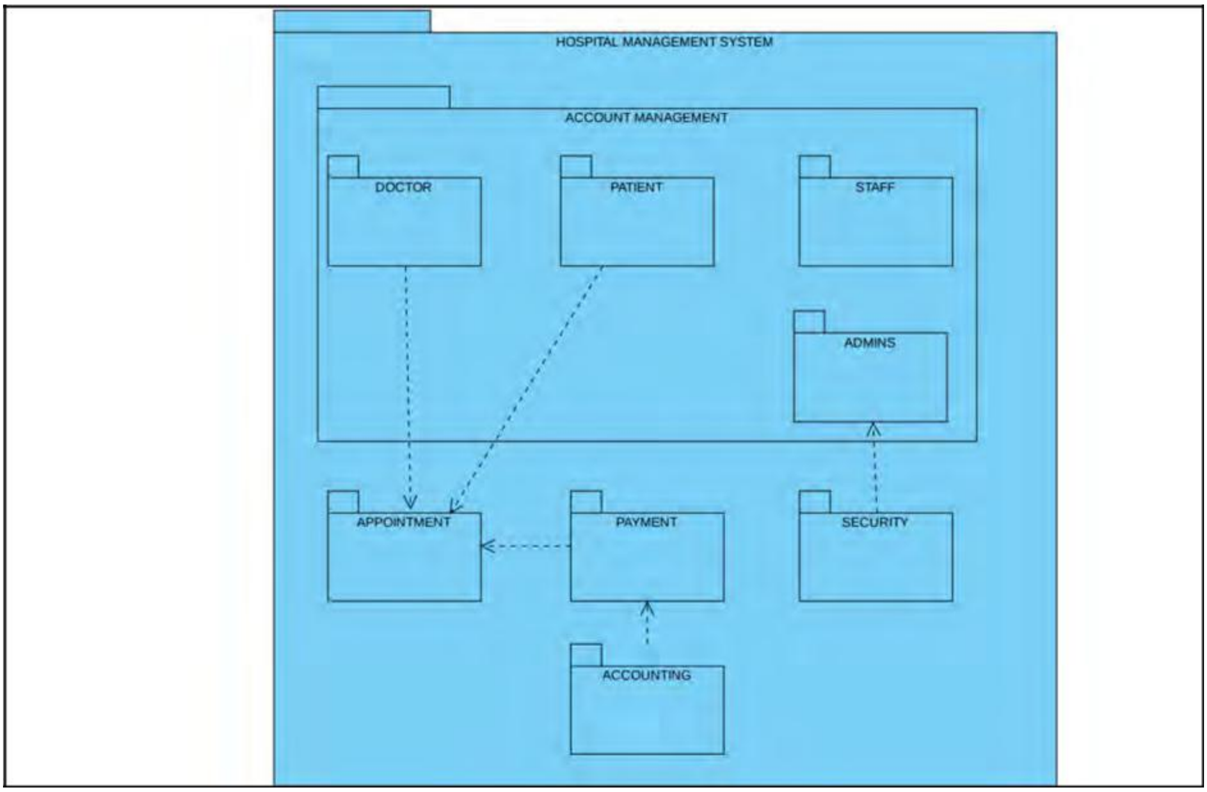


Fig 4.4: Package Diagram for a Hospital Management System

COMPONENT DIAGRAM:

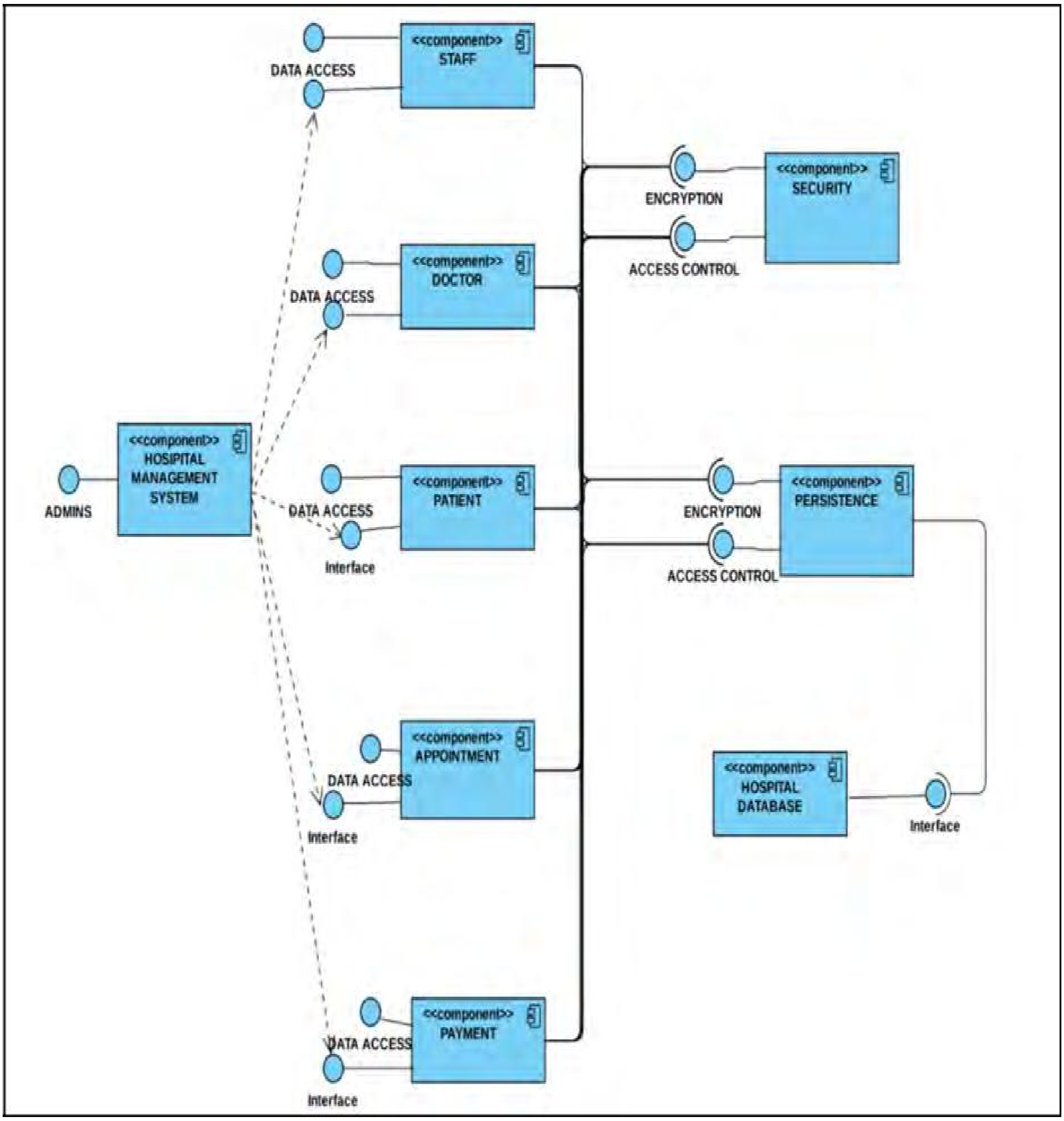


Fig 4.5: Component Diagram for a Hospitable Management System

05 TO DRAW THE UML DIAGRAM FOR A PAYROLL MANAGEMENT SYSTEM

THEORY:

The Employee and Payroll Systems objective is to provide a system which manages the employee details, the payroll activity done in a company depending upon the employee’s attendance and its calculation which is very huge. The users will consume less amount of time through computerized system rather than working manually. The system will take care of all the payroll activities like managing each employee’s attendance, the number of leaves taken by that particular employee and calculation in a very quick manner and it avoids.

Data storing is easier. Paper work will be reduced and the company staffs spend more time on monitoring the progress. The system is user friendly and easy to use. All the important data’s will be stored in the database and it avoids any miscalculation.

The “Employee and Payroll System “is based on maintaining each employee records and calculating his/her salary depending on the workdays. The first activity is based on saving the employees details where each employee will be given a unique Employee ID. Now based on the no of days an employee attended per month, salary will be calculated by checking the no of workdays of a company and his/her basic salary and a separate salary slip will be provided for reference.

INPUTS:

Employee details (employee no, name, address, designation, department, achievements)
Accounts details (salary of each employee, deduction, TA, DA, HRA, other allowance, PF)
Leave information (no of leave taken by each employee)

OUTPUTS:

Salary slip
Detailed salary report
Deduction details
Leave information

CONSTRAINTS:

Admin must have the permission to update the employee information manually
Manager should be able to see the details of an employee
Salary slip, detailed salary report, deduction details, and leave and information of a particular employee must not be accessible to another employee
The leave information must be update automatically, if an employee takes leave.

EXPLANATION:

To begin, we have three characters: a manager, an employee, and an administrator. The manager may examine personnel data and achievements, much like in the earlier trial with the Online Campus Security Management System. The employee can seek leave, which the boss can either approve or refuse. A manager may also see how many days an individual has taken off. The management then calculates the employee's pay after deducting deductions, TA/DA/HRA/PF, and any miscellaneous allowances, and generates a full salary report. The employee has access to his most recent pay stub. Admin is also an employee with additional access to employees' leave information and personal information, which the admin may change and modify.

USECASE DIAGRAM:

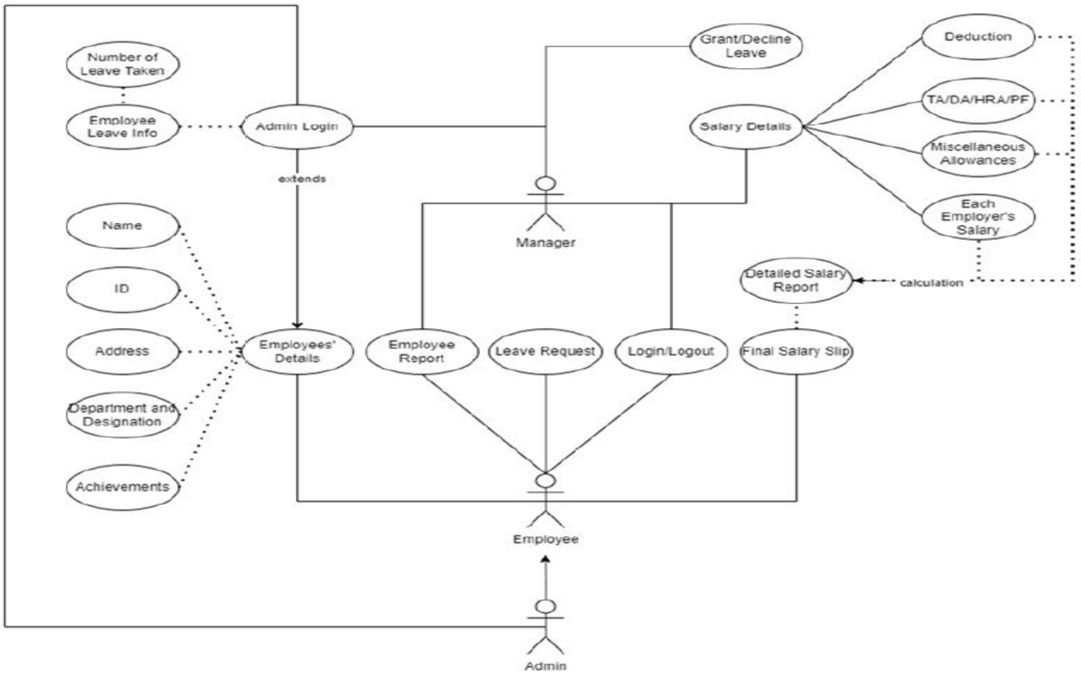


Fig 5.1:Use Case Diagram for a Payroll Management System

CLASS DIAGRAM:

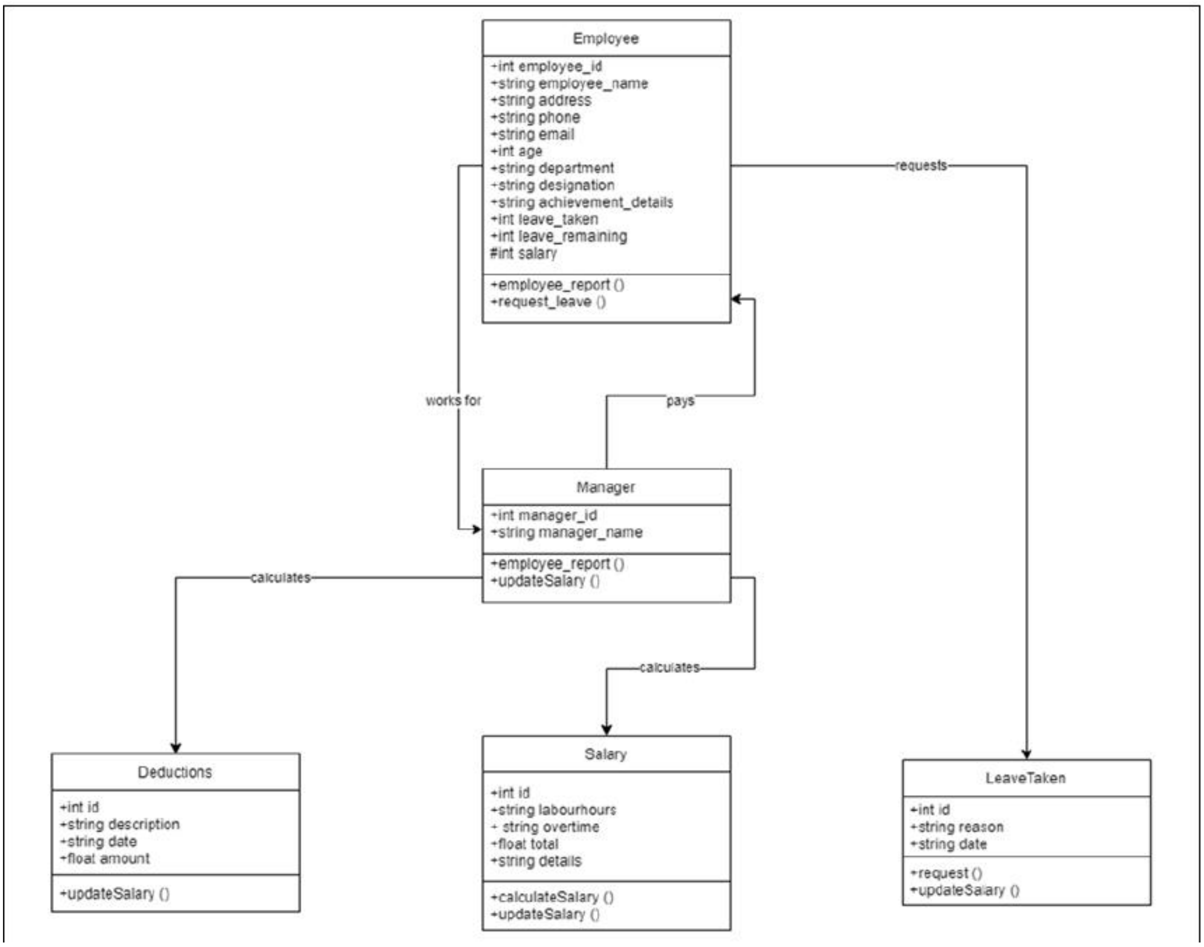


Fig 5.2: Class Diagram for a Payroll Management System

SEQUENCE DIAGRAM:

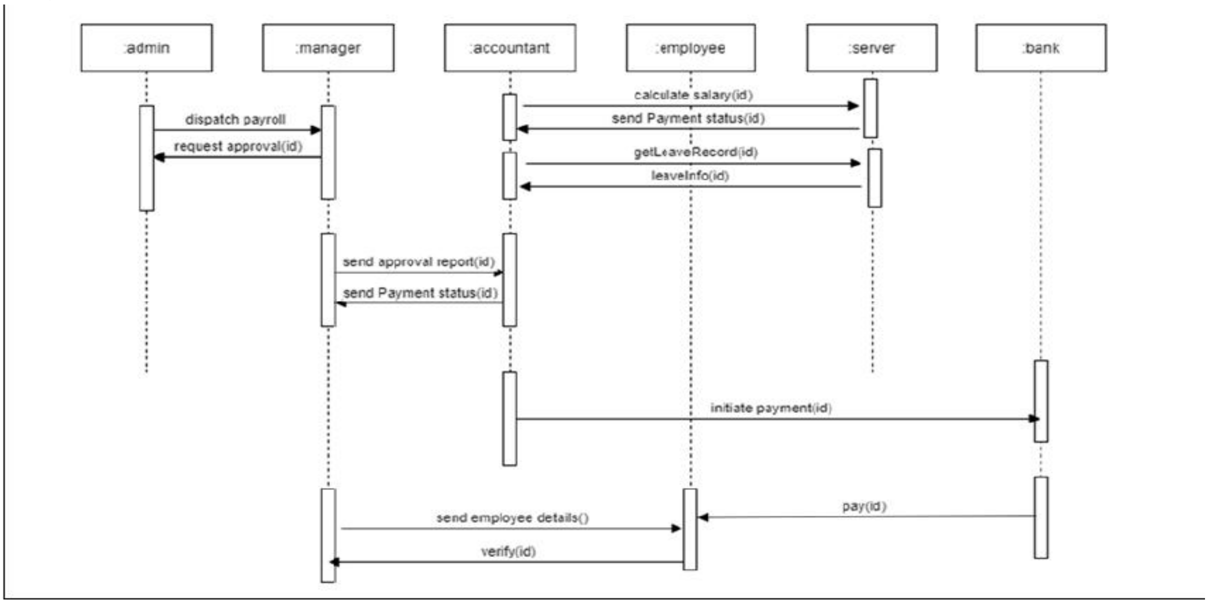


Fig 5.3: Sequence Diagram for a Payroll Management System

STATE DIAGRAM:

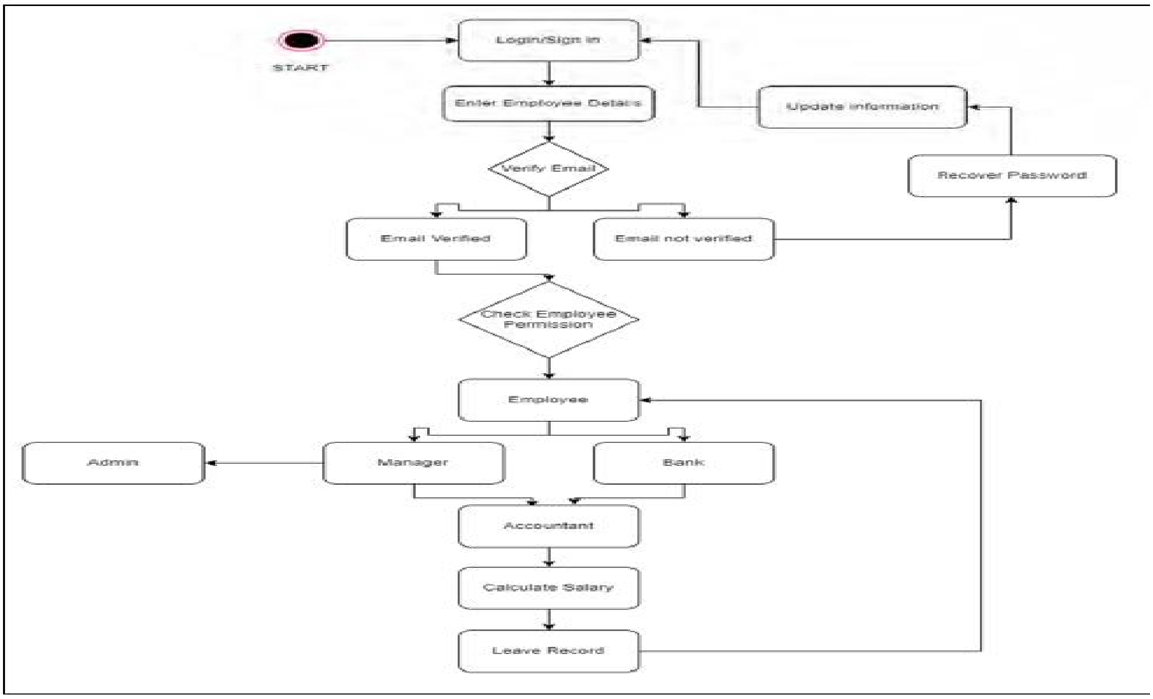


Fig 5.4: State Diagram for a Payroll Management System

COLLABORATION DIAGRAM:

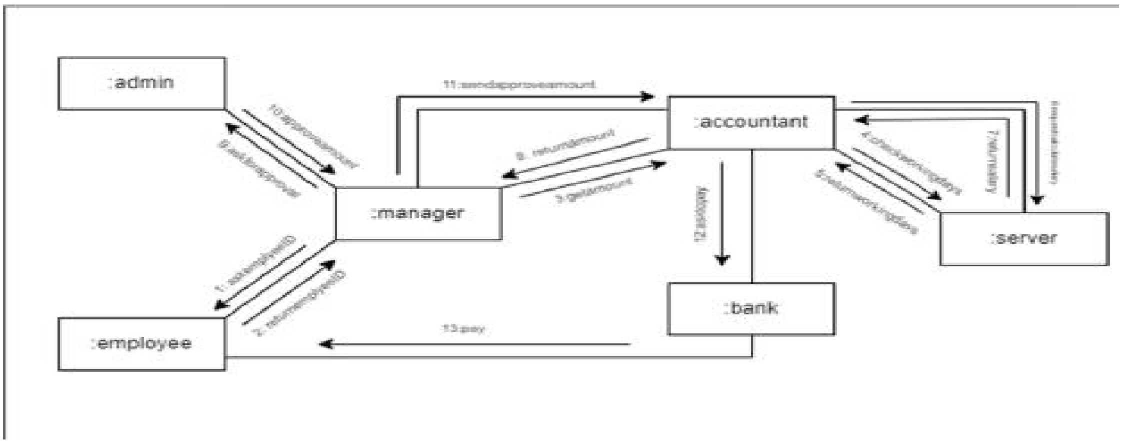


Fig 5.5: Collaboration Diagram for a Payroll Management System

PACAKGE DIAGRAM:

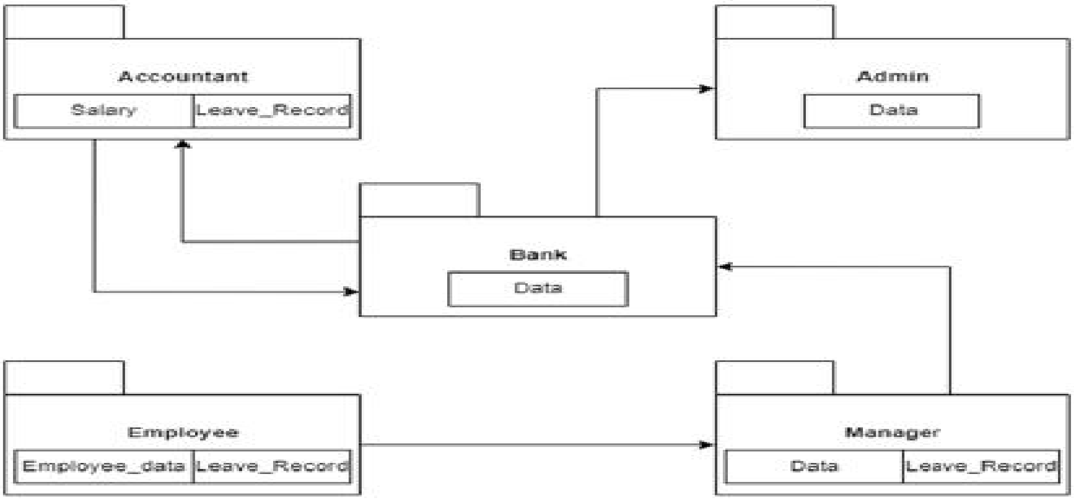


Fig 5.6: Package Diagram for a Payroll Management System

COMPONENT DIAGRAM:

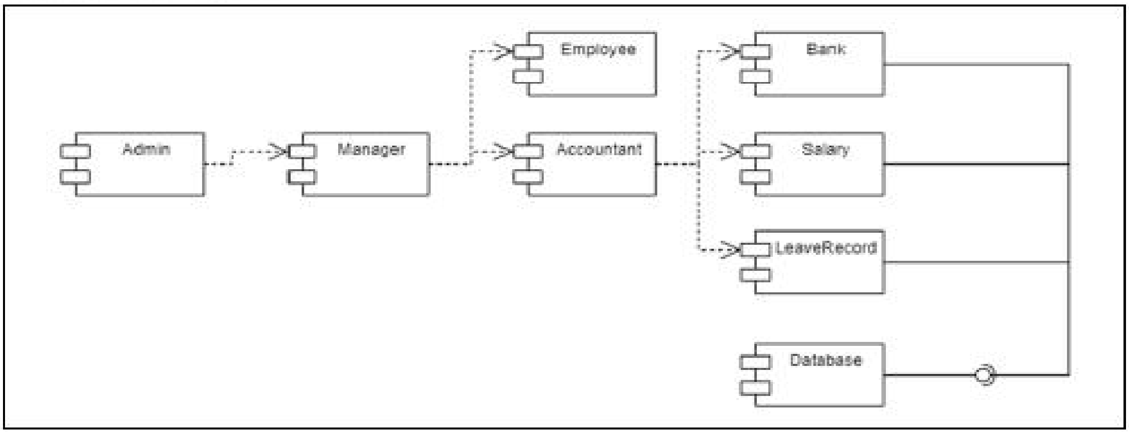


Fig 5.7: Component Diagram for a Payroll Management System

DEPLOYMENT DIAGRAM:

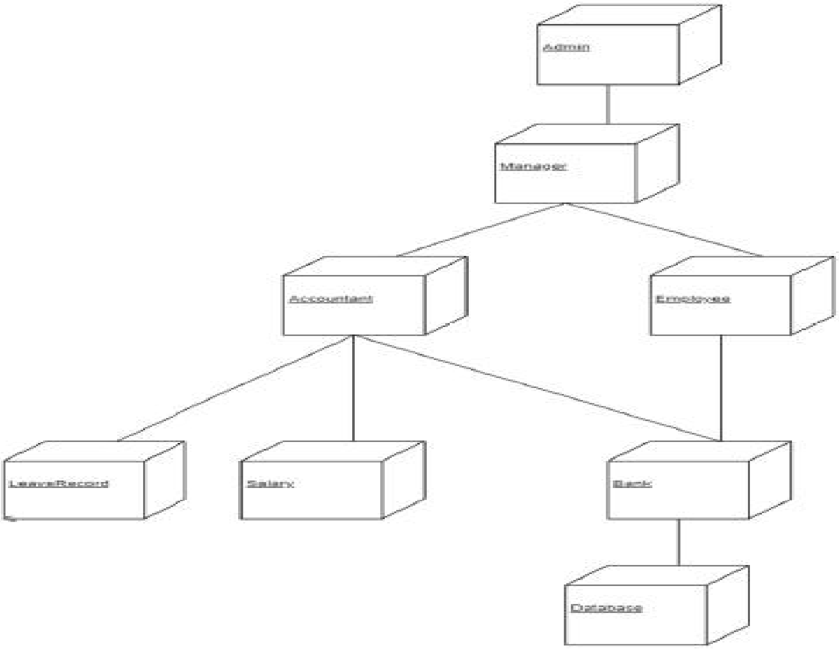


Fig 5.8: Deployment Diagram for a Payroll Management System

06: TO DRAW THE UML DIAGRAM FOR A SMART HOME MANAGEMENT SYSTEM

THEORY:

“Smart Home” is the term commonly used to define a residence that has smart devices which are able to communicate with the central controller. The smart home management system is a system which enabled the automation of smart home. Typically, it includes following Four Components:

Electronics Devices and Sensors: It includes devices such as air conditioning, TVs, computers, entertainment audio &amp; video systems, and camera systems. In addition to that, it can include sensors (temperature, humidity, light, and sound) for environment sensing.

Wireless network: It is a medium of communication between the central controller and devices/sensors. E.g., Wi-Fi.

Central Controller: The SHMS can operate in two modes (automatic and manual). In automatic mode, it receives streaming data from sensors and on/off the appliances according to the User Environmental Comfort range. However, in manual mode, it on/off the appliances according to the user request.

User Interface (e.g., Android App): It enables user and administrator to sign up and login before availing the services. After login user can give its Environmental Comfort range as per the offered services. The user can also see the available appliances and change its status (on/off) as per the preferences. There should be an option of registering a complaint in the case of system failure. After login the administrator can add new devices/sensors to the existing system. In addition to that he/she gets notified regarding the registered complaint. The billing calculation should be based on the energy consumed by the devices/ sensors. Moreover, the user should be notified through SMS in case of a general event (when operating in manual mode) and through SMS and Call in the case of any emergency.

INPUTS:

User Personal Information: (name, email, age, gender, mobile number, security question)
User Environmental Comfort range (Temperature, Humidity, Lighting, sound)
System Administrator Details: (mailed, contact no.)
Operating Mode (Automatic or manual)
Appliances Information (Type, Power, Unique Id, Status)
Sensor Information (Type, Id, operating range)

OUTPUTS:

Electricity Billing Details.
Complaint and
Feedback Alert by SMS
Alert by SMS and Call, in case of Emergency

CONSTRAINTS:

User/ Administrator must register with the SHMS to access the portal.
The mobile number must be a 10-digit number.
Login information consists of email address and
password Each appliance and sensors assigned a unique
ID
User environmental comfort range must be in the operating range of sensors.
Enable power saver mode only if operating mode is automatic.
In manual mode, every time the operating status of appliances change system must show the current billing status.
Every time user logins system must show current environmental condition.

EXPLANATION:

There are two actors: the user and the system administrator, as well as a sensor management system. The user may customise the information about the appliances, choose the working mode for each one, and add or delete them from the smart home network. The user may also change the comfort level of the surroundings. Furthermore, the user has the option of filing a complaint or providing feedback to the system administrator. The system administrator can view and respond to complaints and feedback. In addition, the system administrator is responsible for computing and verifying electrical billing information. All of the appliances that the user has entered, as well as the user's comfortable environment range, are accessible to the sensor management system. In automated operation mode, the sensor takes over control of the appliance's administration.In the event of a system breakdown or other emergency, the sensor system will send an SMS and an automated phone call to both the user and the system administrator. In addition, in the event of a small system failure, the sensor management system can provide automated feedback to the system administrator.

USE-CASE DIAGRAM:

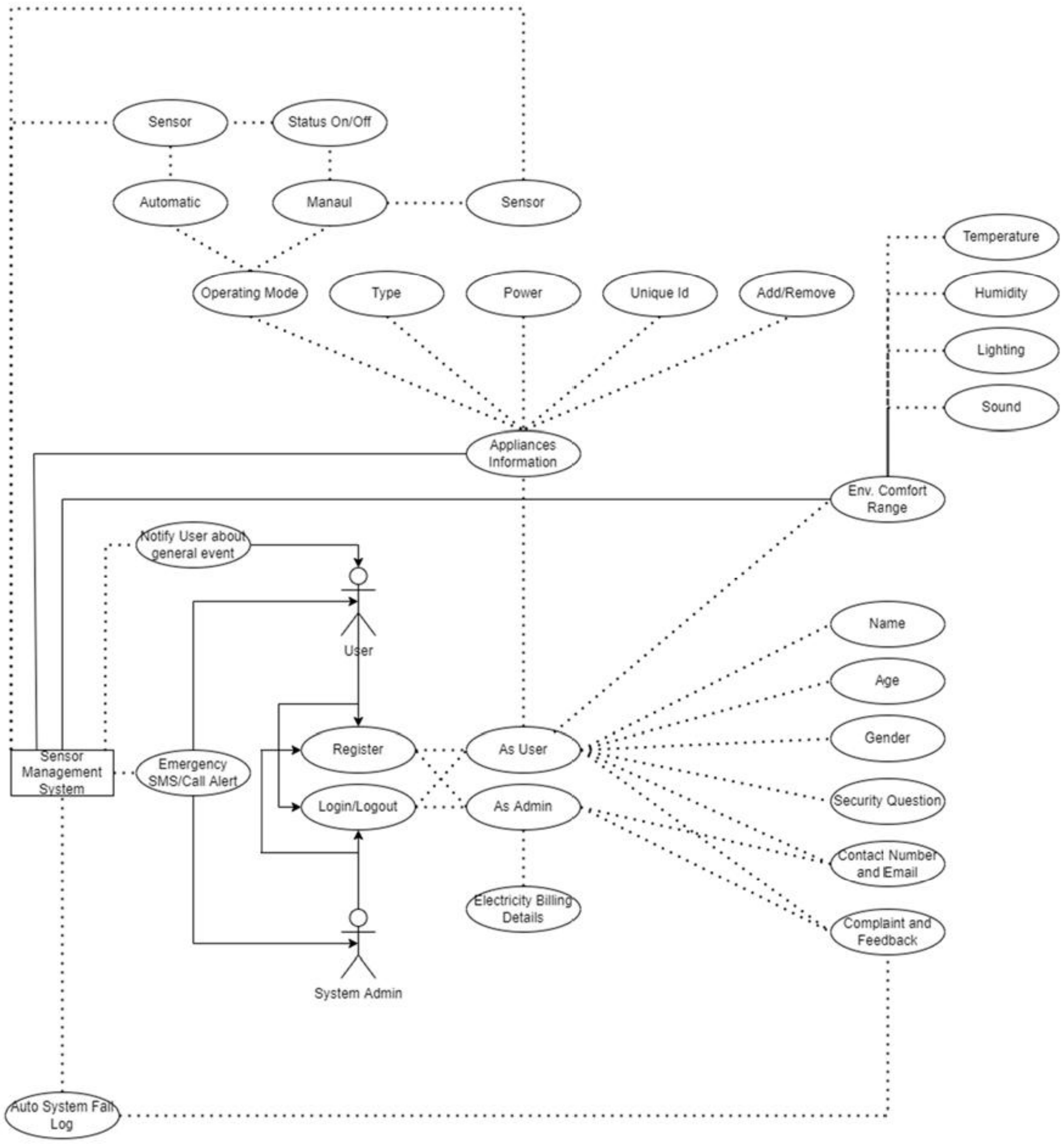


Fig 6.1: Use-Case Diagram for a Smart Home Management System

SEQUENCE DIAGRAM:

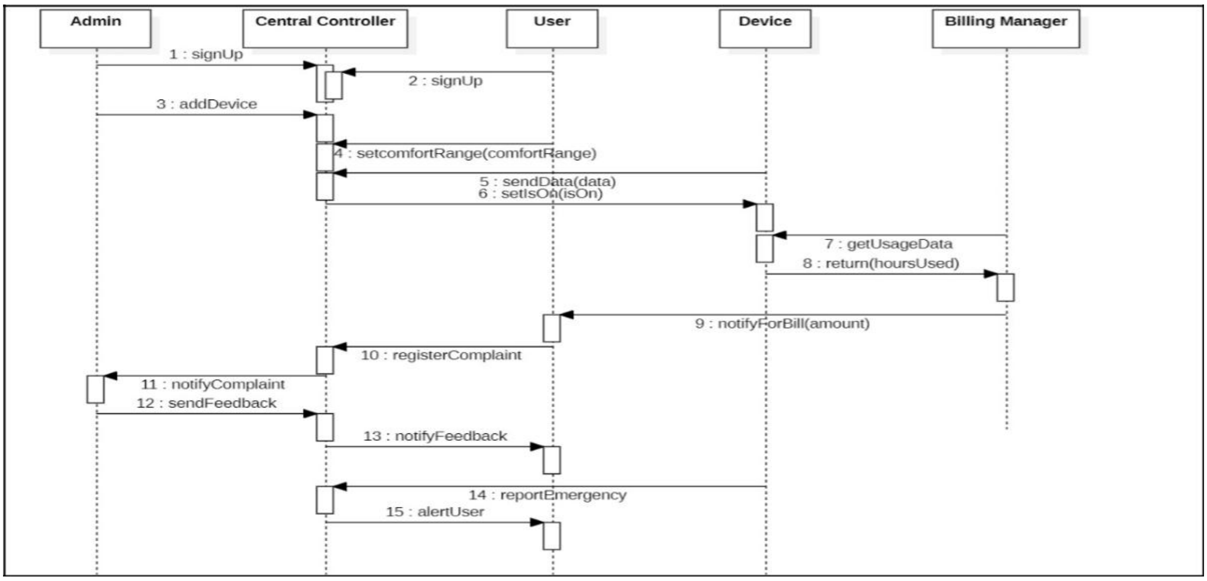


Fig 6.2: USE-Case Diagram for a Smart Home Management System

DEPLOYMENT DIAGRAM:

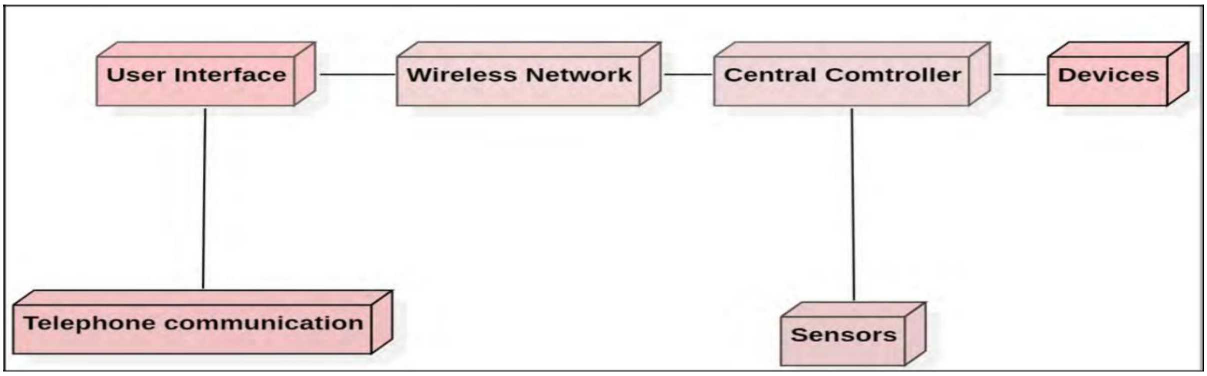


Fig 6.3: Deployment Diagram for a Smart Home Management System

STATE DIAGRAM:

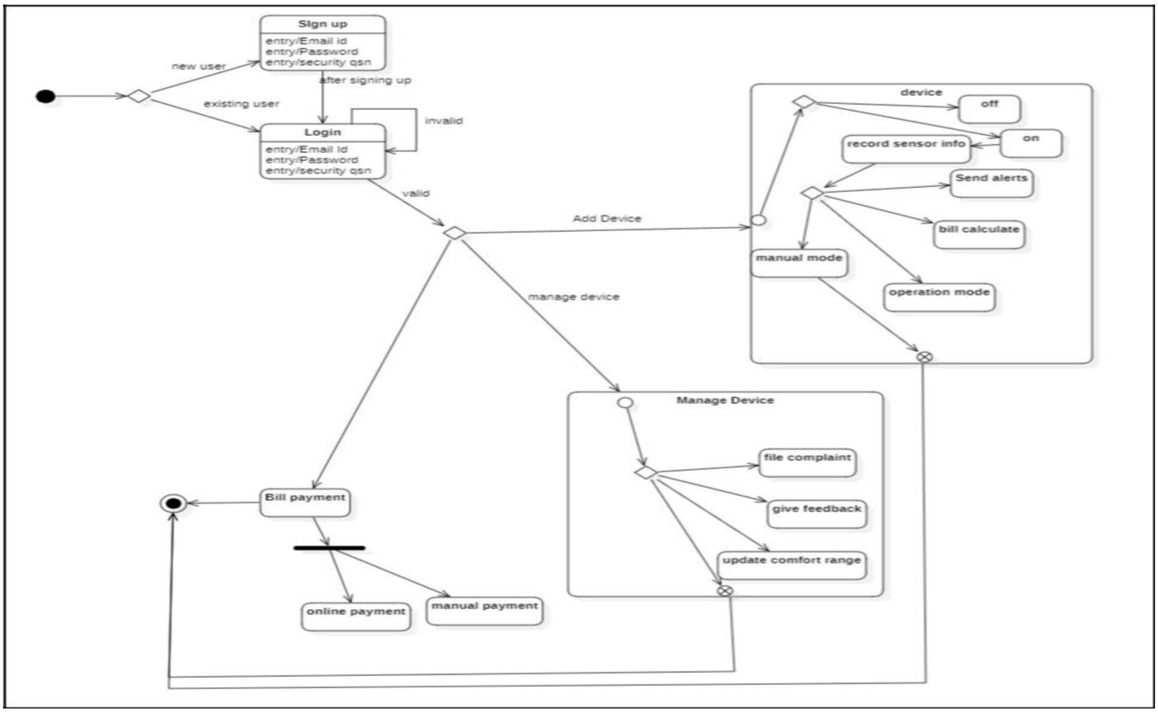


Fig 6.4: State Diagram for a Smart Home Management System

COLLABORATION DIAGRAM:

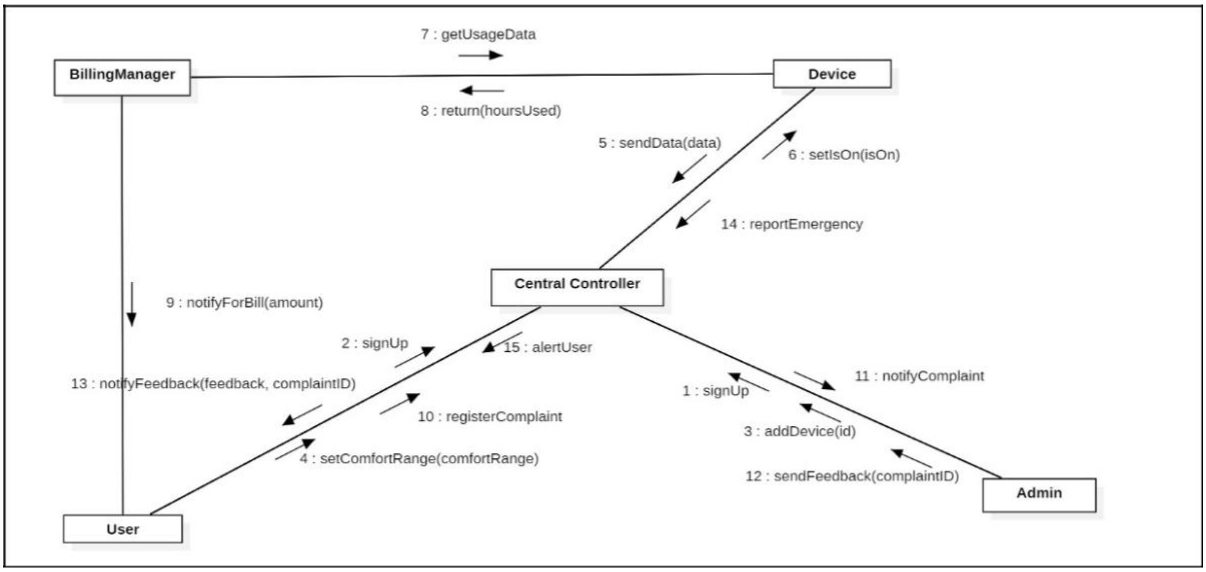


Fig 6.5: Collaboration Diagram for a Smart Home Management System

PACKAGE DIAGRAM:

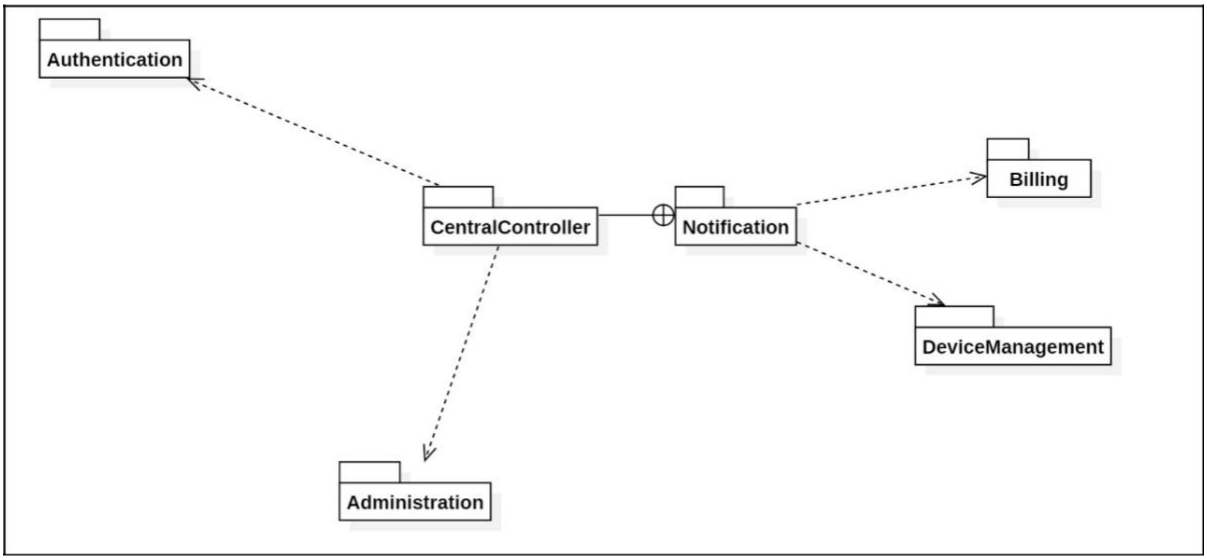


Fig 6.6: Package Diagram for a Smart Home Management System

COMPONENT DIAGRAM:

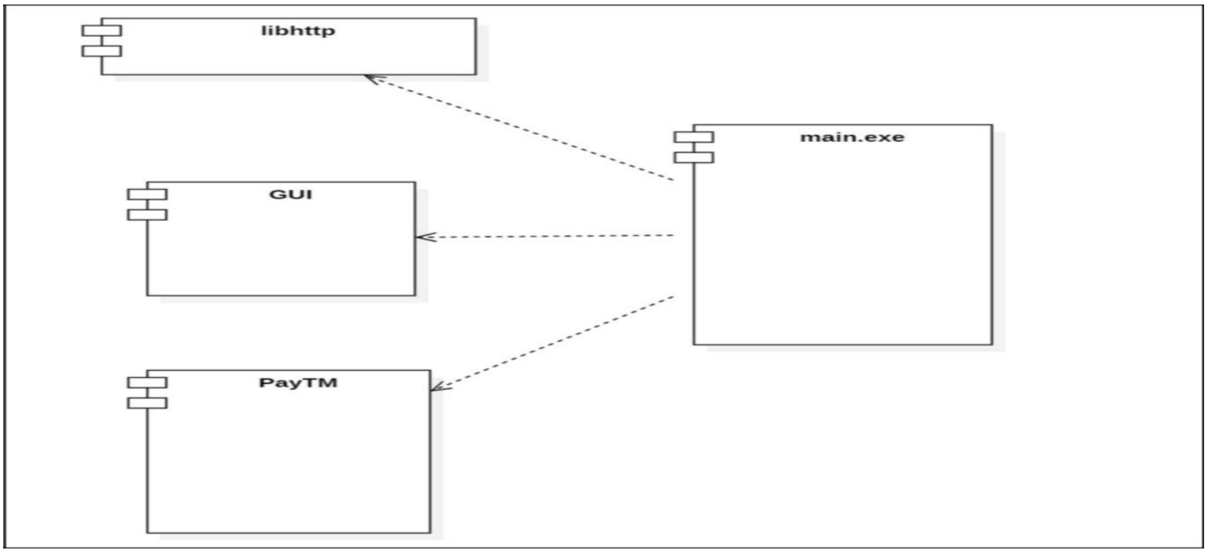


Fig 6.7: Component Diagram for a Smart Home Management System

06 TO DRAW THE UML DIAGRAM FOR AN ONLINE EXAMINATION SYSTEM

THEORY:

Now-a-days, Online examination system has become popular for competitive examinations because of its unique features such as auto-evaluation, speed and accuracy. Moreover, it also helps environments by reducing the use of paper. In such a system, students are asked to select answers from multiple options given for a single question. Likewise, there are several questions which appear in the students' systems. The questions and multiple options are saved in a database along with desired answers. Typically, a student can edit an answer after saving it, however, editing cannot be done after submitting the answer. Another user is also there – administrator. The administrator can create, modify and delete questions and accordingly, the question is updated in the system.

INPUTS:

User Information with Code, so that all subjects can be identified using unique codes.
User Information
If Student- Student Information (Name, Roll No, Email Address, Contact Number, Password)
If Administrator (Email Address, Password)
Set of Questions with multiple answers for each stored in a database along with desired answers.

OPERATIONS:

Administrator

- Log-In
- CREATE, MODIFY or DELETE questions. Accordingly, the question set must be updated.
- Log-Out

Student

- Log-In (Time starts)
- Answer the questions – SAVE and SUBMIT
- Log-Out (Automatically logged out after Timeout)

OUTPUTS:

Display the result in DESCENDING order according to obtained marks with Roll Number.
The result is also saved into a database for future use.

CONSTRAINTS:

All users MUST register themselves into the system.
Same set of questions should appear to all students.
Log-In information contains only Email Address and Password. After clicking on SUBMIT, selected answers cannot be changed. After clicking on SAVE, selected answers can be changed.

EXPLANATION:

We only have two actors in an online examination system: students and instructors, as well as a database system that maintains question papers and correct answers. The professors are also administrators. A question or a question paper can be created, modified, or deleted by the faculty. Faculty can edit accurate answers to questions in the database system using admin safe access. The student can next look through the question paper and fill in the blanks on the response sheet. The answer sheet can be saved or submitted by the student. The answers on the answer paper cannot be changed once they have been submitted, and the answer paper will be exposed to the faculty for review. After that, the faculty assesses the answer sheet and calculates the grades. The students can then see the final published result, which is listed in descending order.

USECASE DIAGRAM:

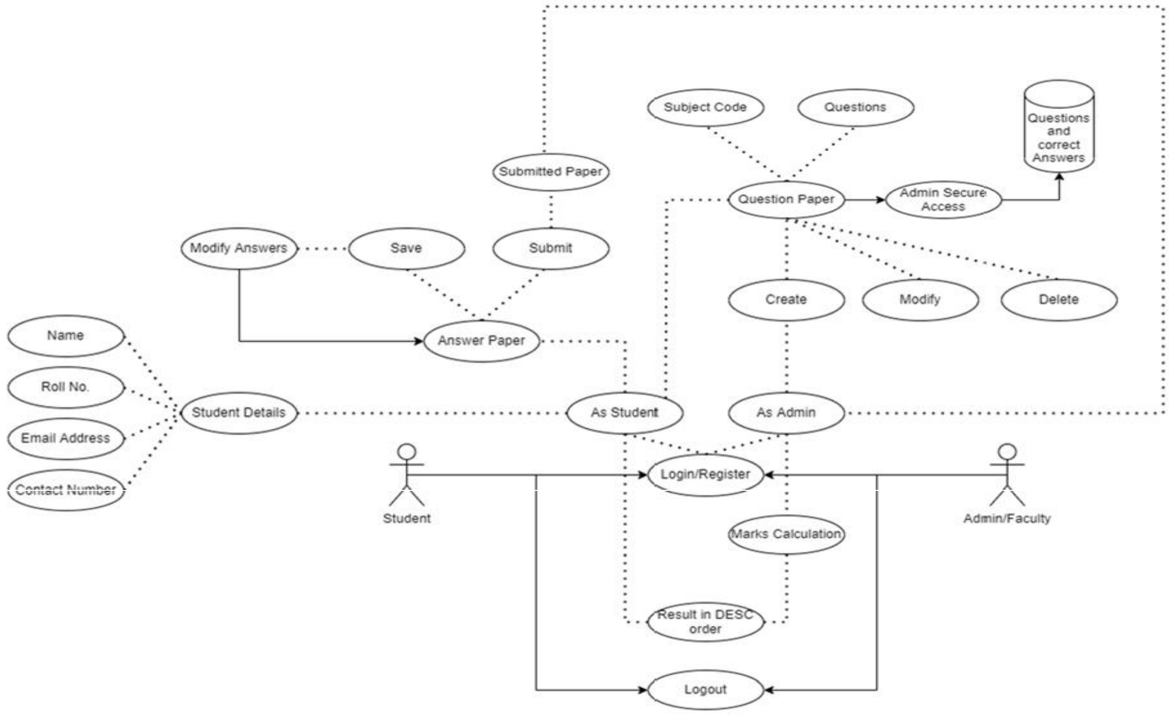
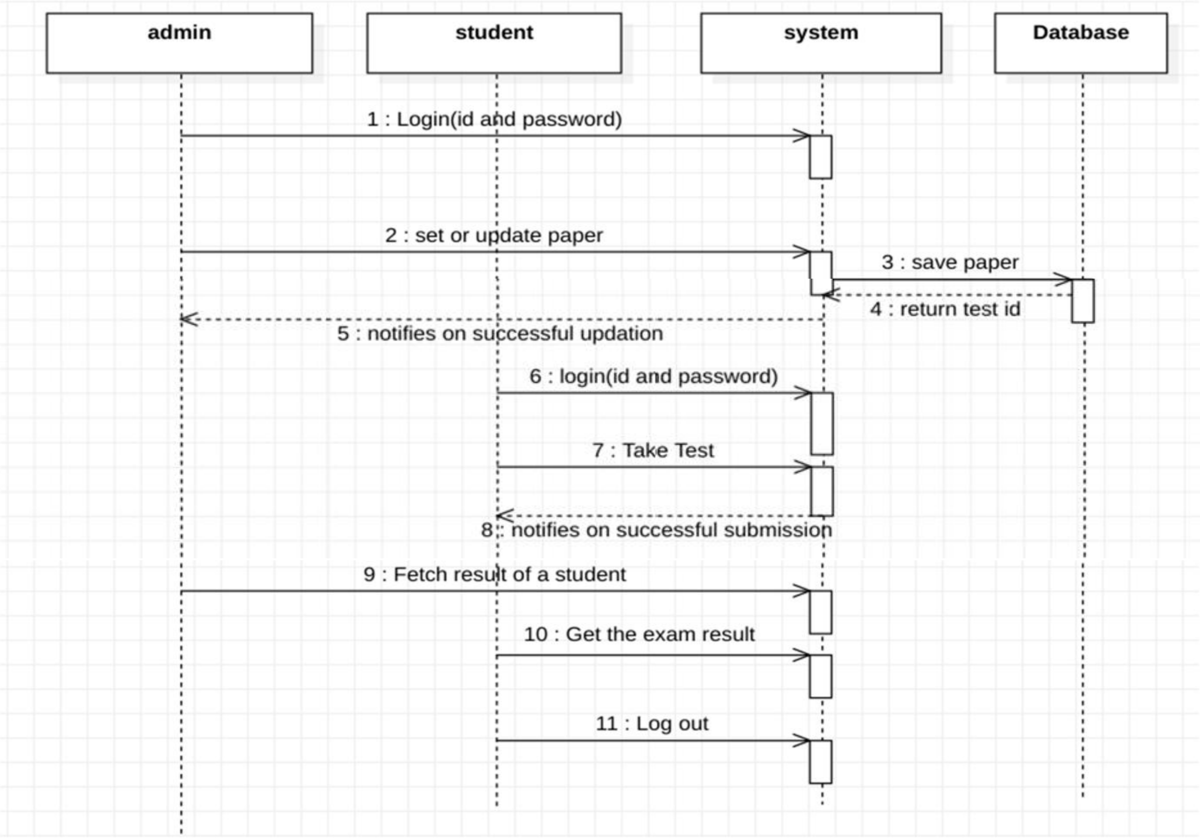


Fig 7.1: UseCase Diagram for an Online Examination System

SEQUENCE DIAGRAM:



admin
student
system
Database
1 : Login(id and password)
2 : set or update paper
3 : save paper
4 : return test id
5 : notifies on successful updation
6 : login(id and password)
7 : Take Test
8 : notifies on successful submission
9 : Fetch result of a student
10 : Get the exam result
11 : Log out

Fig 7.2: Sequence Diagram for an Online Examination System

STATE DIAGRAM:

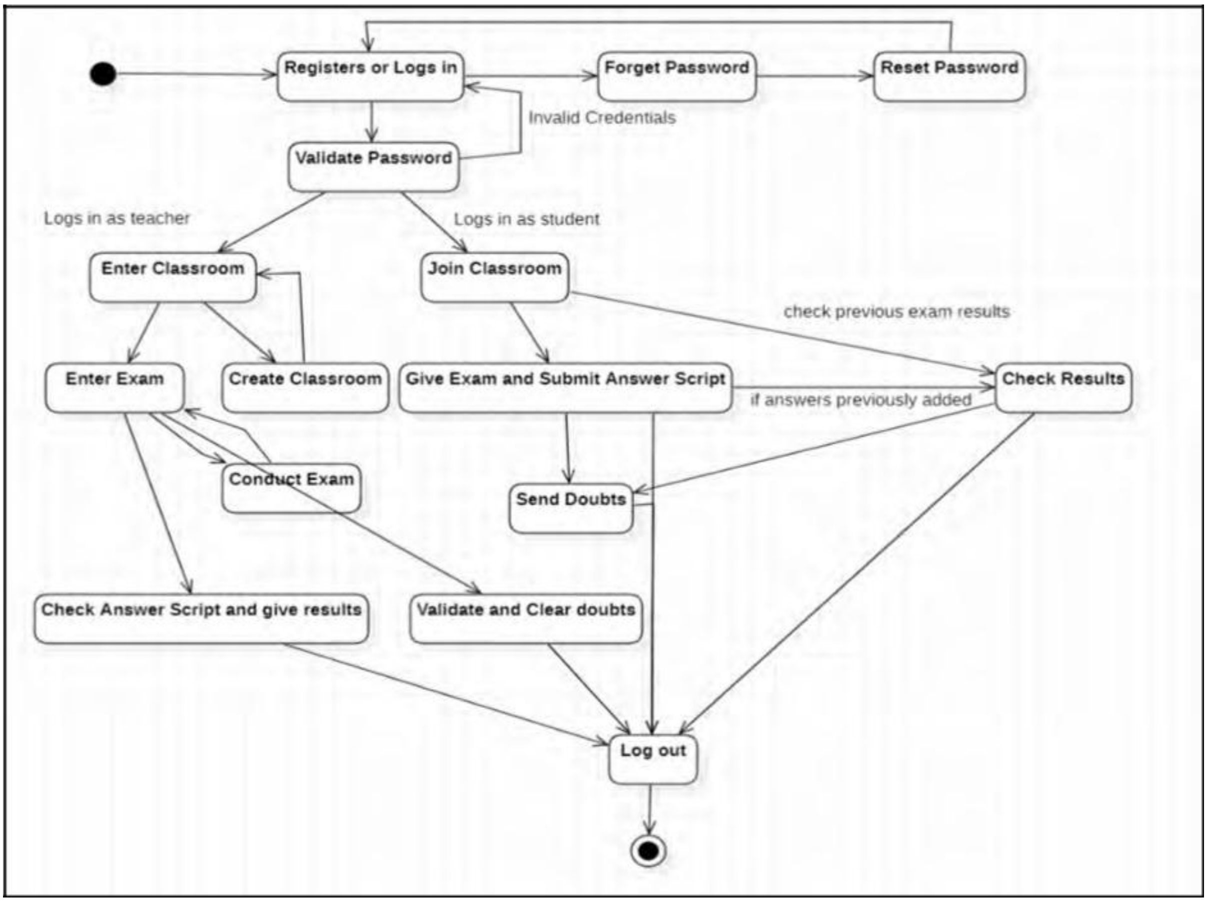


Fig 7.3: State Diagram for an Online Examination System

COLLABORATION DIAGRAM:

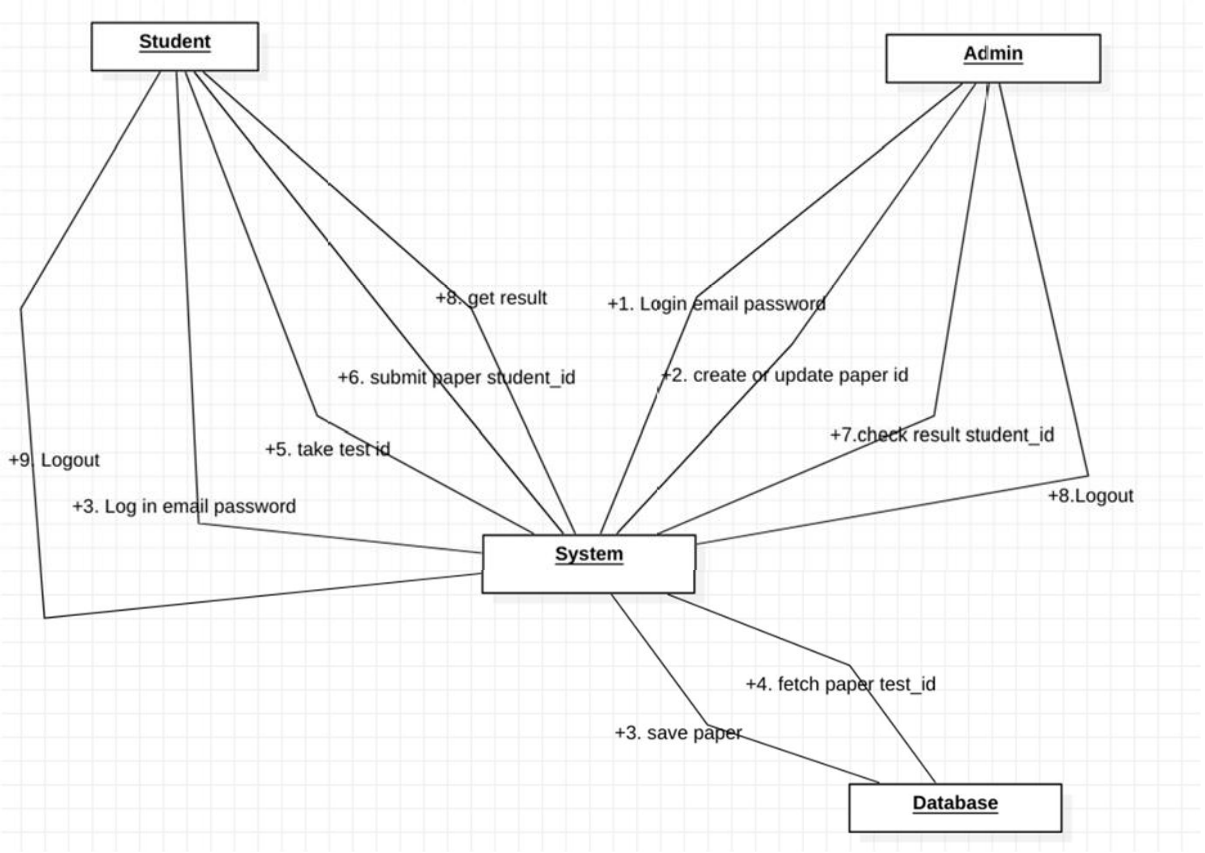


Fig 7.4: Collaboration Diagram for an Online Examination System

DEPLOYMENT DIAGRAM:

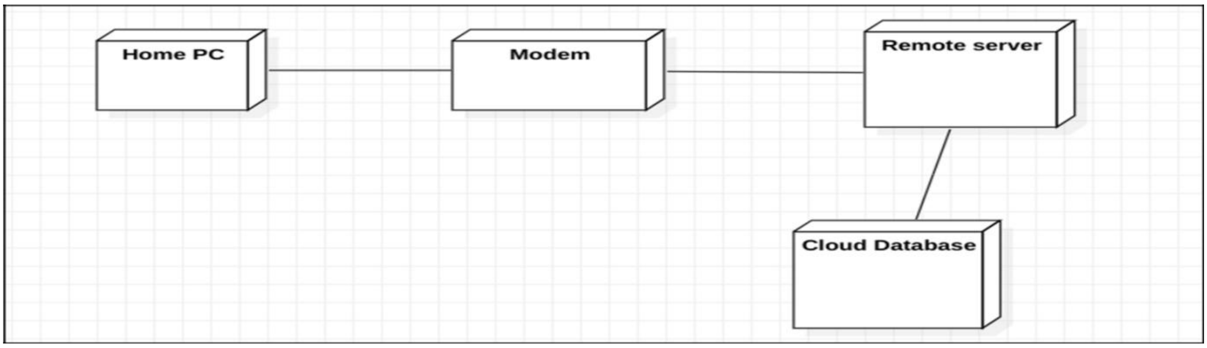


Fig 7.5: Deployment Diagram for an Online Examination System

PACKAGE DIAGRAM:

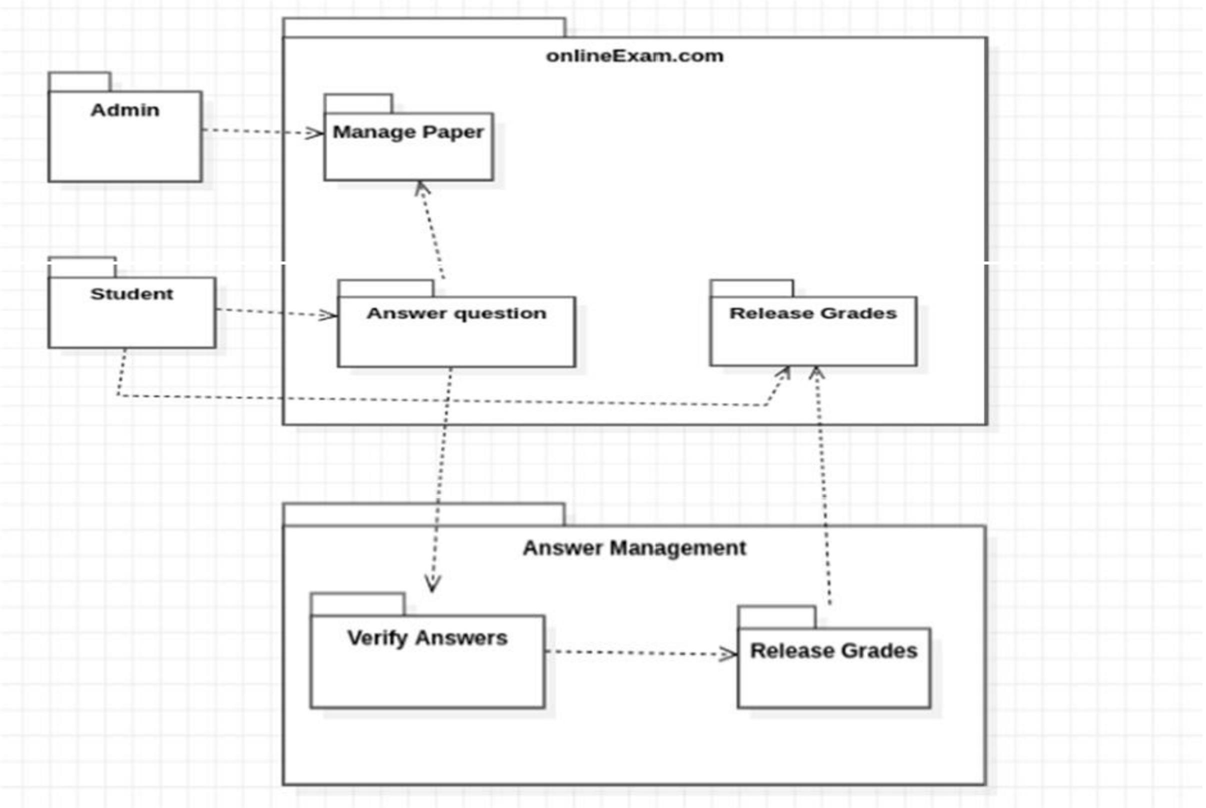


Fig 7.6: Collaboration Diagram for an Online Examination System

COMPONENT DIAGRAM:

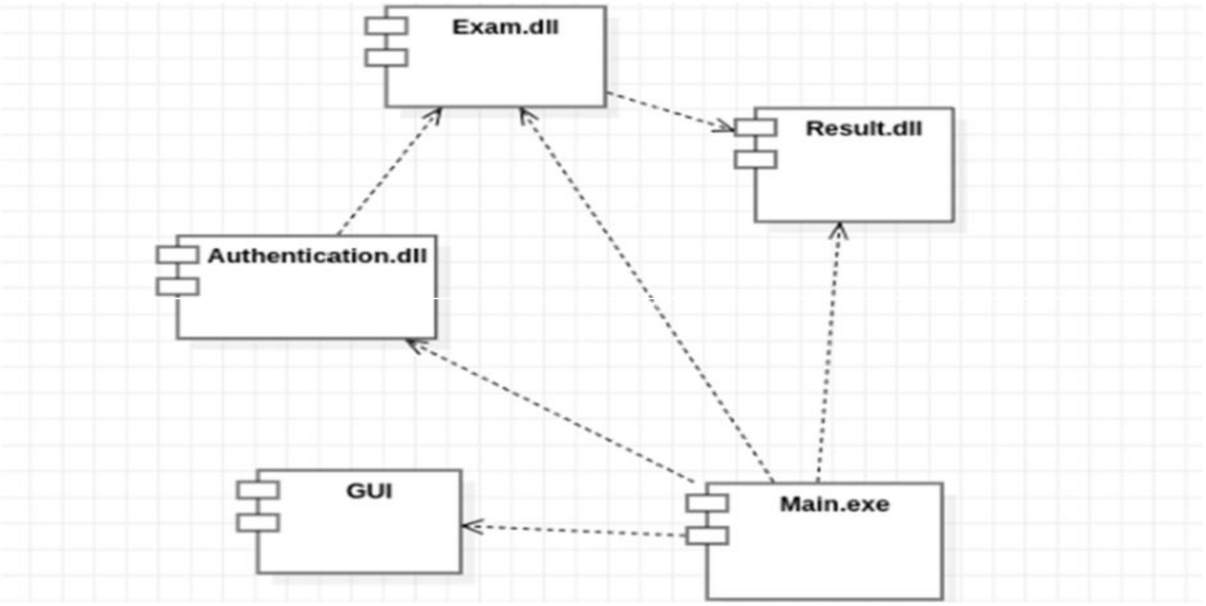


Fig 7.7: Collaboration Diagram for an Online Examination System

08 TO DRAW THE UML DIAGRAM FOR AN ONLINE TOLL PLAZA SYSTEM

THEORY:

Now-a-days, cashless transaction is becoming popular among the users because it is easy to handle, and it does not require to carry cash in hand. Typically, in India, road tolls are collected from cars manually for which the cars need to stop to pay the toll fee. In contrast, the objective is to make the system Online, so that the toll fee is automatically deducted from the user. Therefore, users credit their Online account (consider this as e-Wallet), and money is automatically deducted when the cars pass the toll system. As a result, the users do not have to wait for manual toll fee payment. Concurrently, administrator can also view all transactions from anywhere. Finally, the administrator can view the total income in a day-to-day basis, and can also analyse the traffic pattern as well.

INPUTS:

User Information (Name, Car Number, Email Address, Password, Money in e-Wallet)

Administrator Information (Email Address, Password)

OPERATIONS:

Patient

- Log-In
- Credit in e-Wallet
- Check e-Wallet Balance
- Log-Out

System

- Check the car number
- Required Fee Available
    - Allow the car to pass
    - Deduct money from e-Wallet
- Required Fee NOT Available
    - Do Not Allow the car to pass
    - Fee Payment is done manually
- Allow the car to pass
- Total Income is stored in a

database Administrator

- Log-In
- View transactions
- View total income

OUTPUTS:

Display day-wise transactions to administrator

User can view his/her own transactions

CONSTRAINTS:

All users MUST register themselves into the system with their car numbers.

A user can only view his/her own transactions

Administrator can view all transactions

EXPLANATION:

The online toll plaza system contains two actors: a user and an administrator, as well as an automated system and a database system for storing total revenue. Personal information, vehicle information, and eWallet information are all updated by the user. The administrator then double-checks the state of the car and estimates the amount of money that has been saved. To recognise the automobile, the automatic system pulls data from the car details and analyses eWallet transaction records to determine whether or not to provide passage access to the vehicle. If the fee deducted from the eWallet is adequate, the car is permitted passage; however, if the fee deducted is insufficient, the user is given the option to pay the remaining amount manually, after which the vehicle is granted passage. The automatic system then updates the database with all of these transaction details.

USECASE DIAGRAM:

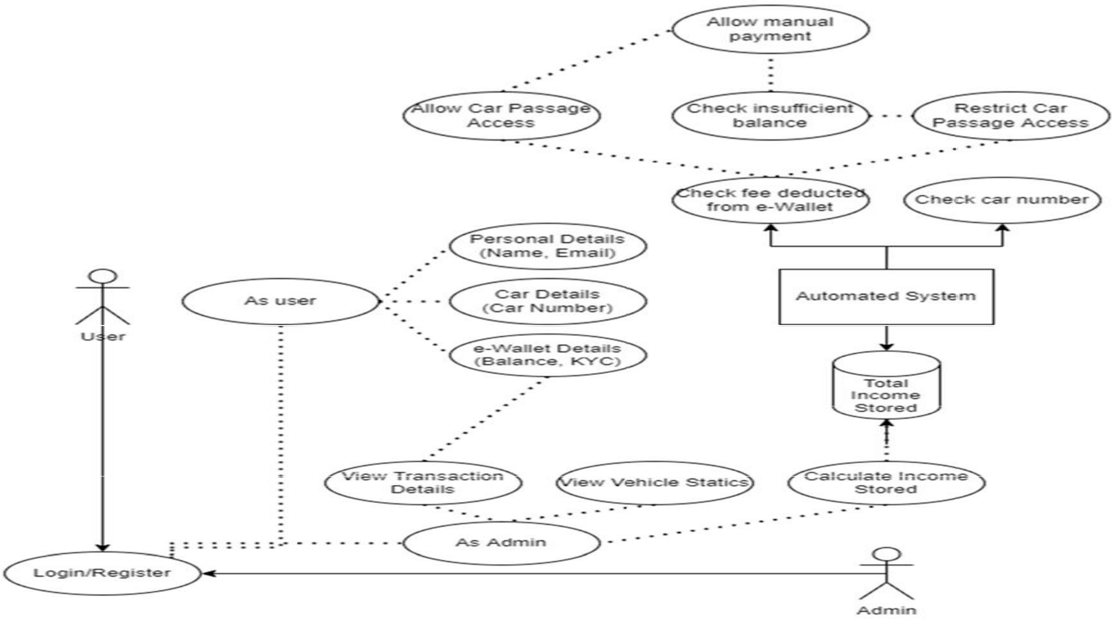


Fig 8.1: USE-Case Diagram for an Online Toll Plaza System

STATE DIAGRAM:

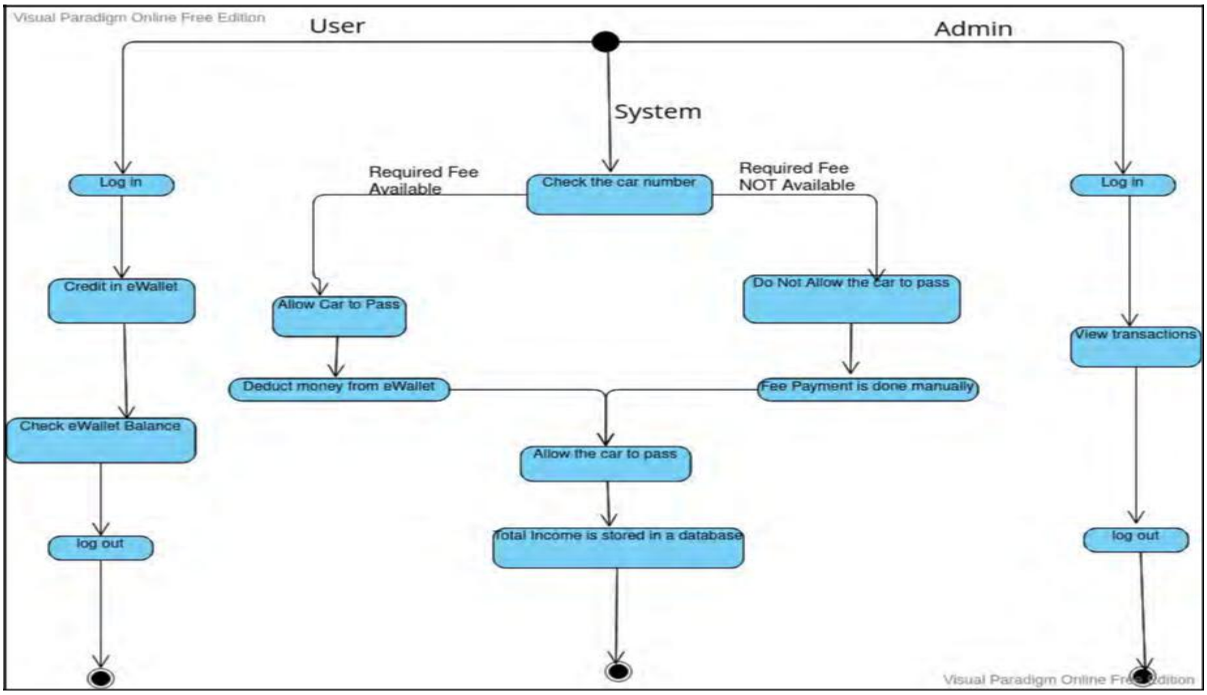

Visual Paradigm Online Free Edition
User
Admin
System
Log in
Required Fee Available
Check the car number
Required Fee NOT Available
Log in
Credit in eWallet
Allow Car to Pass
Do Not Allow the car to pass
View transactions
Deduct money from eWallet
Fee Payment is done manually
Check eWallet Balance
Allow the car to pass
log out
Total Income is stored in a database
log out
Visual Paradigm Online Free Edition

Fig 8.2: State Diagram for an Online Toll Plaza System

COLLABORATION DIAGRAM:

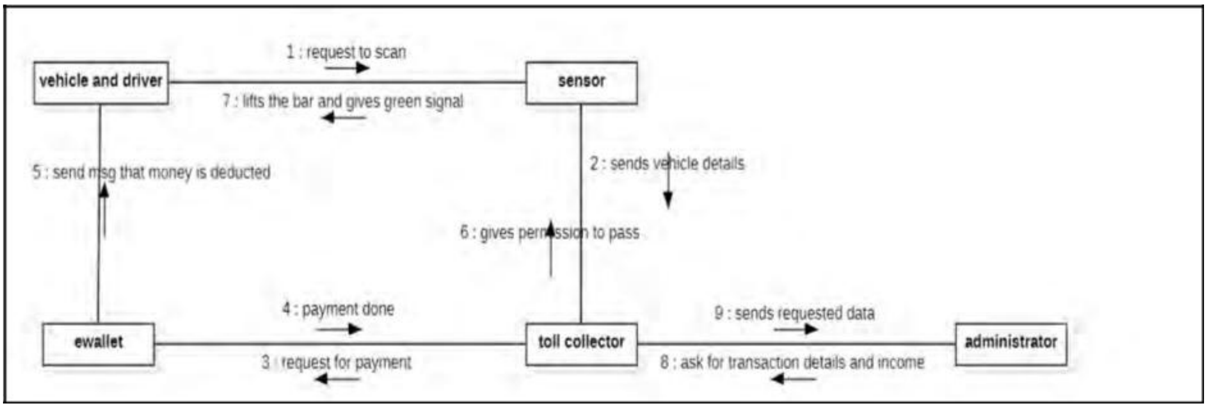

1 : request to scan
vehicle and driver
sensor
7 : lifts the bar and gives green signal
5 : send msg that money is deducted
2 : sends vehicle details
6 : gives permission to pass
4 : payment done
9 : sends requested data
ewallet
toll collector
administrator
3 : request for payment
8 : ask for transaction details and income

Fig 8.3: Collaboration Diagram for an Online Toll Plaza System

PACKAGE DIAGRAM:

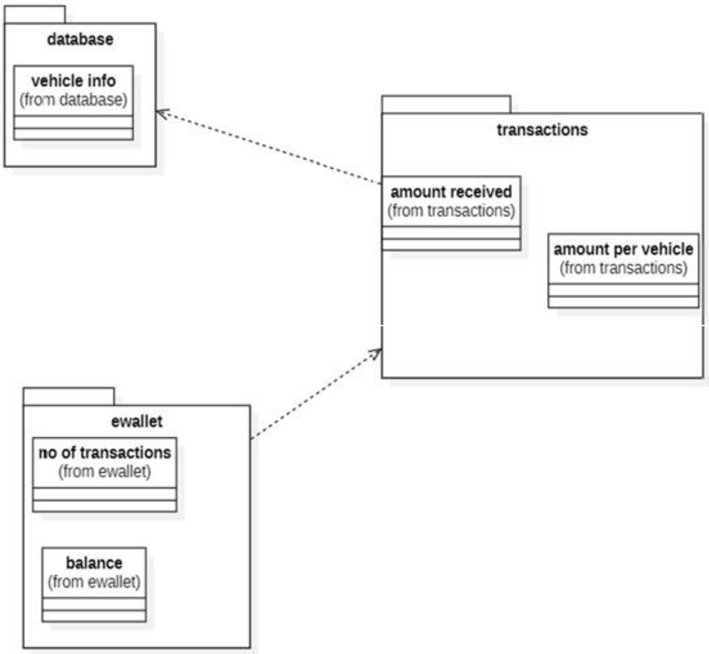


Fig 8.4: Package Diagram for an Online Toll Plaza System

COMPONENT DIAGRAM:

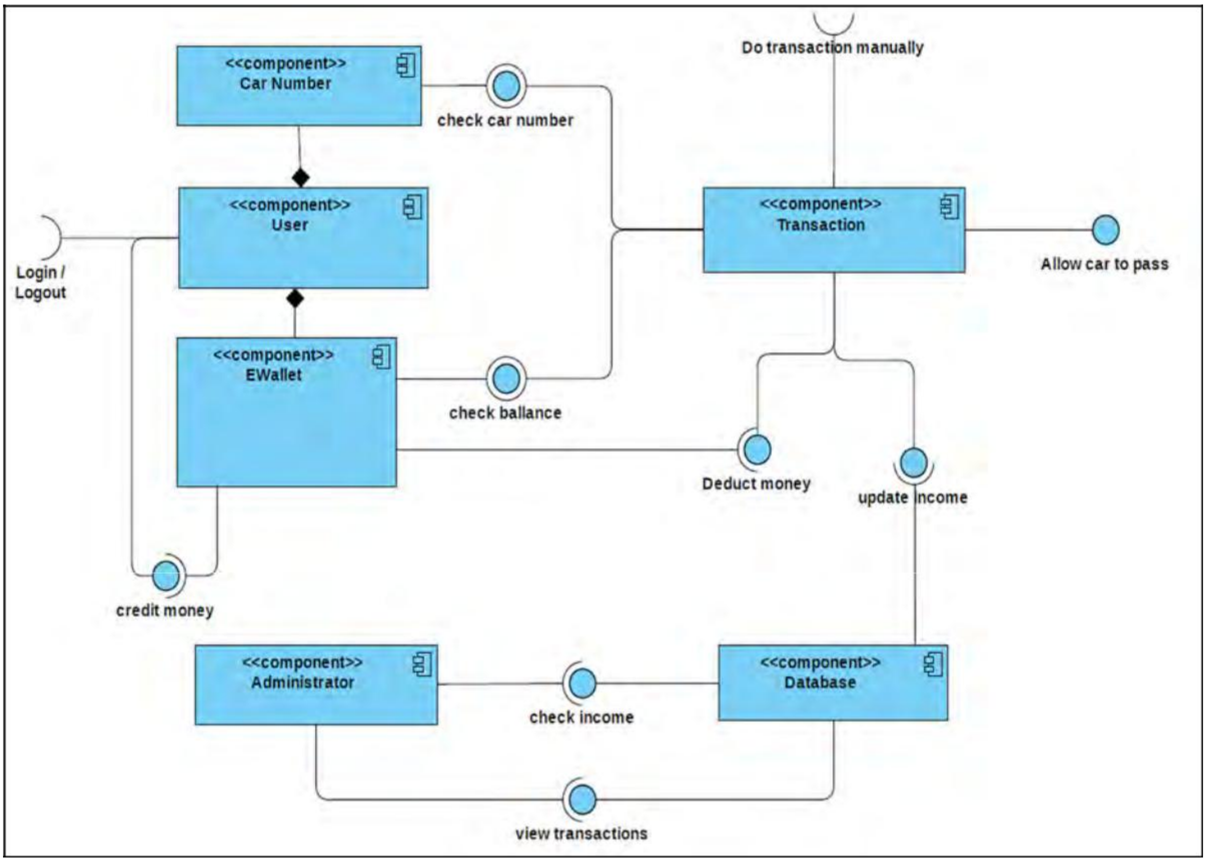


Fig 8.5: Component Diagram for an Online Toll Plaza System

09 TO DRAW THE UML DIAGRAM FOR AN ONLINE HEALTH MONITORING SYSTEM

THEORY:

“Smart Online health monitoring or Online patient monitoring system (OHMS) is a promising technology to enable patient monitoring outside the conventional clinical system, i.e., the patient can be monitored remotely. Consequently, such system increases the access to care the patients and decreases the delivery cost related to healthcare.

Typically, in OHMS, two types of users are there – doctors and patients. Different physiological parameters of the patient are monitored (using sensors), and the monitored data is stored in a server. The stored data is accessible from anywhere through user authentication. On the other hand, doctors can check the health status of a patient registered with the doctor. Therefore, the doctor can only access the physiological data of a patient if and only if he/she is registered with the doctor. Depending on the monitored values, adequate measures can be taken by the doctors. The patient can also view his/her health status. For authenticity, both the users need to login into the system.

INPUTS:

User Information
Doctor – (Name, Email Address, Password)
Patient – (Name, Email Address, Password, DOB)
Predefined Sensors (such as temperature, blood pressure and heart rate)

OPERATIONS:

Patient

- Log-In
- View health status
- Ask doctor(s) to consult
- Payment (Consultancy Fee)
- Log-Out

Doctor

- Log-In
- Monitor health status of registered patients to him/her ○ ◦ Ask patient(s) to consult
- Log-Out

OUTPUTS:

Display health status
Consult with doctors/patients

CONSTRAINTS:

All users MUST register themselves into the system.
Age of patients automatically calculated using DOB information.
Doctors can access the health information of patients registered to them only.
Nobody can modify the stored information.
A patient can only view his/her health status.

EXPLANATION:

To begin, we have three characters: the patient, the doctor, and the administrative staff. The admin's job is to manage and update the patient and doctor profiles, verify payment status, and allocate patients to physicians. As a doctor, you have access to the patient's health state via health sensors, which offer real-time data on the patient's heart rate, temperature, pressure, oxygen level, and other factors. The doctor can get a list of patients and verify their own schedule. By checking the physicians' schedules or which doctors are available, the patient can request a doctor consultation and arrange an appointment after paying. The patient has access to the doctor's most recent health report.

USECASE DIAGRAM:

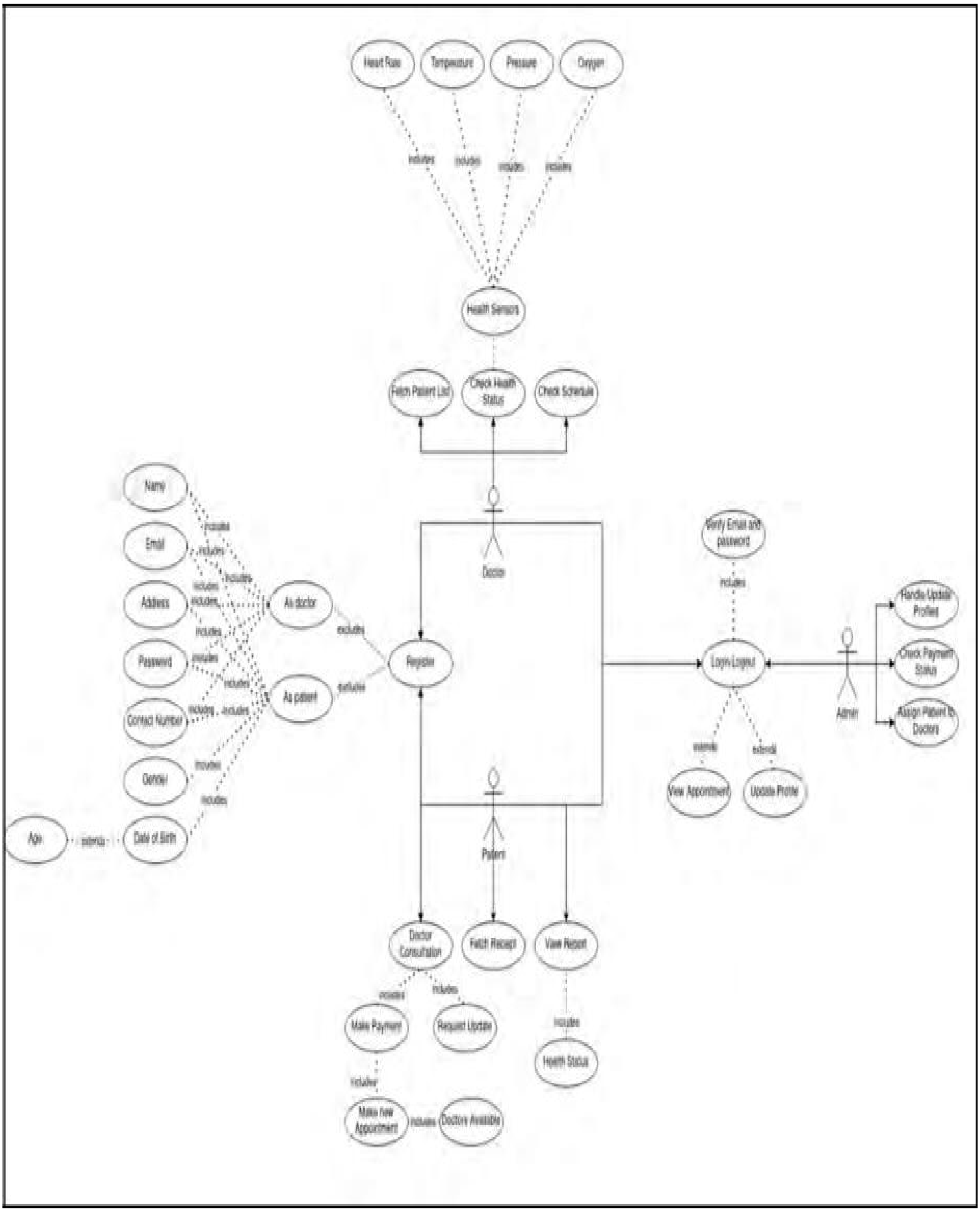


Fig 9.1: Use-Case Diagram for an Online Health Management System

SEQUENCE DIAGRAM:

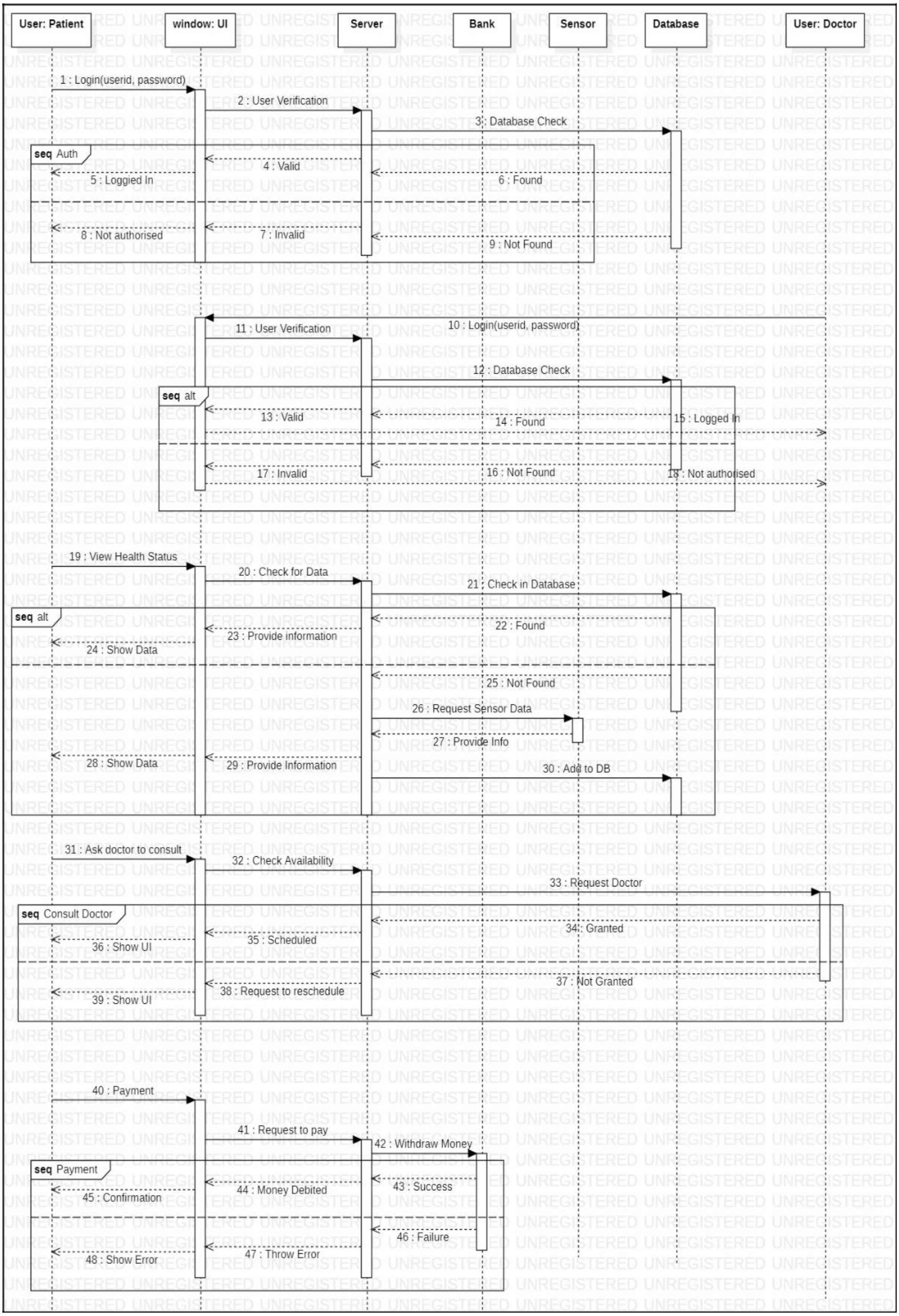


Fig 9.2: Sequence Diagram for an Online Health Management System

STATE DIAGRAM:

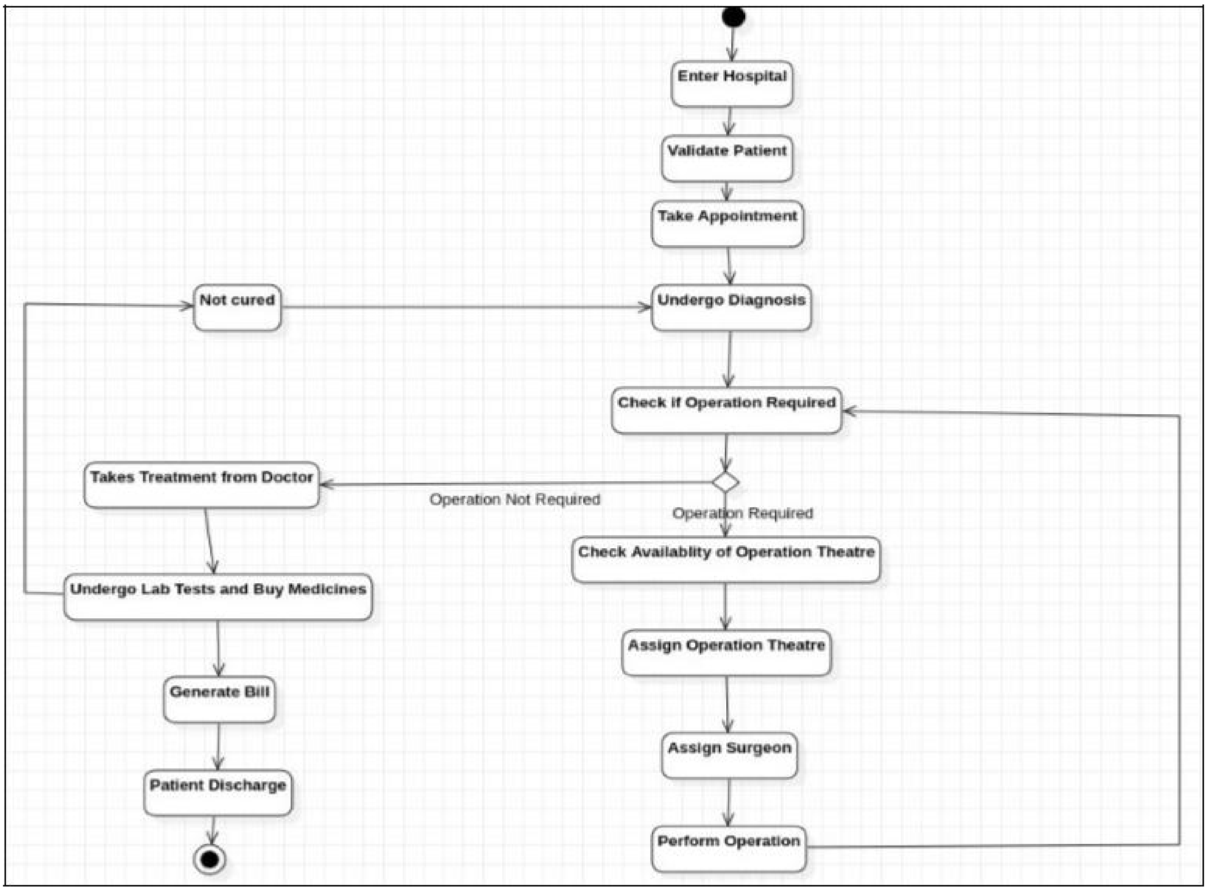


Fig 9.3: State Diagram for an Online Health Management System

COLLABORATION DIAGRAM:

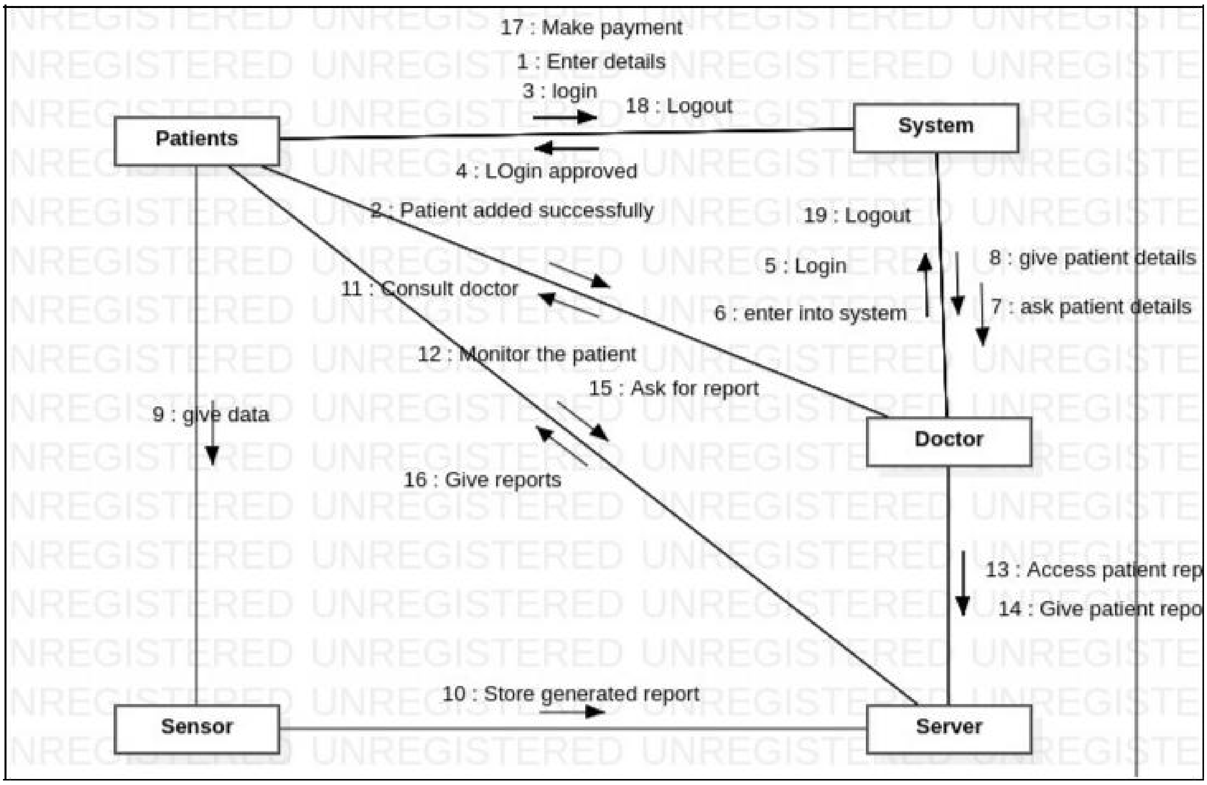


Fig 9.4: Collaboration Diagram for an Online Health Management System

PACKAGE DIAGRAM:

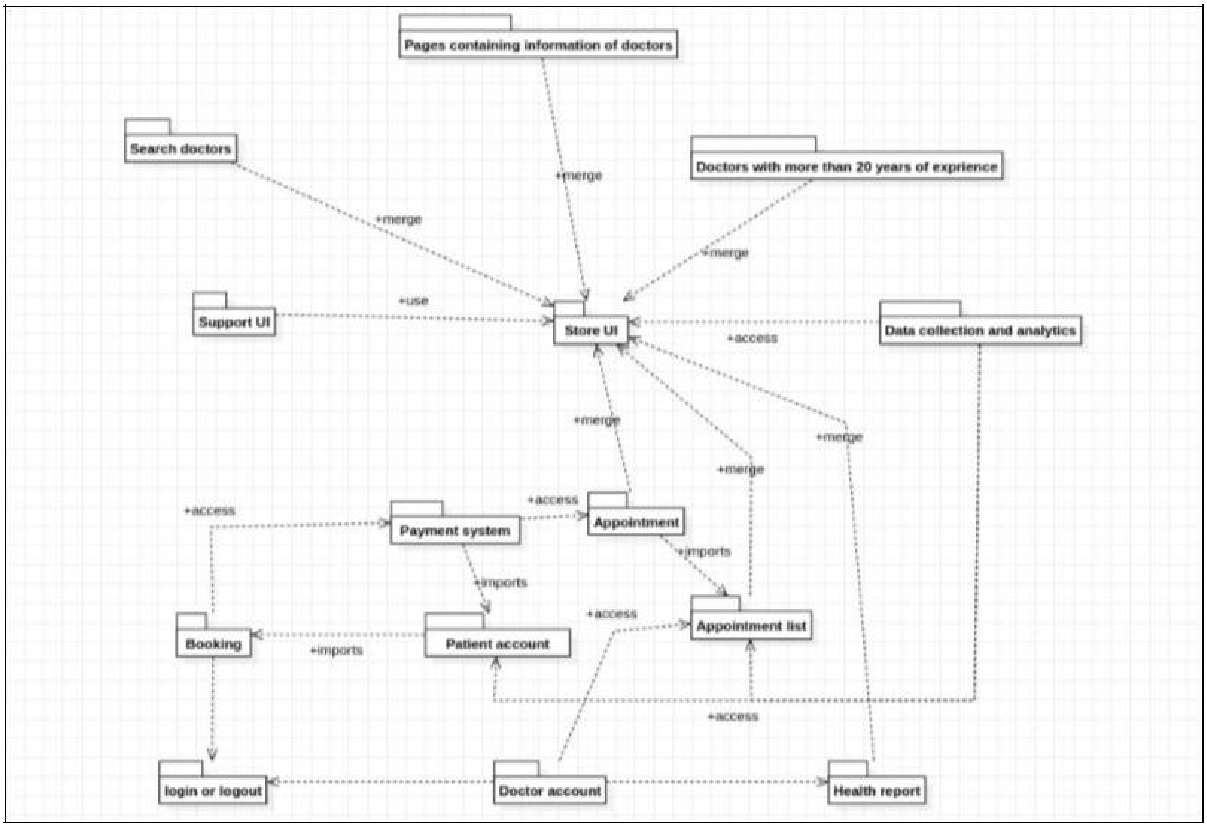


Fig 9.5: Package Diagram for an Online Health Management System

COMPONENT DIAGRAM:

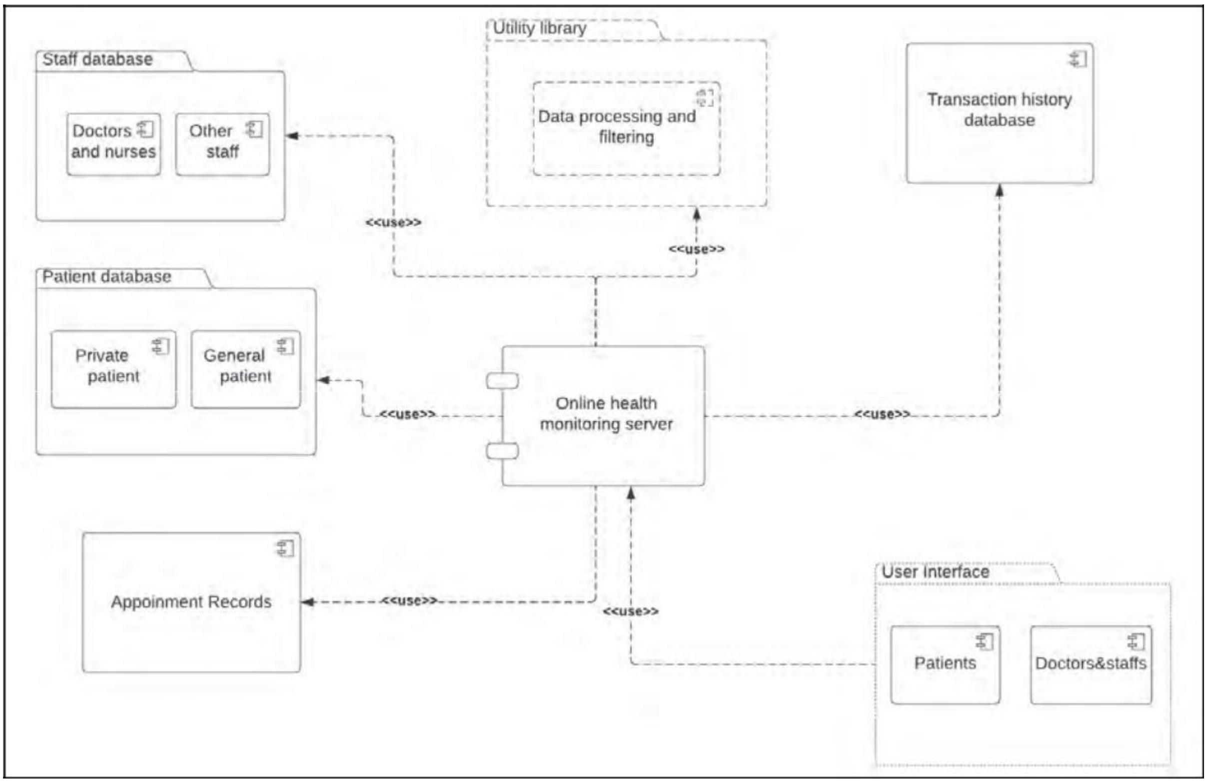


Fig 9.6: Component Diagram for an Online Health Management System