

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.



LAB REPORT on

OBJECT ORIENTED MODELLING AND DESIGN

Submitted by

PRATHIKSHA KAMATH(1BM19CS118)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING



B.M.S. COLLEGE OF ENGINEERING

(Autonomous Institution under VTU)

BENGALURU-560019

May-2022 to July-2022

**B. M. S. College of Engineering,
Bull Temple Road, Bangalore 560019**
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering



CERTIFICATE

This is to certify that the Lab work entitled "**OBJECT ORIENTED MODELLING AND DESIGN**" carried out by **PRATHIKSHA KAMATH(1BM19CS118)**, who is bonafide student of **B. M. S. College of Engineering**. It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2022. The Lab report has been approved as it satisfies the academic requirements in respect of an Object Oriented Modelling and Design - **(20CS6PCOMD)** work prescribed for the said degree.

Name of the Lab-Incharge
Designation
Department of CSE
BMSCE, Bengaluru

Dr. Nandini Vineeth
Assistant Professor
Department of CSE
BMSCE, Bengaluru

Index Sheet

Sl. No.	Experiment Title	Page No.
1	College Information System	4
2	Hostel Management System	10
3	Stock Management System	15
4	Coffee Vending Machine	21
5	Online Shopping System	27
6	Railway Reservation Machine	33
7	Graphics Editor	39

Course Outcome

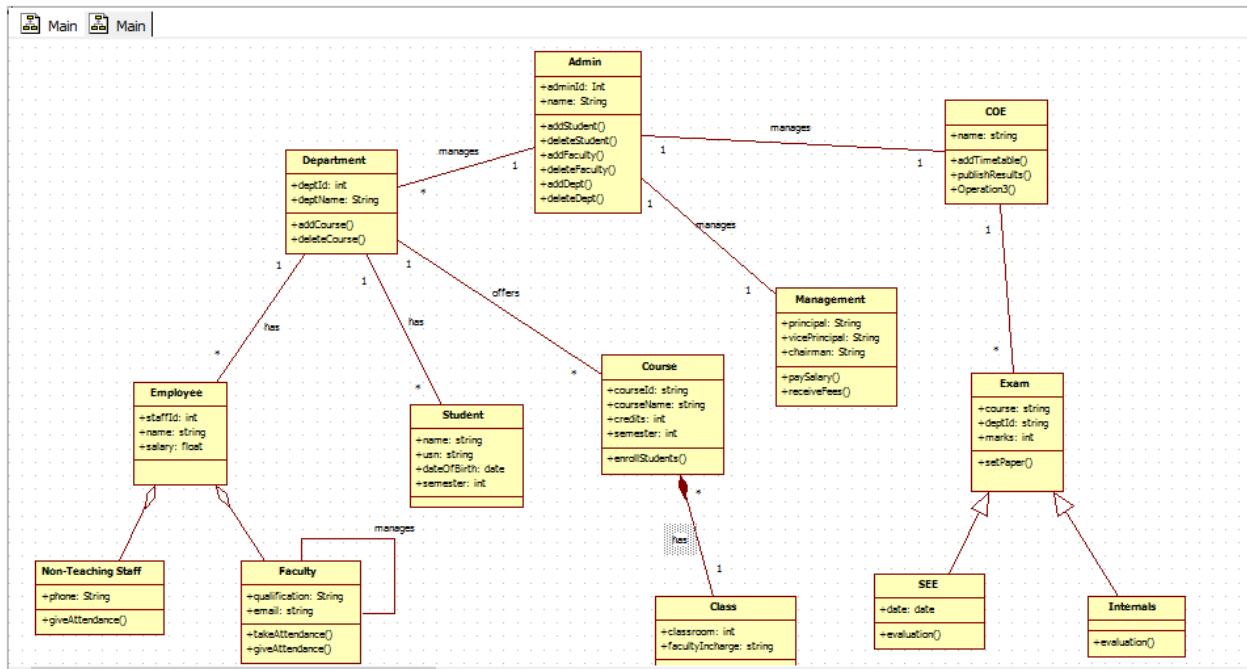
CO4	Ability to conduct practical experiment to solve a given problem using Unified Modeling language.
-----	---

LAB 1:College Information System

1.1 Software Requirement Specification

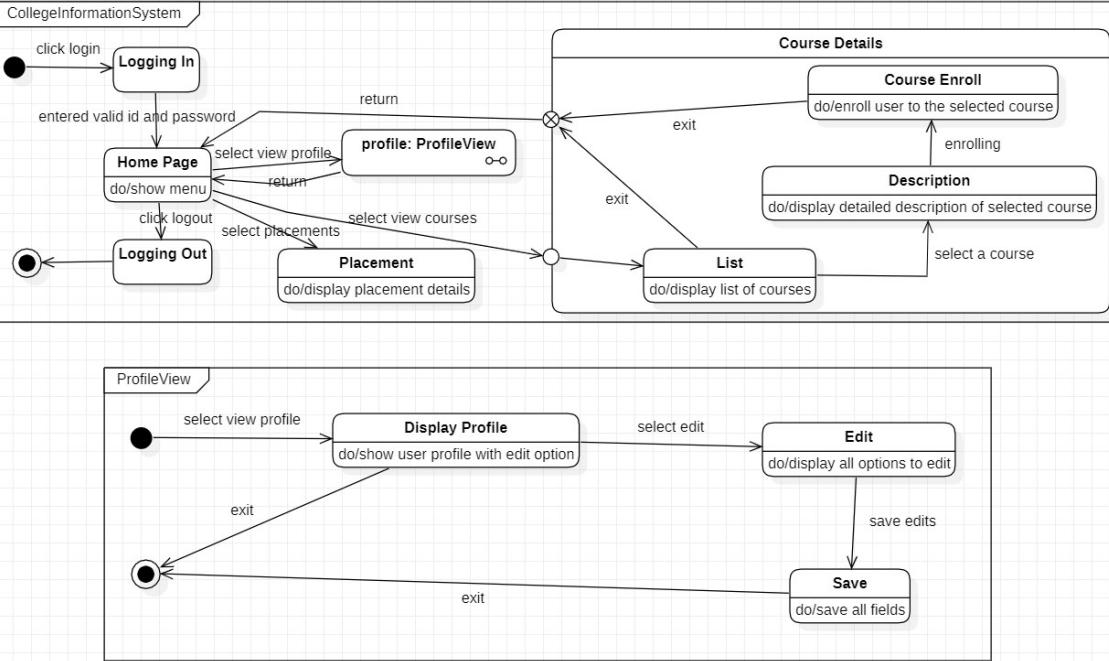
- College information system has an admin who manages the staff ,students and department.
- Admin can view and modify the student's records like student's profile, attendance, fee, results, and details of teachers and other employees in college, their personal information and their attendance for their salaries.
- In this system, user authentication will be done by login by username and password and classified by user type.
- Staff in college teach more than one course to many students and the staff who are teachers conduct examinations for students of the college.
- The students of the college register themselves in the department and for the courses they are interested in and join the college by taking admission and following all the admission procedures.
- There are different types of examinations conducted by the college for the students. Internalsand semester end examinations are two of them.
- Every course has a name and its unique name. Every course has different subjects and every subject has its own unique name.
- Department will provide the details about departments within a college with their name and every department have its Department name.

1.2 Advanced Class Diagram



- Admin is associated with the Department, Management and COE. One admin manages Management, COE and many departments in college.
- Department is associated with Employee, Student and Course. Department has many employees, many students and offers many courses.
- Employee class is generalized into non-teaching staff and faculty. Faculty is self associated.
- Because the class cannot exist without courses ,the class is composed inside courses.
- Exam is associated with COE. One COE manages many exams.
- Exams are generalized into SEE and Internals as they use all the attributes of Exam.

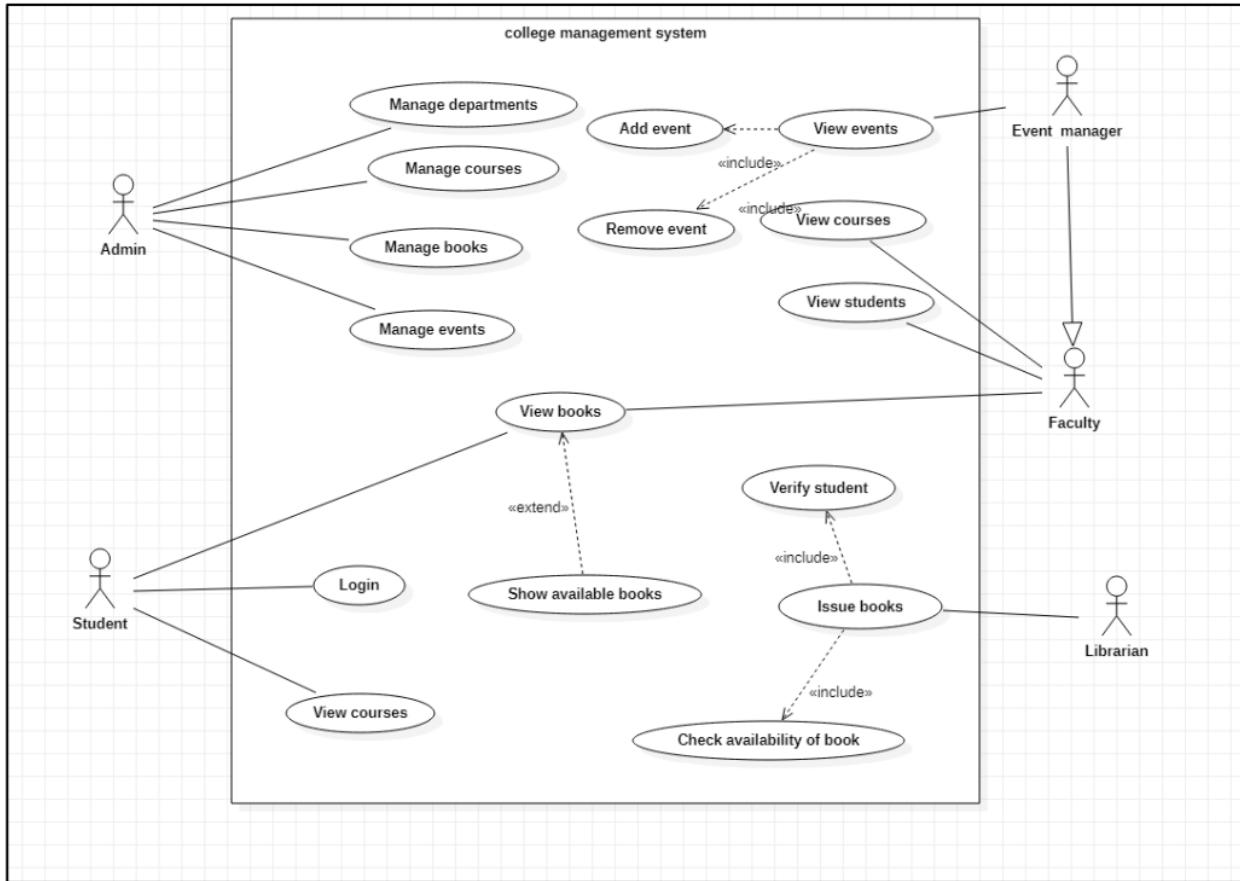
1.3 Advance State Diagram



The advanced state diagram contains one nested state and one submachine, which on successful login shows the course details and profileView procedure of the student. It contains initial state and termination state with Courses as a nested state including the required simple states. It also has a submachine state named ProfileView with initial, termination state along with simple states; Display profile, Edit, Save.

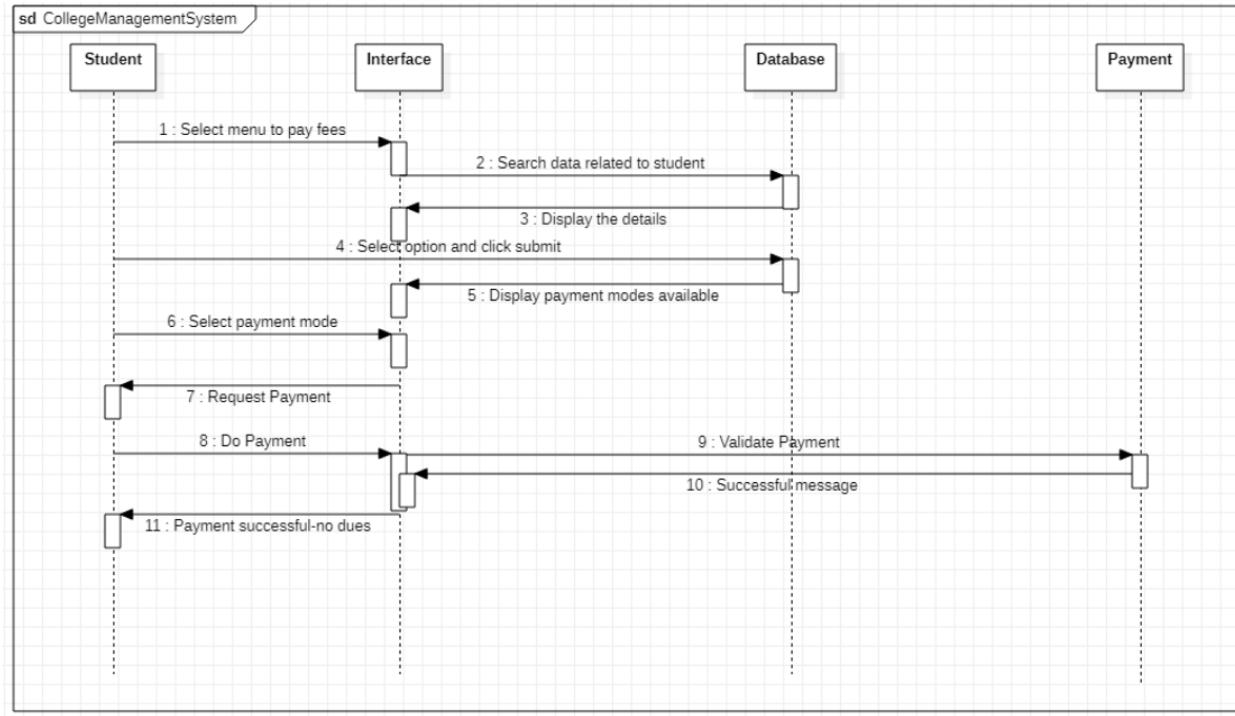
1.4 Advanced Use case Diagram

The advanced use case diagram has extra functionalities which includes extends, includes and generalization. The show available books use case extends view books use case, view events use case includes add events and remove events, issue books use case includes verify student and check availability of book.



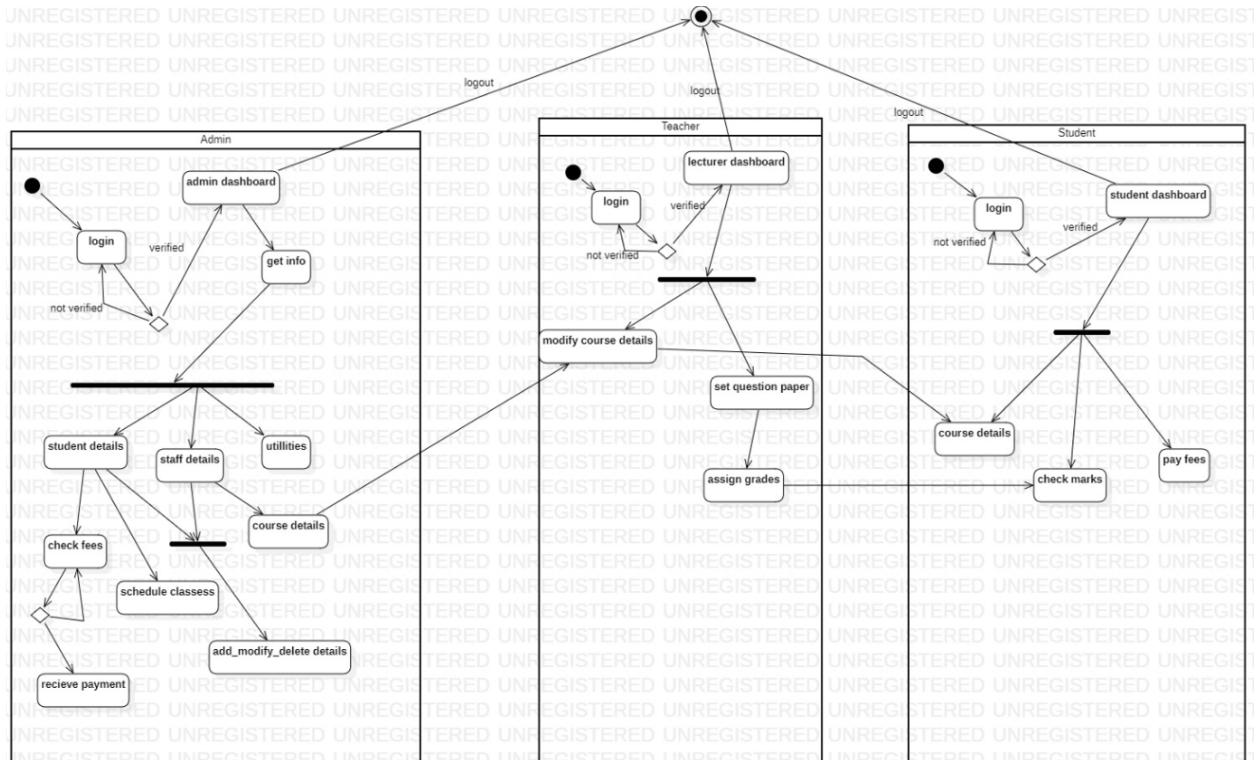
1.5 Advanced Sequence Diagram

The lifeline is the dotted line and the rectangles represent the period of time the object is executing and is hence called activation. The recursive function of verify is shown by double activation rectangle of verify payment and successful message. The scenario shown here is student logging in and paying fees.



1.6 Advanced Activity Diagram

There are three swimlanes for Admin, Teacher and Student. On successful login, they can see their respective dashboard and on logout the activity is terminated. The admin can check and modify all details. The teacher can set question paper and assign grades. The student can pay fees.



LAB 2: Hostel Management system

2.1 Software Requirement Specification

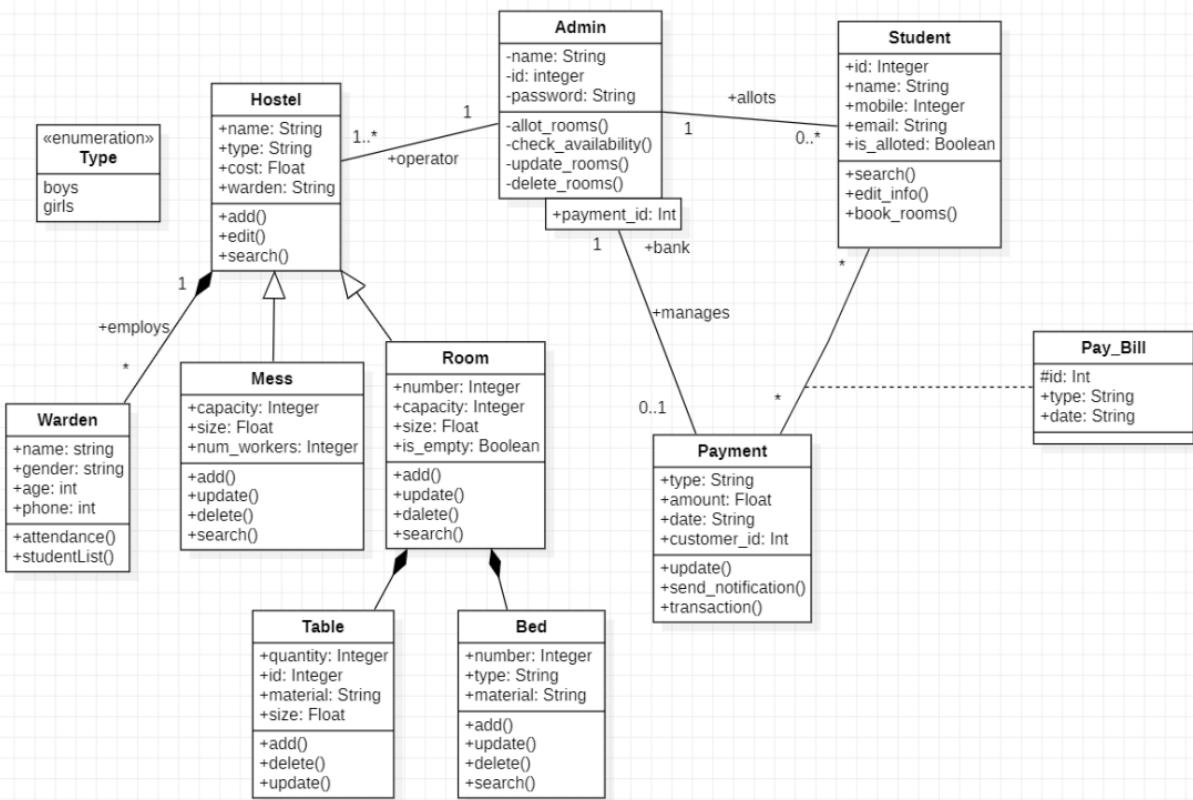
- Hostel management system has an admin who manages the hostel, allot-ees and payment methods. The admin will allocate a room to students according to the section or class. The admin will also keep track of the payment made by the student/allot-ees.
- As the student's course is over they will vacate their rooms. So it is required for the administrator to remove their records from the database tables.
- The allot-ees makes payment according to the bill generated which have the attributes bill number, type and date.
- The details of the students staying in the hostels like name, place, address, contact details maintained in the database.
- The hostel is categorized into two types i.e boys and girls hostel. Each hostel type has different costs, warden and name.
- A hostel is made up of mess and rooms. A mess account will also generate. This account has the mess status of the whole month. On the basis of this account monthly charges of a student will be defined.
- The hostel management system will allow renewing the student's registration every year.

2.2

Advanced

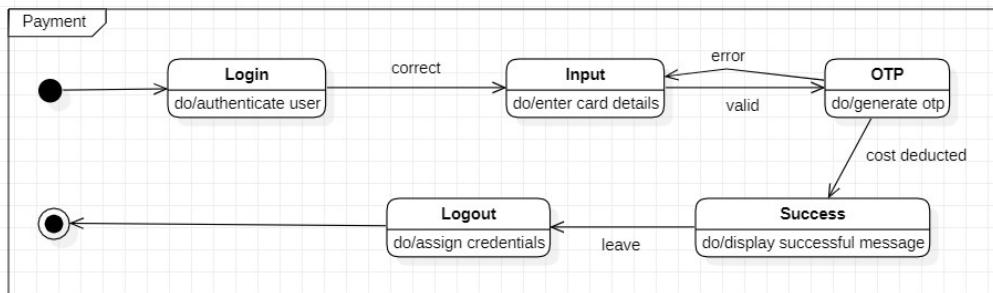
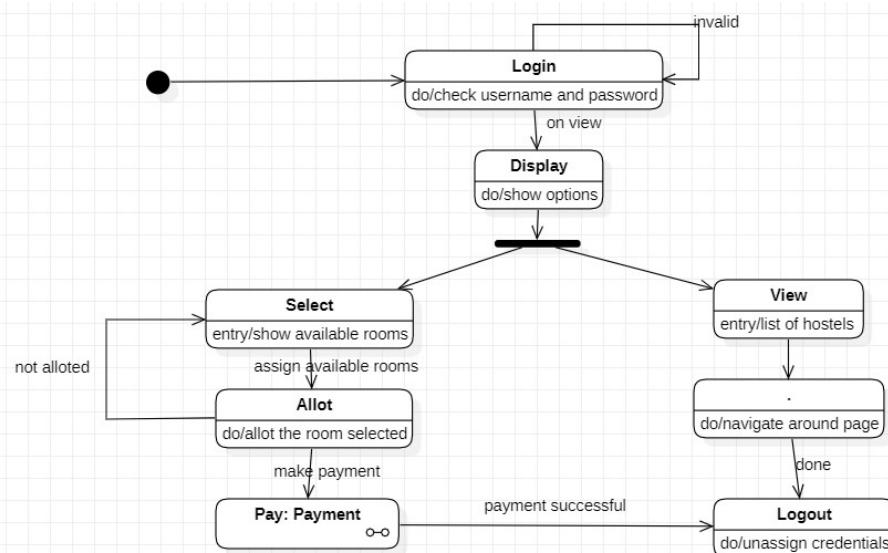
Class

Diagram



- Here we have an Admin class that has an association with Student, Hostel and payment class. One admin can add zero or more students and can control many hostels.also admin manages payments.
- The hostel class is made up of the composition of Warden class as warden class cannot exist without a hostel.Many hostels can have one or more warden.Also hostel class is generalized into mess and room class.
- In hostel type is an enumeration made up of boys and girls as hostel can be girls hostel or boys hostel.
- The Students make payment according to the bill generated which have the attributes bill number,type and date.
- Room class is made up of composition of Table and Bed class as without room these classes cannot exist

2.3 Advanced State Diagram

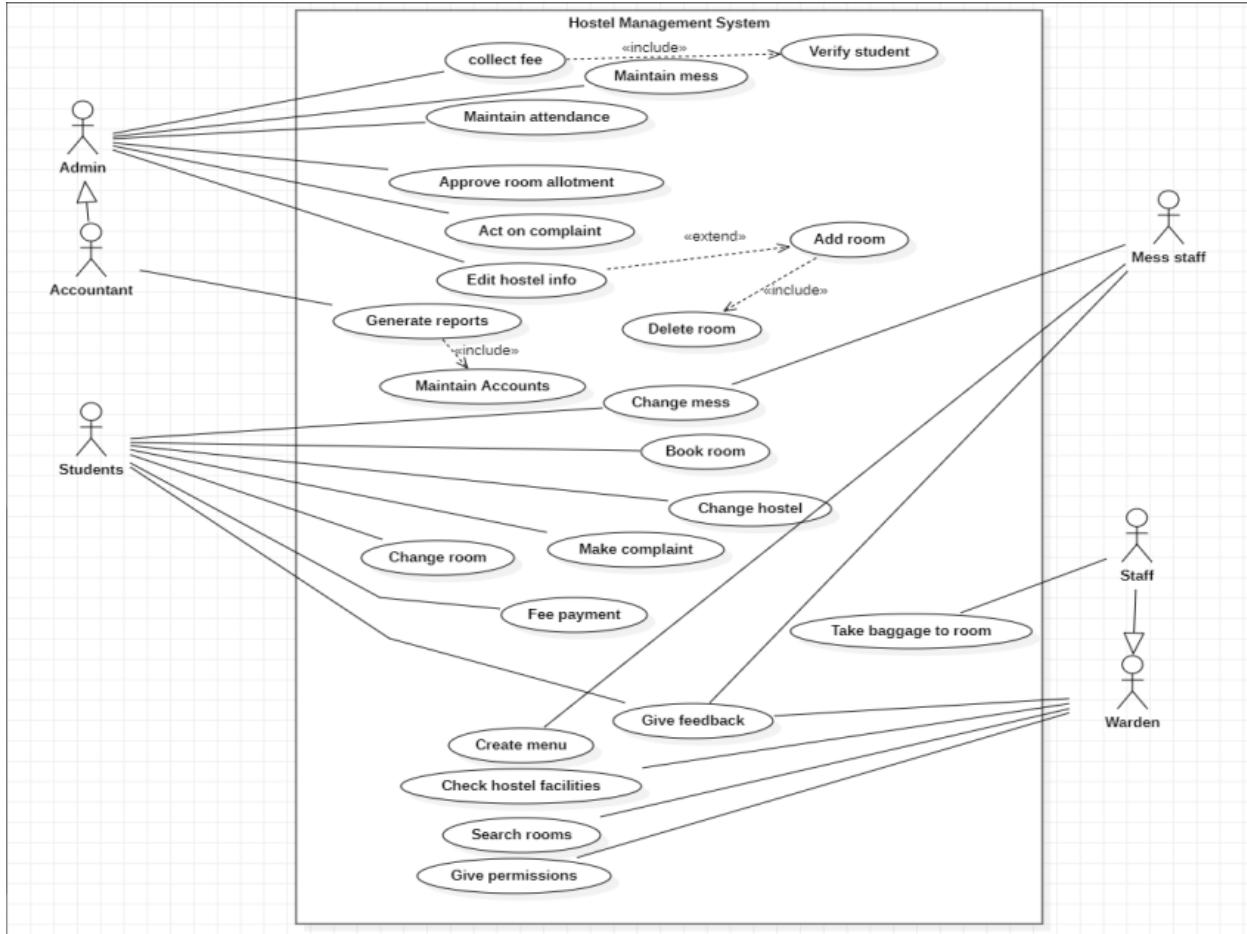


The above state diagram gives the movement of states in allotting a room to a student. The admin allots rooms for students. The admin first logins to the database ,which displays a set of options.the admin then chooses to allot rooms and finds the availability for rooms. If rooms are available then the admin allots room to the student and when

successful the student makes the payment. If no rooms are available,a message is displayed and control goes back to the displaystate.

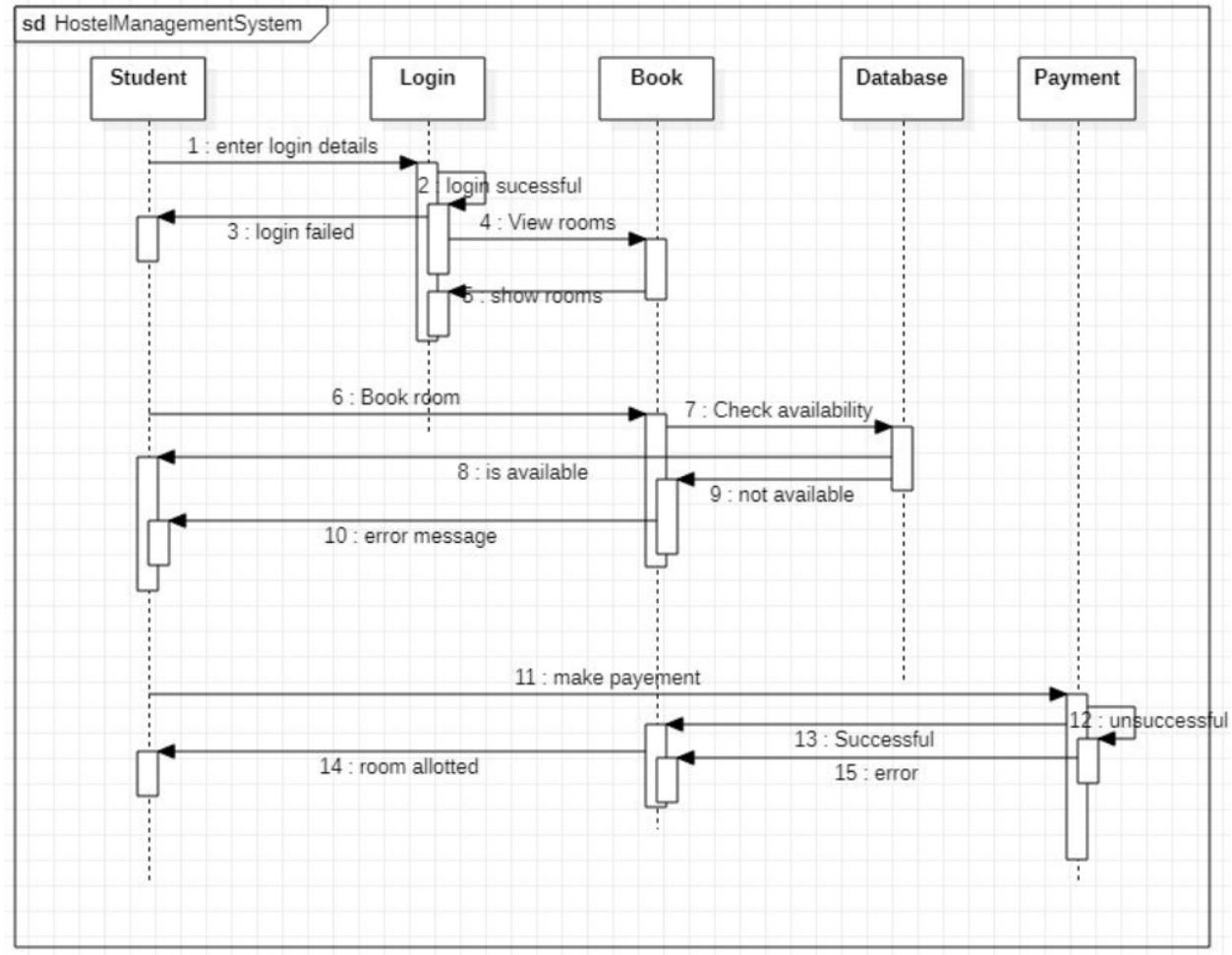
2.4 Advanced Use case Diagram

The advanced use case diagram has extra functionalities which includes extends, includes and generalization. The edit hostel info use case extends add room use case, collect fee use case includes verify student, add room use case includes delete room use case.



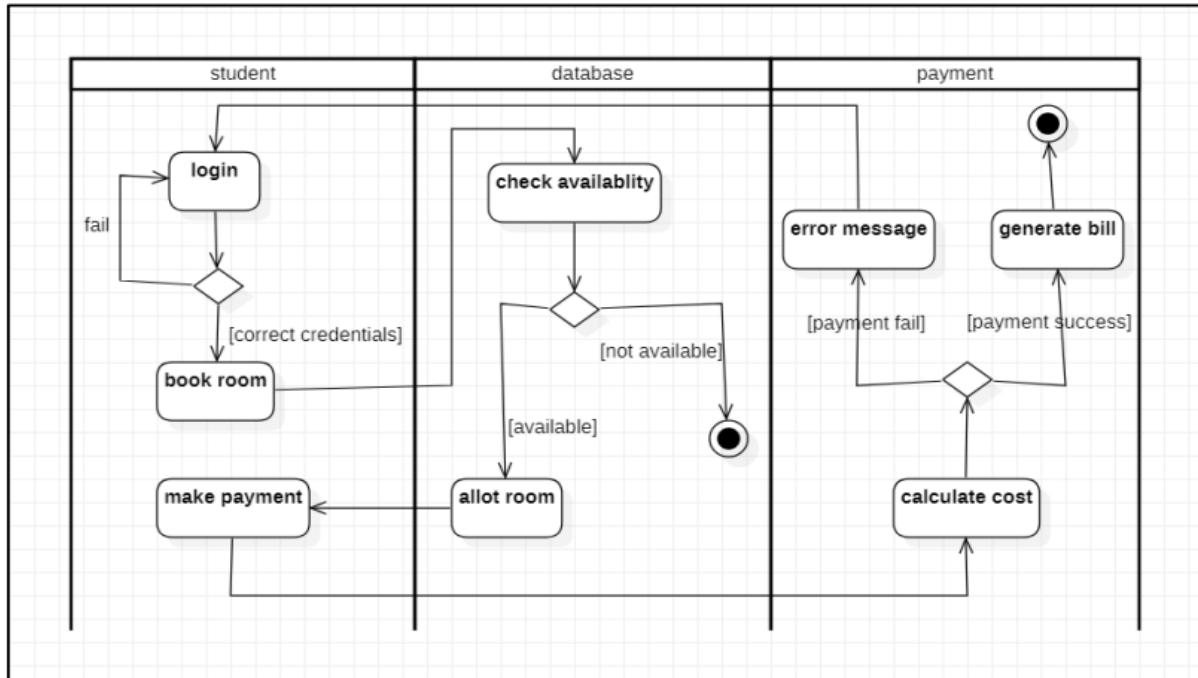
2.5 Advanced Sequence Diagram

The lifeline is the dotted line and the rectangles represent the period of time the object is executing and is hence called activation. The scenario shown here is a student logging in into system, booking a room and making payment.If the payment is successful the room is allotted.



2.6 Advanced Activity Diagram

The advanced activity diagram starts from initiation and then in the student swimlane, student login activity where a signal is sent to the network for request validation and upon confirmation the control flows to profile and then book room activity. There are three swimlanes namely student, database, payment where validate student, update database and confirm payment respectively. Then the control flows to the home page and then termination activities.



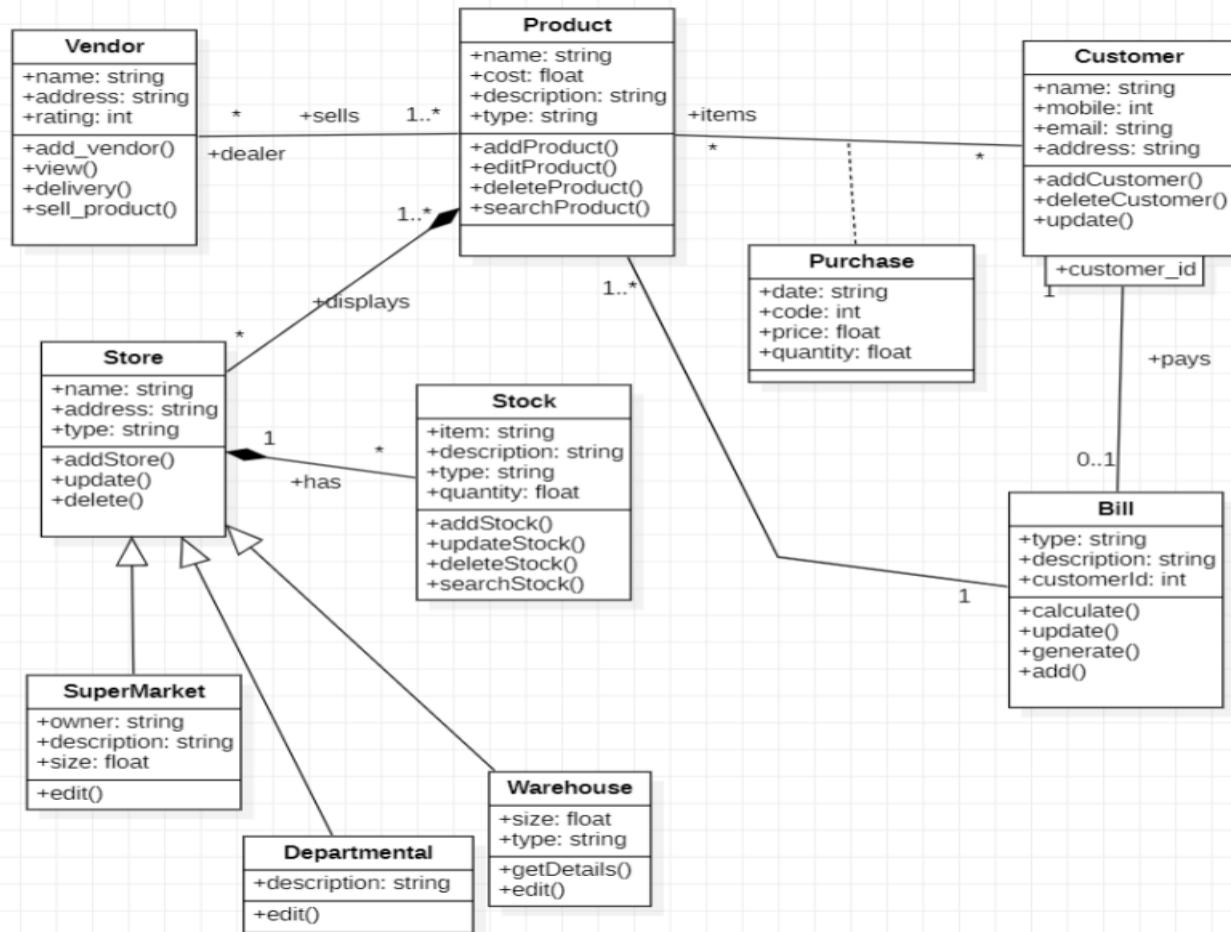
LAB 3:Stock Management System

3.1 Software Requirement Specification

- The customer can purchase one or more products on any day,which will have a code price and quantity.
- The customer will need to pay the bill for the products he or she has purchased.The bill number,type description and customer who is paying the bill is maintained.
- The stock of the products is maintained separately,The stock deals with information about the details of the product that the concern handling.
- Stock consist of details such as the name of the product, id generated, quantity, cost, etc. This information is retrieved during the sales and purchase of a product.
- The vendor deals with the information about the details of the suppliers giving products to the organization.
- Vendors consist of details such as vendor name, address, email id, sales tax number etc. This Information is retrieved when a Purchase is done.
- The products are displayed in stores across the city or world. All the information regarding the store such as store id,name,address and type are used to locate any product.The stores can be of many types.Some of them are departmental stores,supermarkets and warehouses where the products are kept for display.

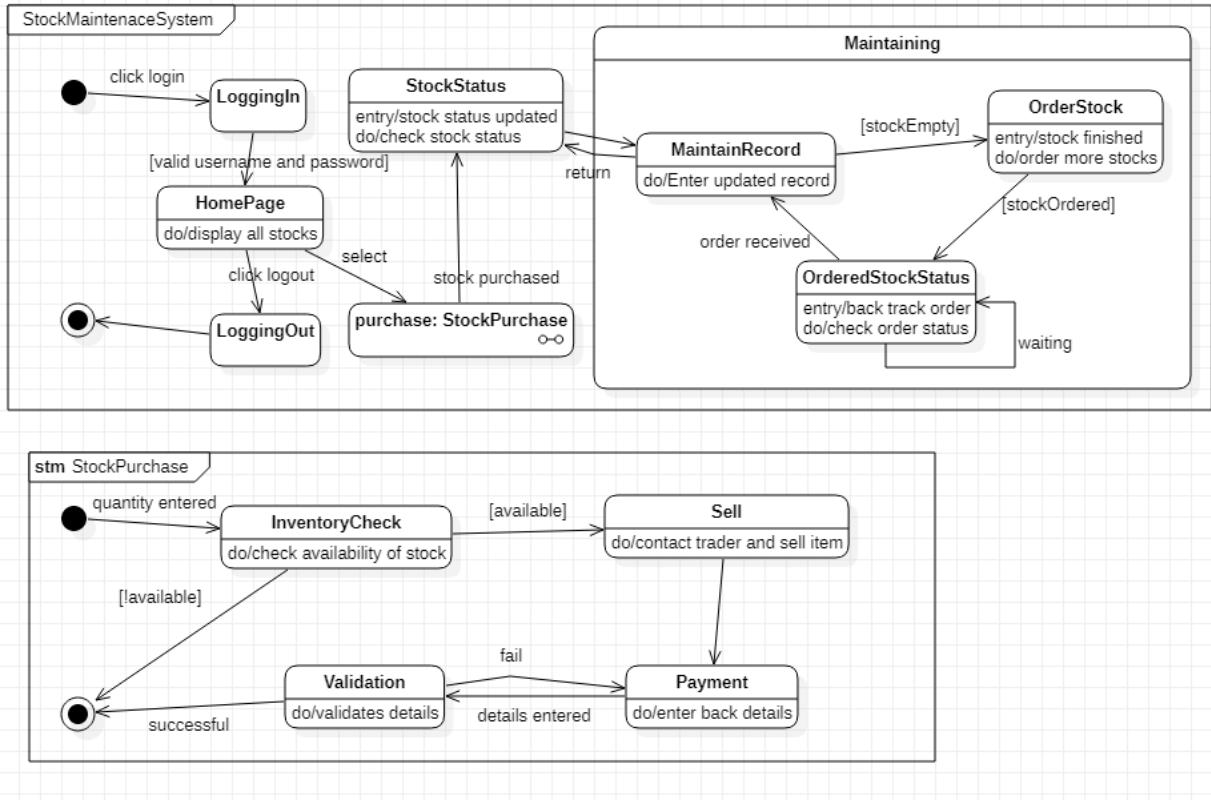
3.2

Advanced Class Diagram



- Vendor deals with a variety of products which are displayed (or stored) in stores.
- A store is generalized to be a superMarket, departmental store or a warehouse.
- A store maintains a stock of the product. The type of association is composition because the stock cannot exist without the store.
- A customer can buy one or more products. We have an association class purchase because of this association.
- To reduce the multiplicity of the association between the customer and bill we have used the qualifier customer id.

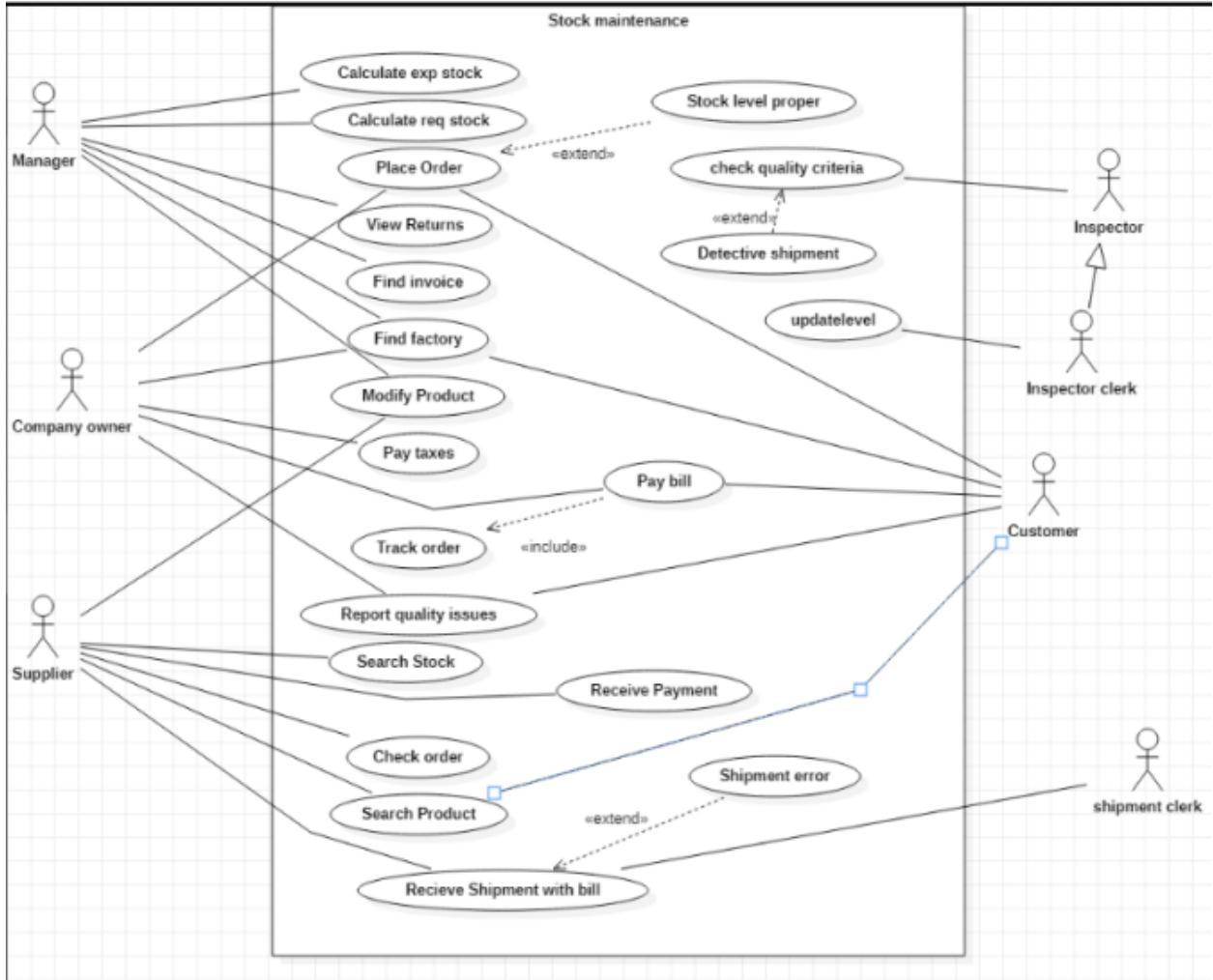
3.3 Advanced State Diagram



The state diagram contains one nested state and one submachine, which on successful login shows the StockStatus details and StockPurchase procedure. It contains initial state and termination state with Maintaining as a nested state including the required simple states. It also has a submachine state named StockPurchase with initial, termination state along with simple states; Inventory check, Sell, Payment, Validation.

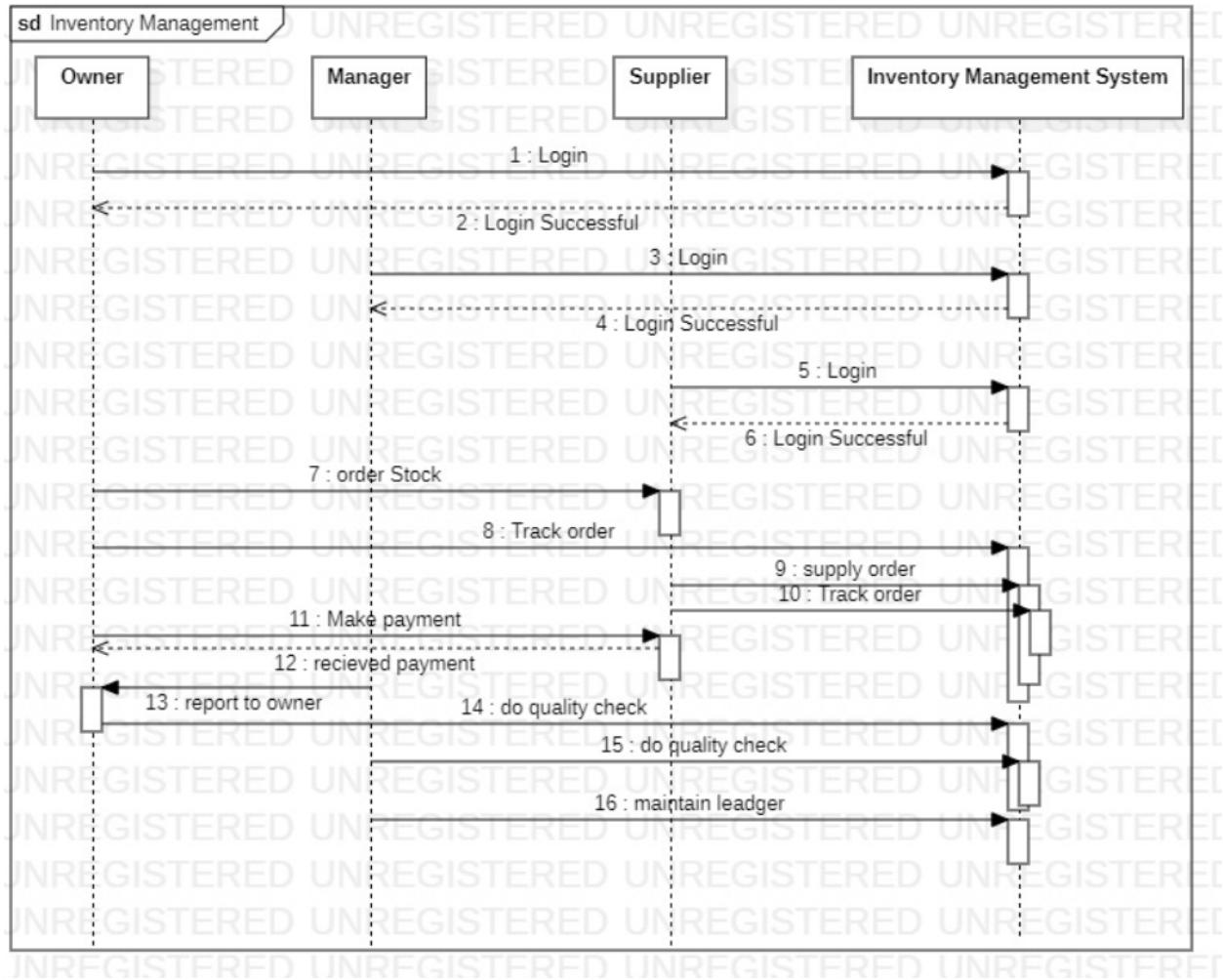
3.4 Advanced Use case Diagram

The advanced use case diagram has extra functionalities which includes extends, includes and generalization. The stock level use case extends place order use case, detective shipment use case extends check quality criteria use case , shipment error use case extends receive shipment with bill use case, pay bill use case includes track order use case.



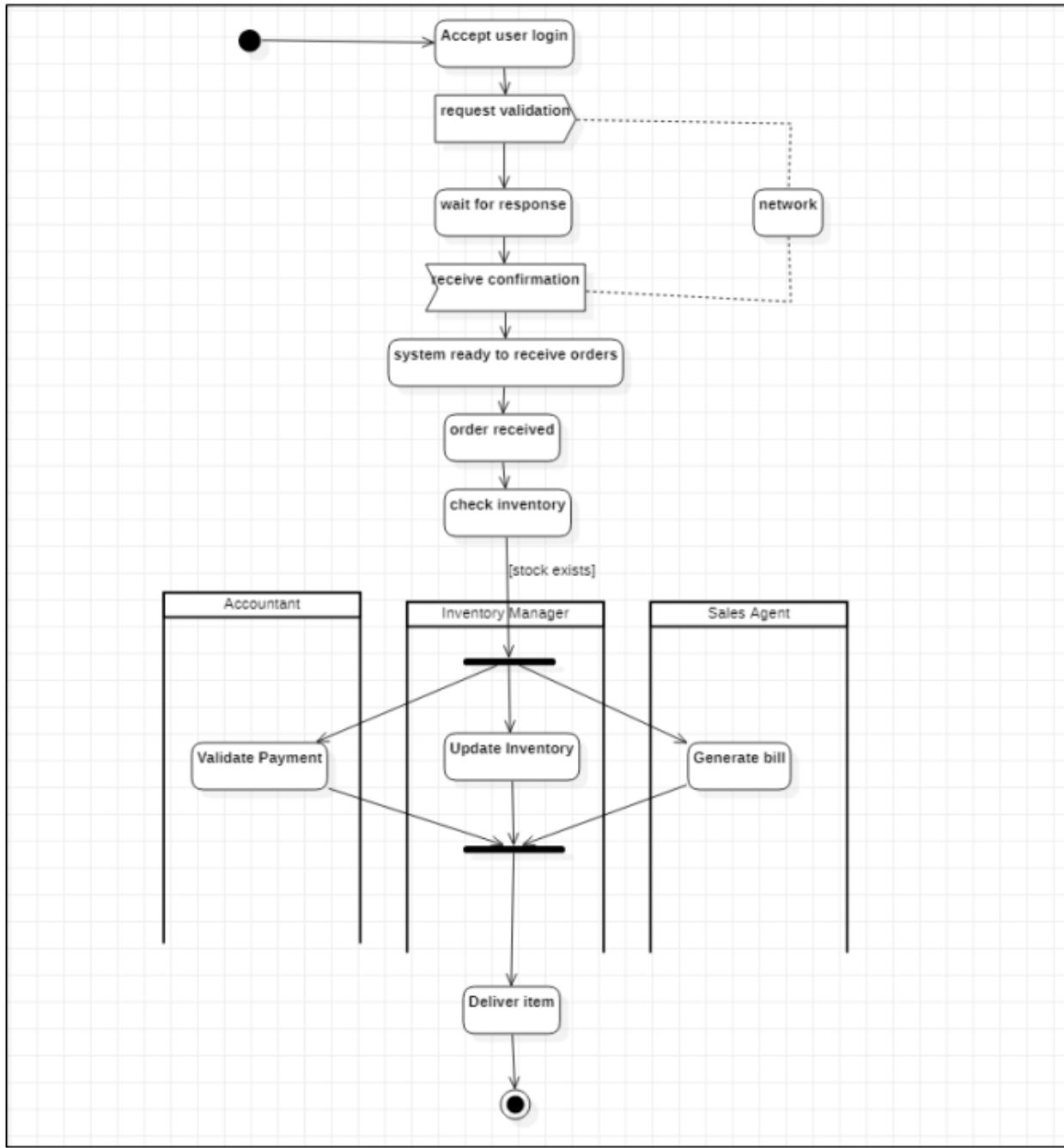
3.5 Advanced Sequence Diagram

The lifeline is the dotted line and the rectangles represent the period of time the object is executing and is hence called activation. Create message signal is used to indicate the display of failure in any failure situation. The scenario shown here is the inventory management owner ordering stocks, making payment and doing quality checks on the received stocks.



3.6 Advanced Activity Diagram

The advanced activity diagram starts from initiation and then user login activity where a signal is sent to the network for request validation and upon confirmation the control flows to order received and then check inventory activity. There are three swimlanes namely inventory manager, accountant and sale agent where update inventory, update payment and generate bill respectively. Then the control flows to the home page and then termination activities.

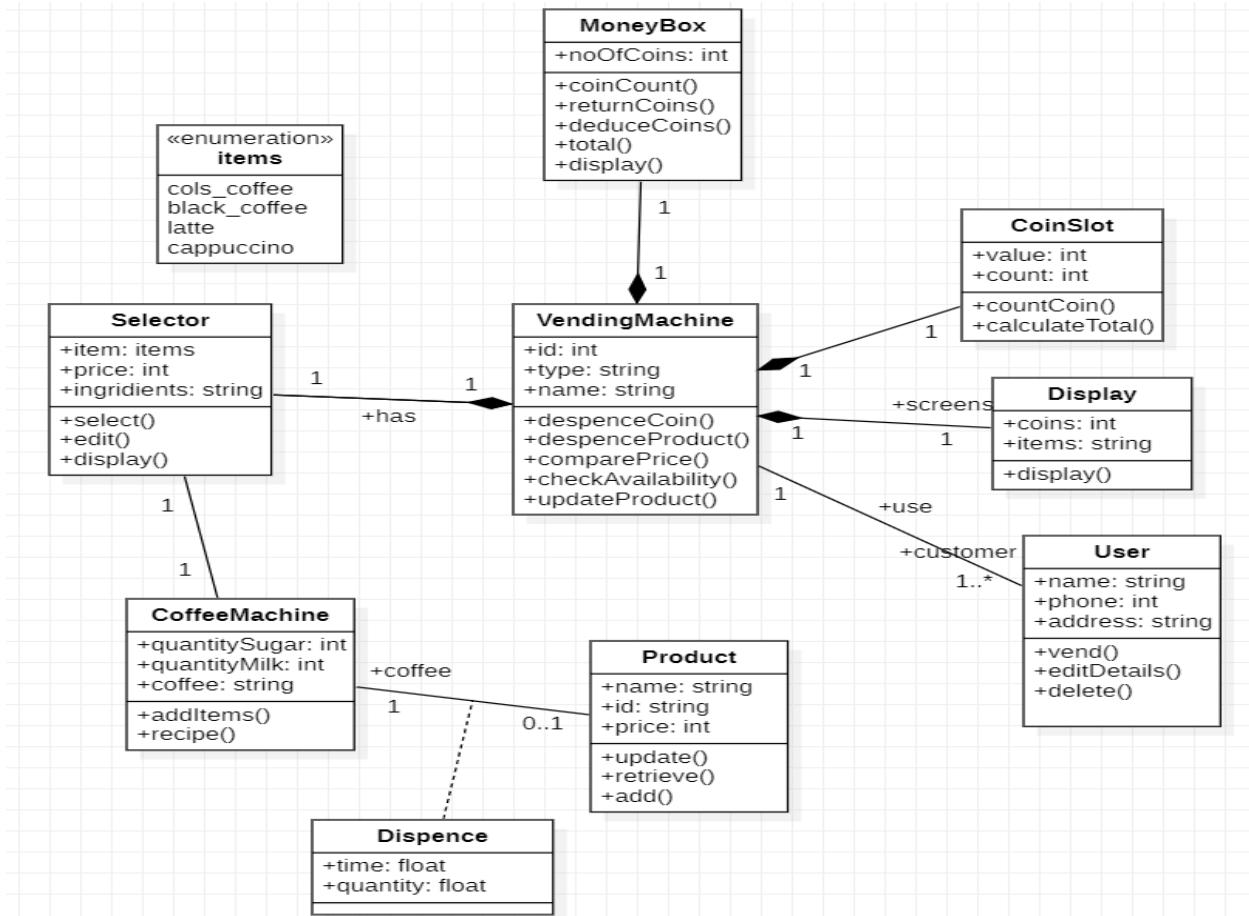


LAB 4:Coffee Vending Machine

4.1 Software Requirement Specification

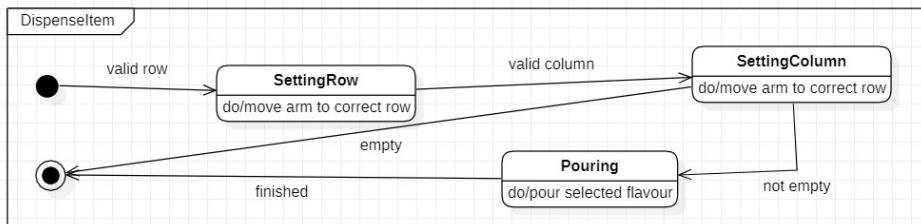
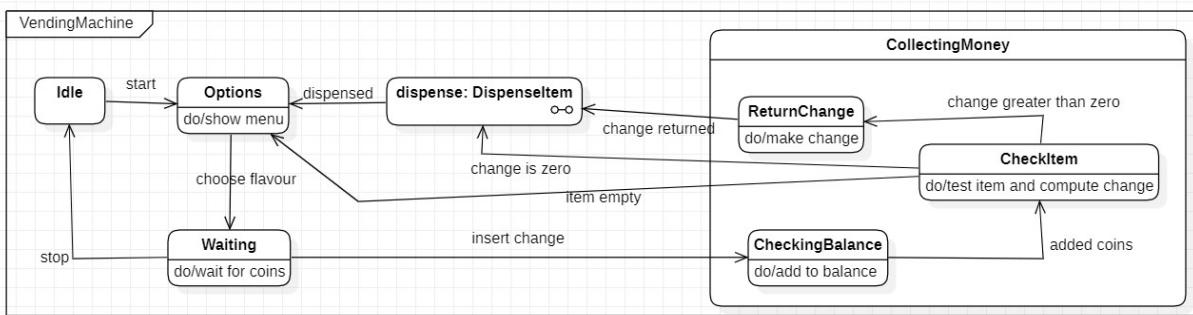
- The vending machine must have a money box, coin slot, display screen and products i.e coffee for the machine to be used.
- The user on selecting a coffee ,the coffee machine must be able to dispense the selected coffee to the user.
- The user shall get an empty cup placed right below the filter.The user shall be able to choose his preferred beverage from the list of options(buttons).
- There must be buttons(start,pause,stop,coffee,tea,milk) for users to interact with the system.
- The user shall be able to purchase one kind of available drink at a time and get back the exact changes if he has put extra money. The user shall be able to quit the dispense of any beverage at any time during the dispensing.
- The system(machine) shall check for properly inserted coins.
- The system shall be able to dispense coffee(or selected beverage) after a coin has been inserted.
- The system must accept coins of different amounts and the system must compare the item cost with the entered coin.
- The system must check the validity of coins.
- The system shall be able to detect the low amount of ingredients and low number of cups and indicate with an indicator(small LED).

4.2 Advanced Class Diagram



- A vending machine is a composition of a coin slot, money box, display and selector.
- The user selects the coffee through the selector. Then the coffee machine prepares the coffee.
- The coffee machine dispenses the product. Here dispense is an association class.
- Item is an enumeration of the type of coffee.

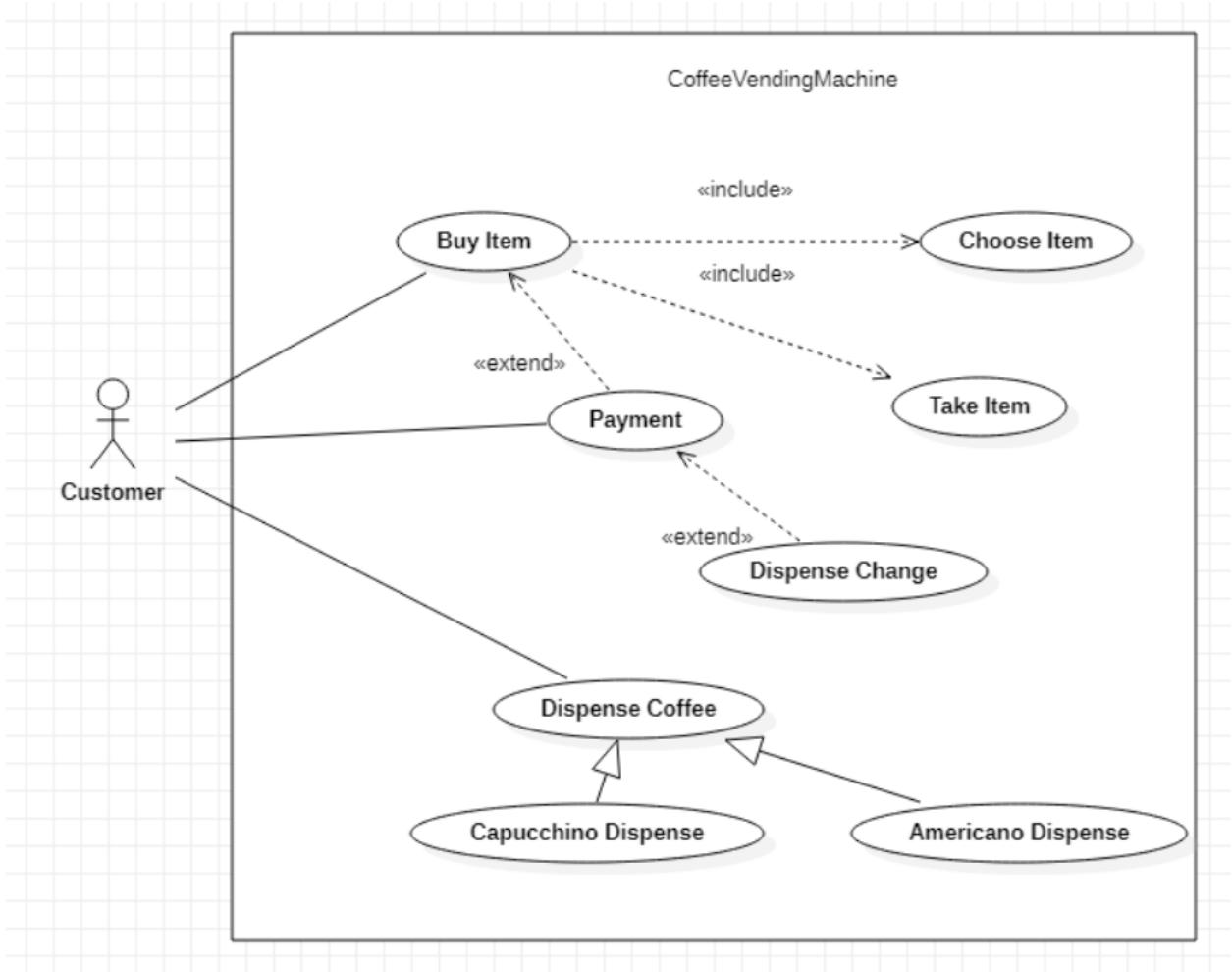
4.3 Advanced State Diagram



The state diagram contains one nested state and one submachine, which on successful login shows the CollectingMoney procedure and DispenseItem procedure. It contains initial state and termination state with CollectingMoney as a nested state including the required simple states. It also has a submachine state named DispenseItem with initial, termination state along with simple states; SettingRow, SettingColumn, Pouring.

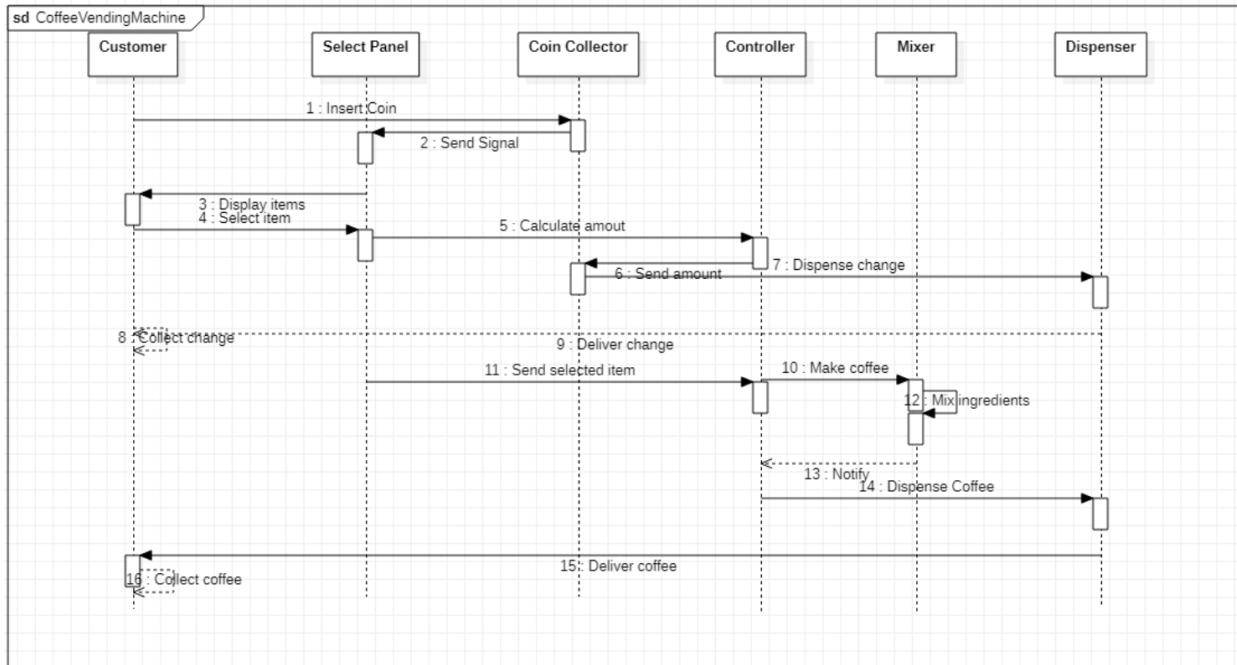
4.4 Advanced Use case Diagram

The advanced use case diagram has extra functionalities which includes extends, includes and generalization. The dispense change use case extends payment use case, payment use case extends buy item use case, buy item use case includes choose item and take item use case. Cappuccino dispenser and American dispense is generalized to super class dispense coffee.



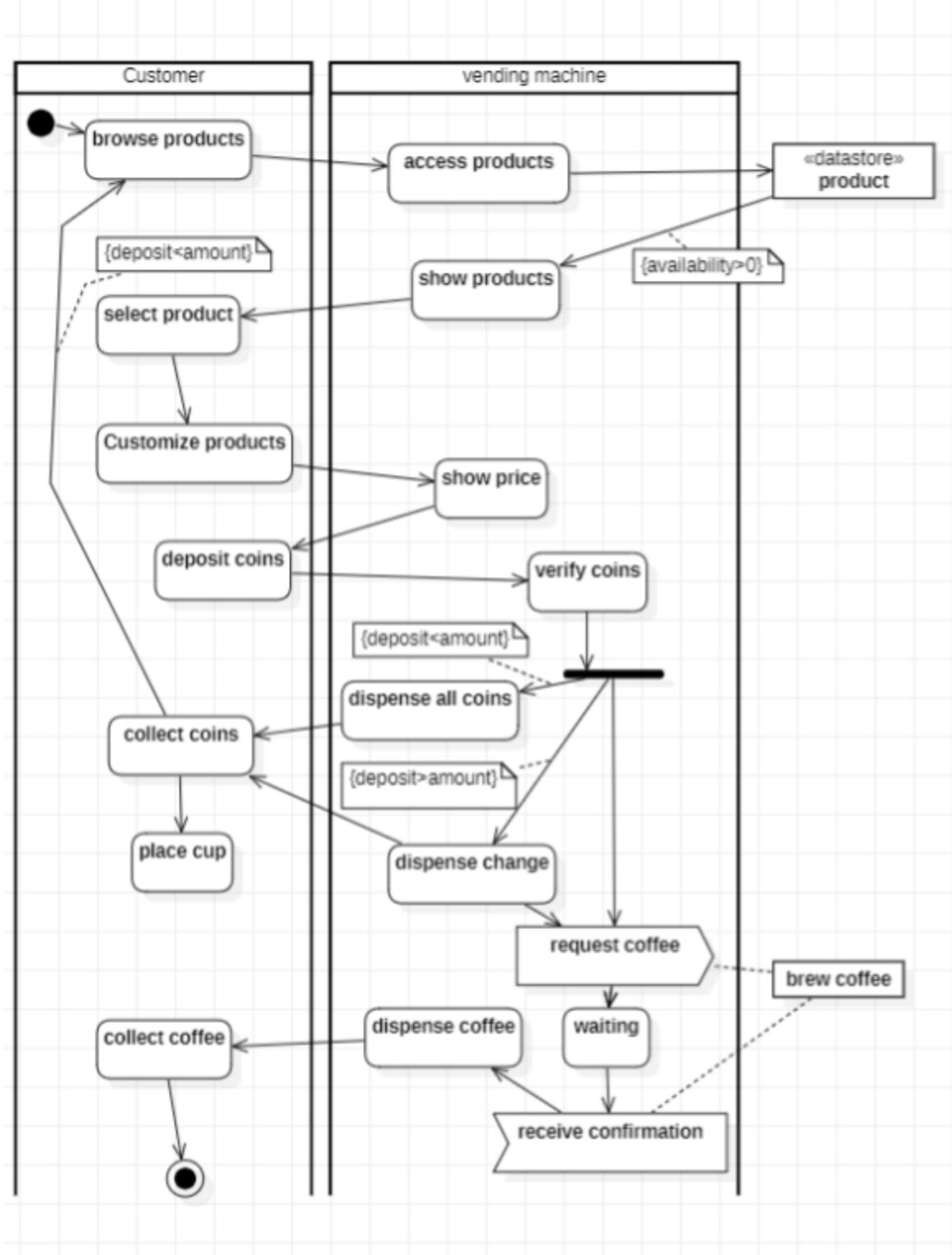
4.5 Advanced Sequence Diagram

The lifeline is the dotted line and the rectangles represent the period of time the object is executing and is hence called activation. The scenario shown here is customer selecting the coffee he needs from the select panel. Then insert coins, if insufficient coins all coins are returned. Else change is dispensed and coffee is prepared and dispensed.



4.6 Advanced Activity Diagram

The advanced activity diagram starts from initiation and in the customer swimlane, customer browses through the products , selects the product and deposits coins. The coins deposited is verified in the vending machine swimlane and a signal is sent to brew coffee , after receiving the signal the coffee is dispensed.

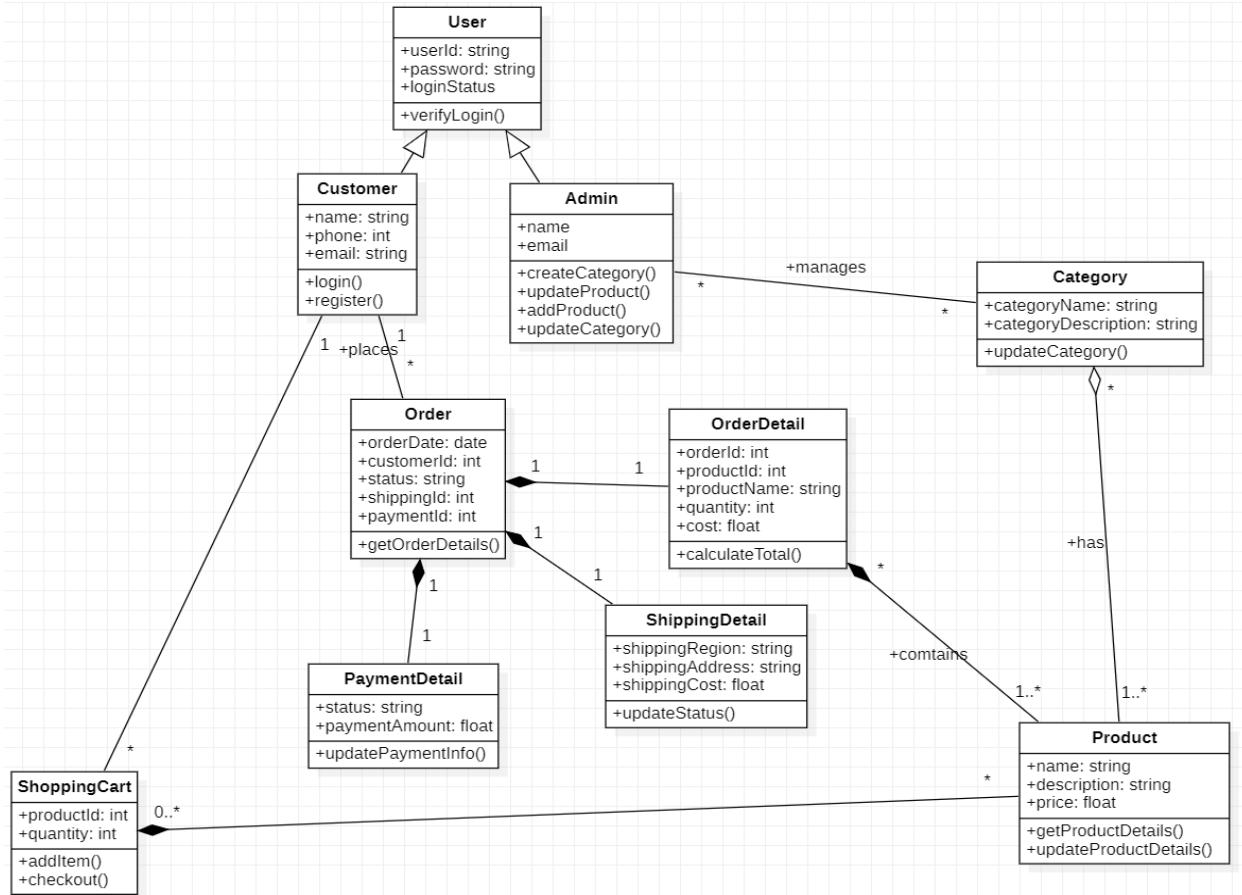


LAB 5:Online Shopping System

5.1 Software Requirement Specification

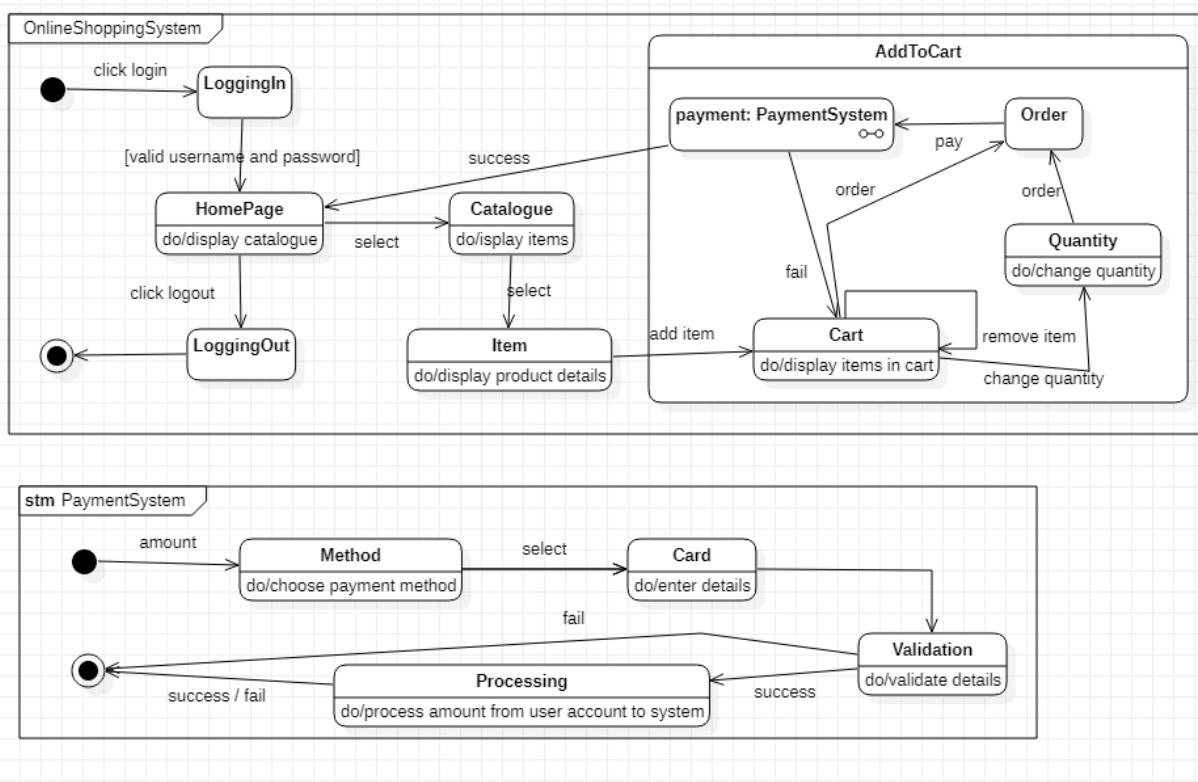
- The customer must have an account in the online website where he/she can purchase products.
- If a customer wants to buy the product then he/she must be registered, unregistered users can't go to the shopping cart.
- Customers login to the system by entering valid user id and password for the shopping.
- Changes to cart means the customer after login or registration can make an order or cancel order of the product from the shopping cart.
- The products sold for customers are sold for various categories like men,women,kids and home products.
- Customers can view all available products ,compare them and make a choice for purchasing the products.
- For customer there are many type of secure billing will be prepaid as debit or credit card, post paid as after shipping, check or bank draft. The security will provide by the third party like Pay-Pal etc.
- After the payment or surf the product the customer will logged out.

5.2 Advanced Class diagram



- An online shopping system user can either be a customer or an admin. So the user is generalized into customer and admin.
- Admin manages the category of different products. He is responsible for adding products.
- The user places orders. An order is a composition of Order detail, shipping detail and payment detail.
- Customer can also add 0 or more product to the shopping cart

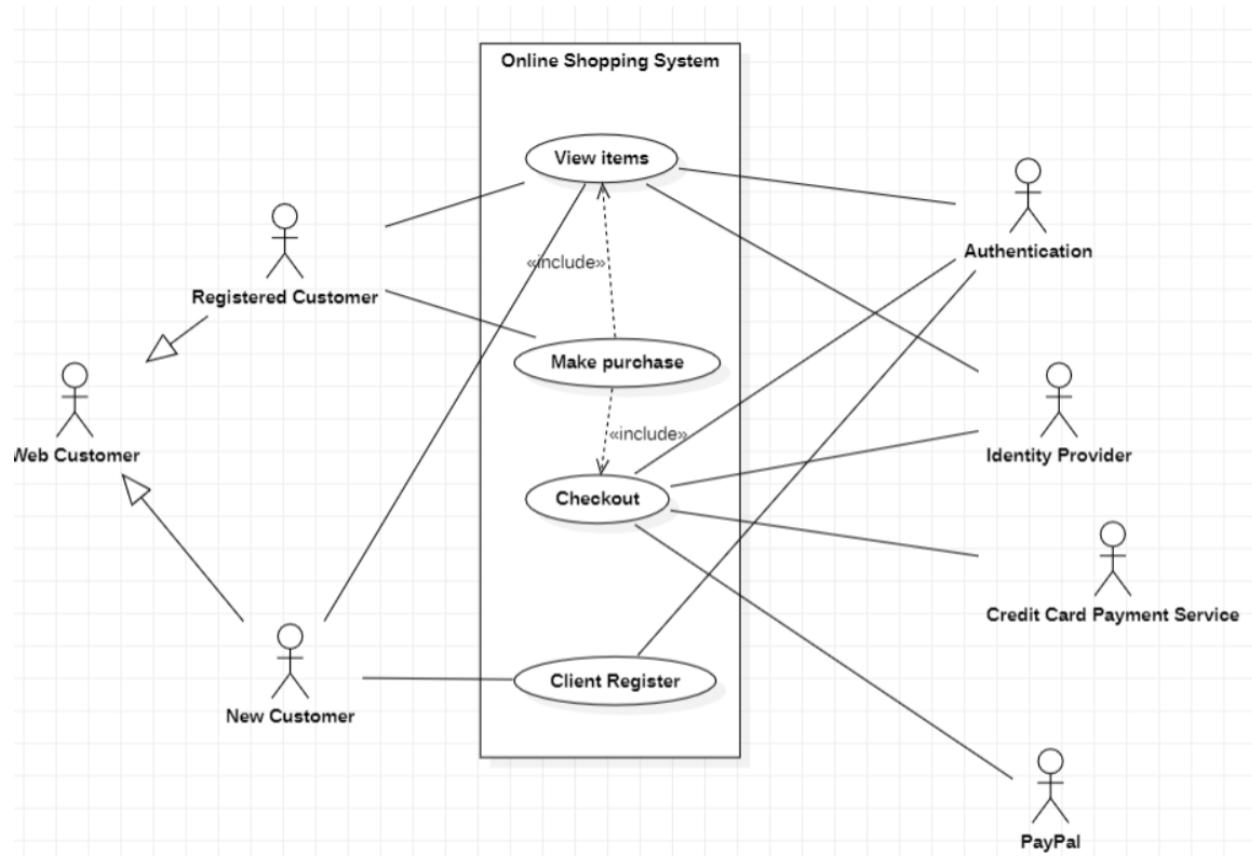
5.3 Advanced State Diagram



The state diagram contains one nested state and one submachine, which on successful login shows the AddToCart procedure and PaymentSystem procedure. It contains initial state and termination state with AddToCart as a nested state including the required simple states. It also has a submachine state named PaymentSystem with initial, termination state along with simple states; Method, Card, Validation, Processing.

5.4 Advanced use case diagram

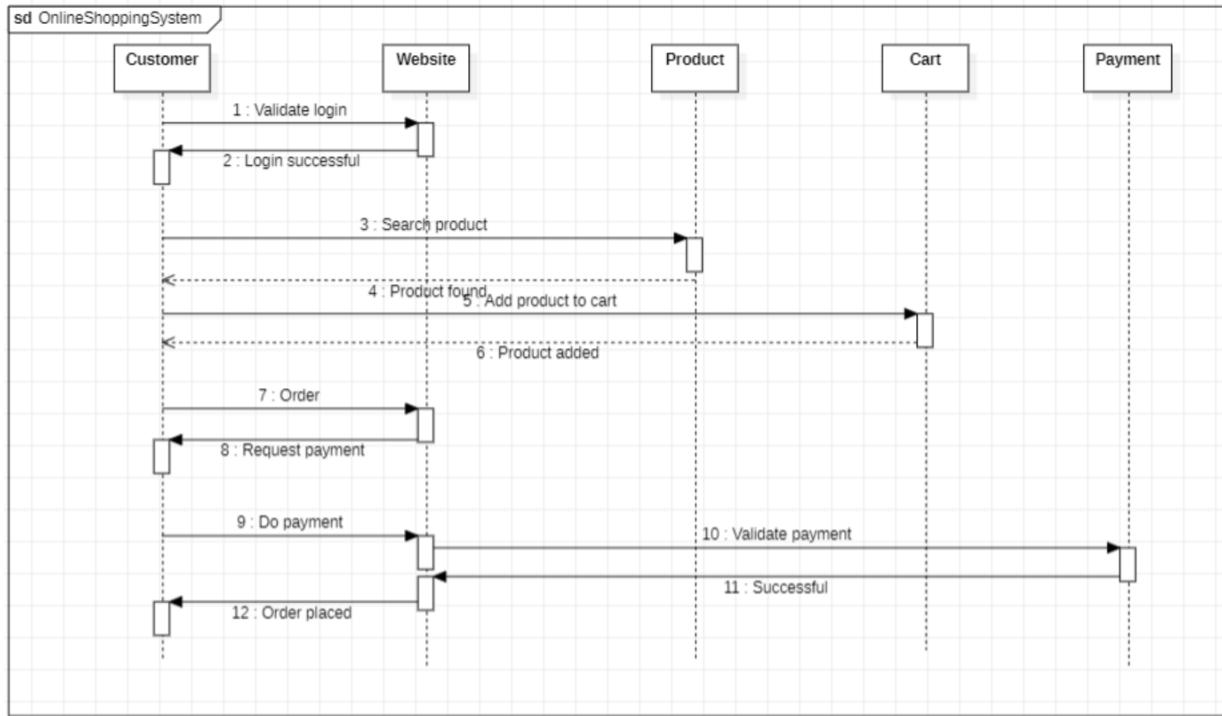
The advanced use case diagram has extra functionalities which includes extends, includes and generalization. Web customer is generalized into registered customer and new customer. Only registered customers can make purchases, while new customers can still view items.



5.5 Advanced Sequence diagram

The actors are: Customer website product cart payment

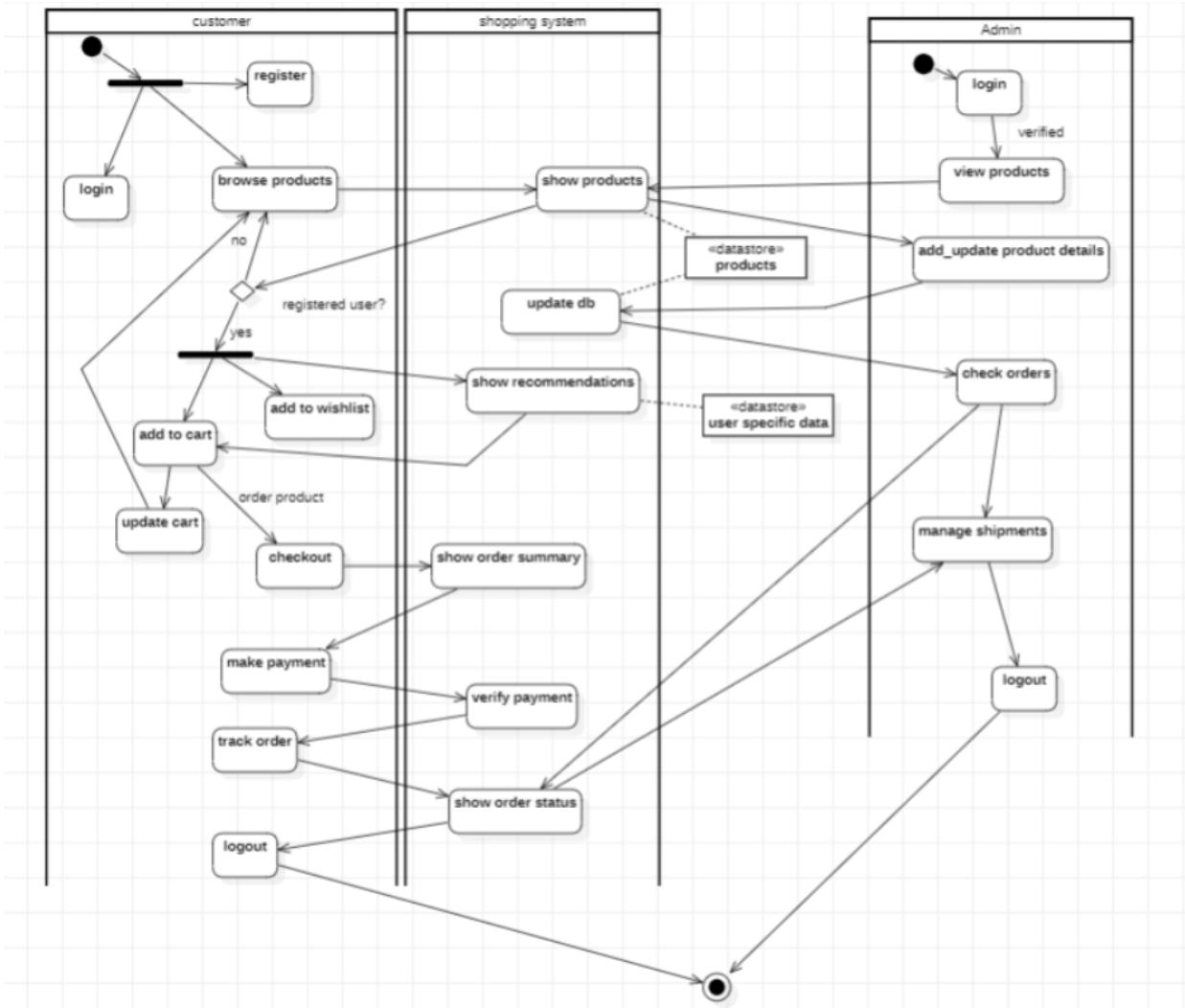
The scenario shown here is customer logging in into the website, searching for the product then adding product to cart checkout. On successful payment, the order is placed successfully.



5.6 Advanced Activity diagram

There are 3 swimlanes: Customer, shopping system and admin.

The scenario shown here is customer logging in into the website, searching for the product then adding product to cart checkout. On successful payment, the order is placed successfully.

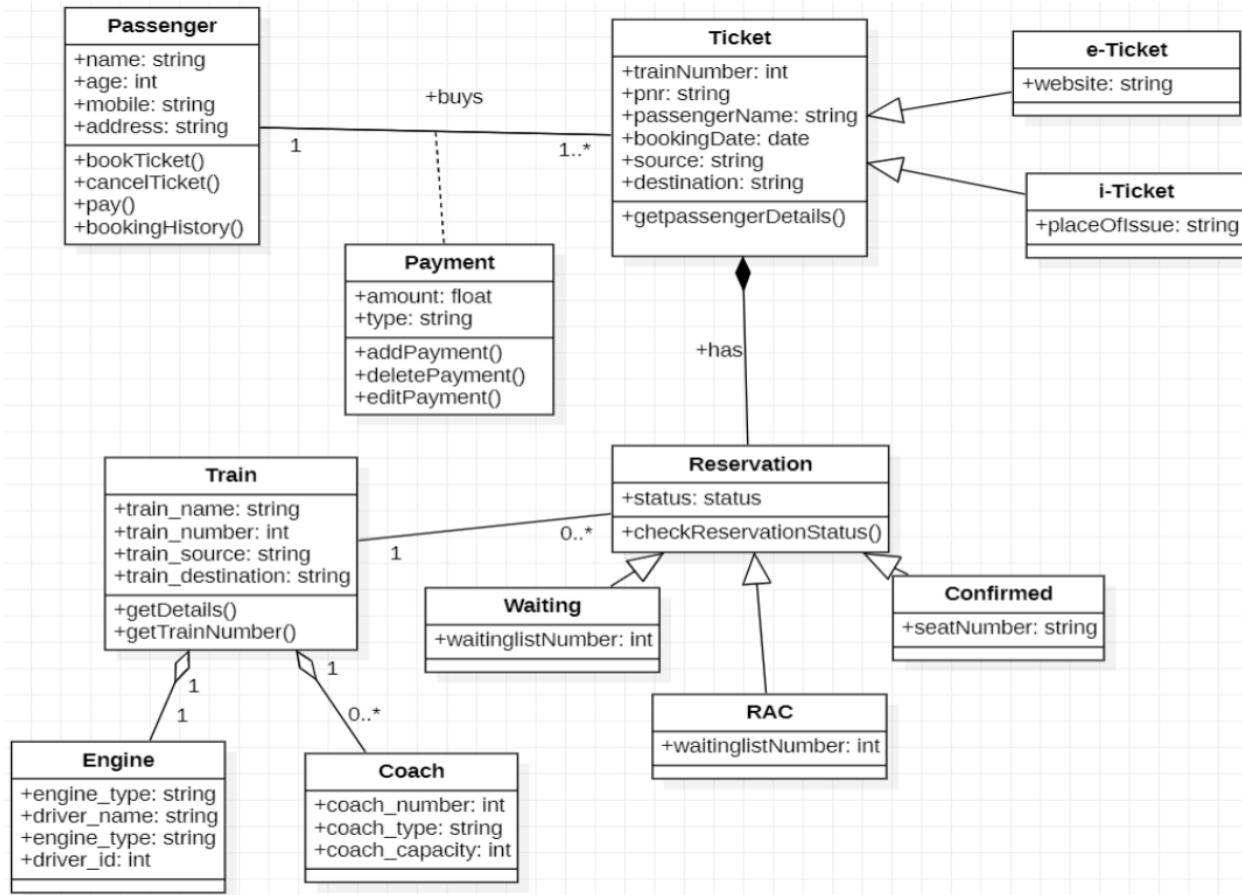


LAB 6:Railway Reservation Machine

6.1 Software Requirement Specification

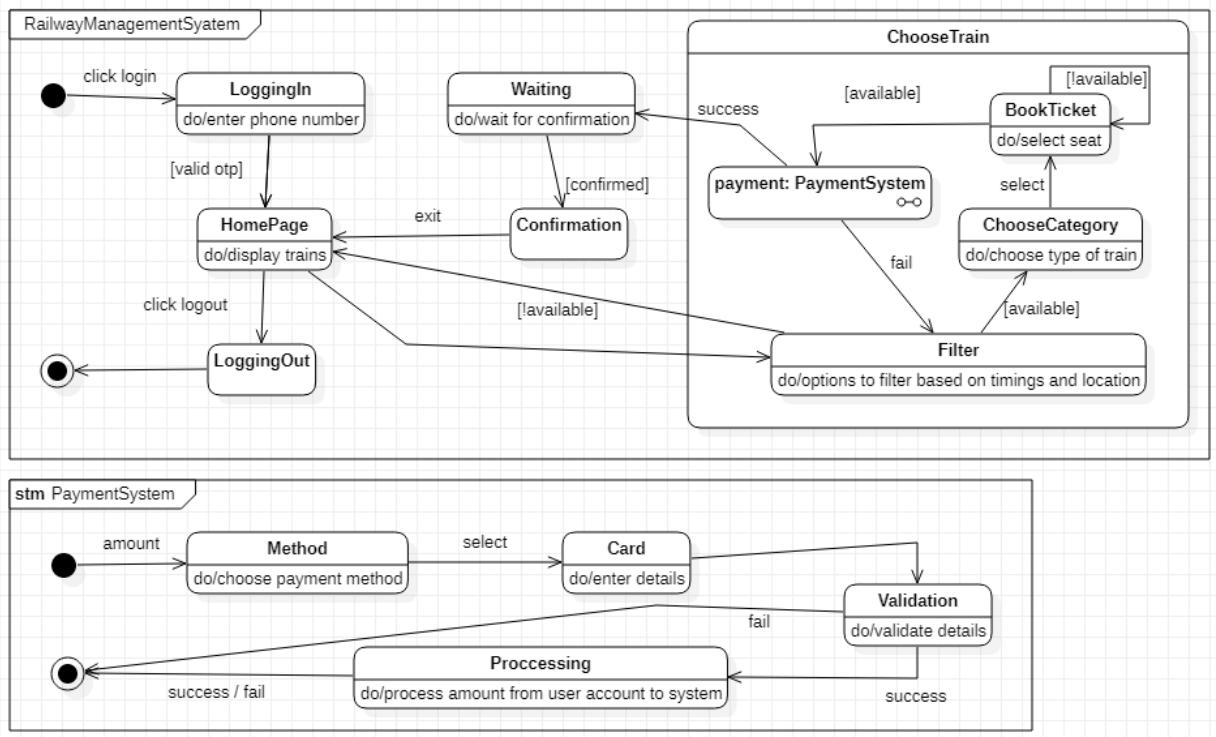
- Each user should have a user id and a password. Record of the users of the system should be kept in the log file. Provision should be made for full backup of the system.
- The customers can view the trains available at any day, the cost and number of tickets available for any train.
- Customer can book a ticket only if the tickets are available. Customer searches for the availability of tickets then if the tickets are available he books the tickets by initially filling details in a form.
- Tickets can be booked in two ways by i-ticket or by e-ticket booking.
- In case of i-ticket booking customer can book the tickets online and the tickets are couriered to Particular customer at their address. But in case of e-ticket booking and canceling tickets are booked and canceled online sitting at the home and customer himself has to take print of the ticket but in both the cases amount for tickets are deducted from customers account.
- For cancellation of a ticket the customer has to go to the reservation office then fill in the cancellation form and ask the clerk to cancel the ticket then the refund is transferred to the customer account.
- After booking a ticket the customer has to checkout by paying fare amount to the clerk.
- The system displays the details of the train of which the user enters the name. The Information is saved and the corresponding updating takes place in the database.

6.2 Class Diagram



- Passenger books ticket. The ticket is generalized as i-ticket and e-ticket.
- Reservation cannot exist without a ticket. So the ticket has a composition relation with the ticket.
- Reservation can be generalized as waiting, RAC and confirmed based on the status of the ticket.
- Train is made of 1 engine and 0 or more coaches. The coaches and engine can exist without the train, so aggregation is used here

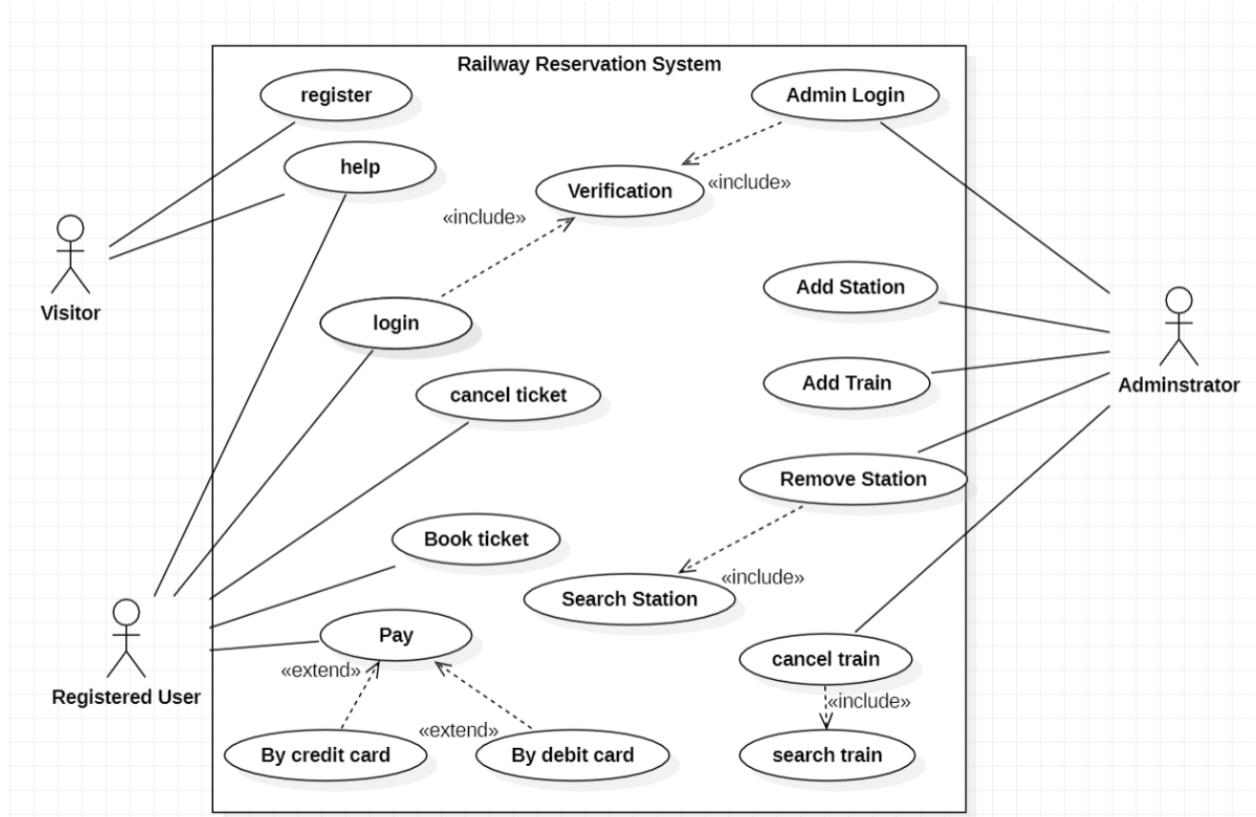
6.3 State Diagram



The state diagram contains one nested state and one submachine, which on successful login shows the ChooseTrain details and PaymentSystem procedure. It contains initial state and termination state with ChooseTrain as a nested state including the required simple states. It also has a submachine state named PaymentSystem with initial, termination state along with simple states; Method, Card, Validation, Processing.

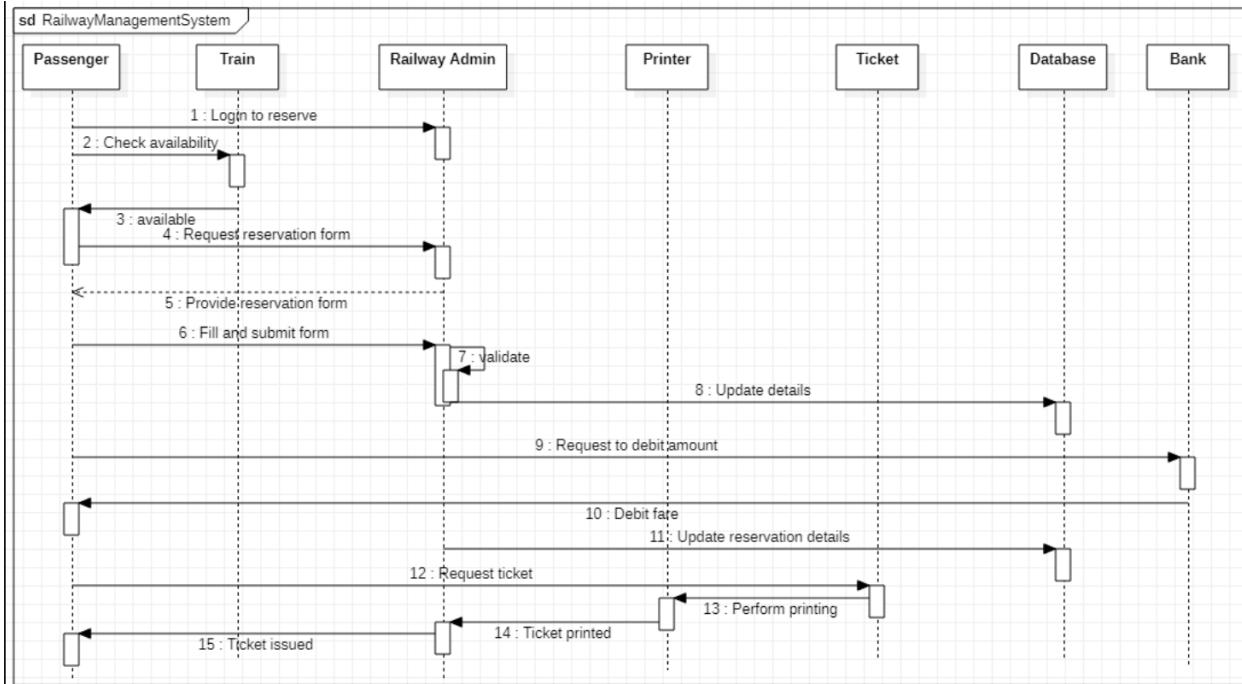
6.4 Advanced use case diagram

The advanced use case diagram has extra functionalities which includes extends, includes and generalization. Here both admin login and user login includes verification. Pay by debit card and pay by credit card extends pay usecase.



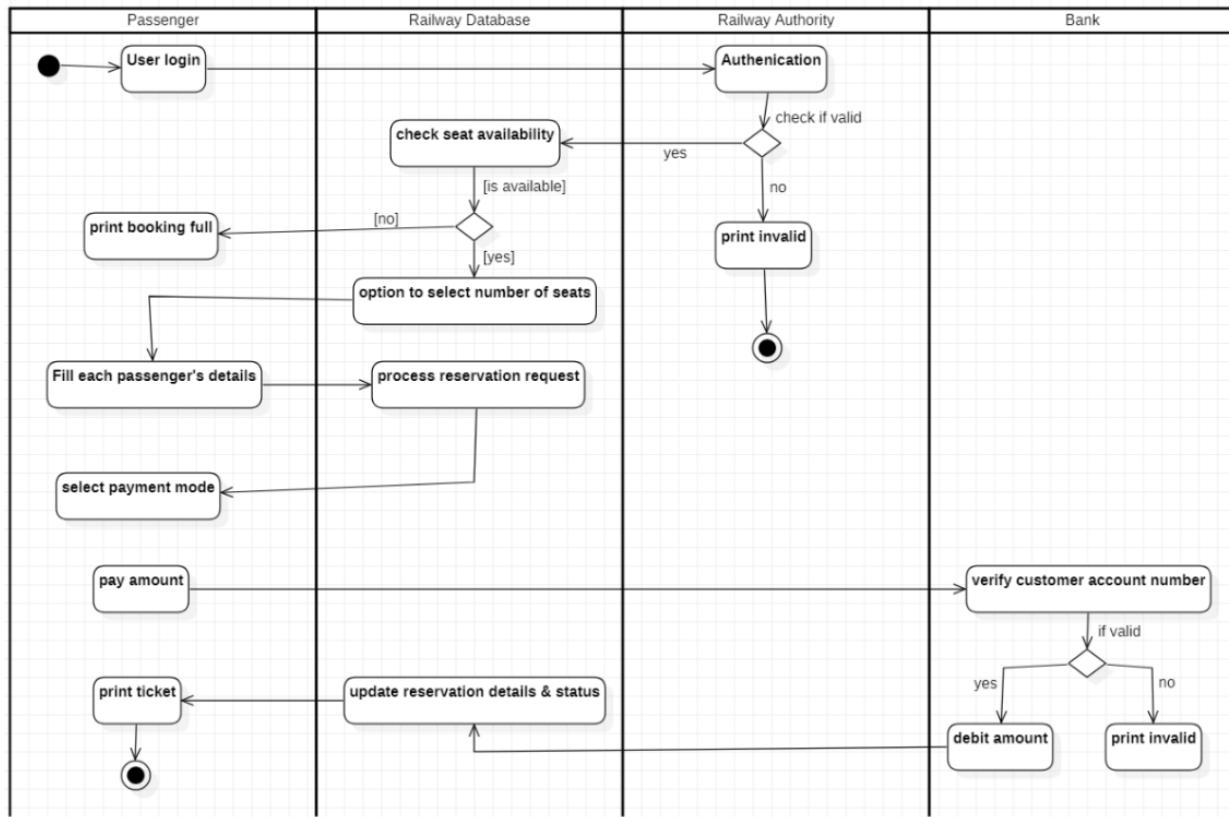
6.5 Advanced Sequence diagram

The lifeline is the dotted line and the rectangles represent the period of time the object is executing and is hence called activation. The recursive function of validation is shown by double activation rectangle of validation with self-transition. Reply message is used to return back to lifelines with the required message. The scenario shown here is customer trying to reserve seats on a train.



6.6 Advanced Activity diagram

The advanced activity diagram starts from initiation and in the passenger swimlane, the passenger login activity where a signal is sent to the network for request validation and upon confirmation the control flows to check seat availability activity. There are four swimlanes namely passenger, railway database, railway authority and bank where each one indicates the passenger operations, check seat availability, check validation, confirm payment respectively. Then the control flows to the home page and then termination activities.

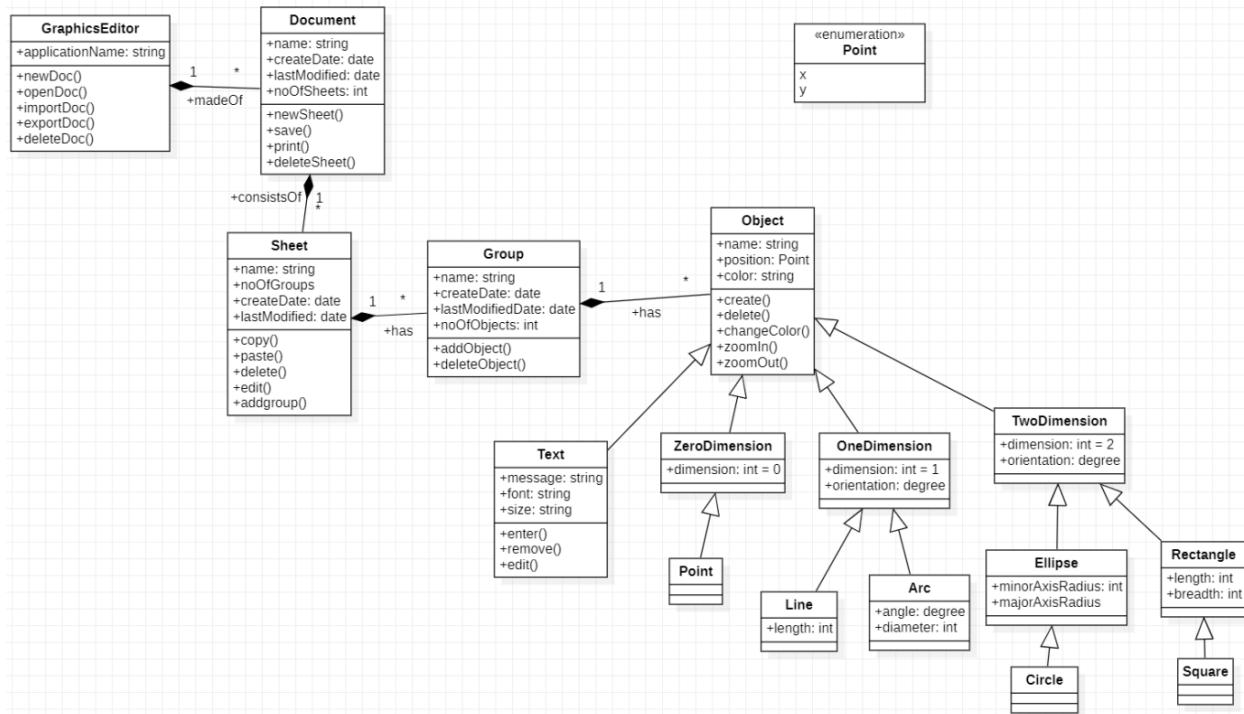


LAB 7:Graphics Editor

7.1 Software Requirement Specification

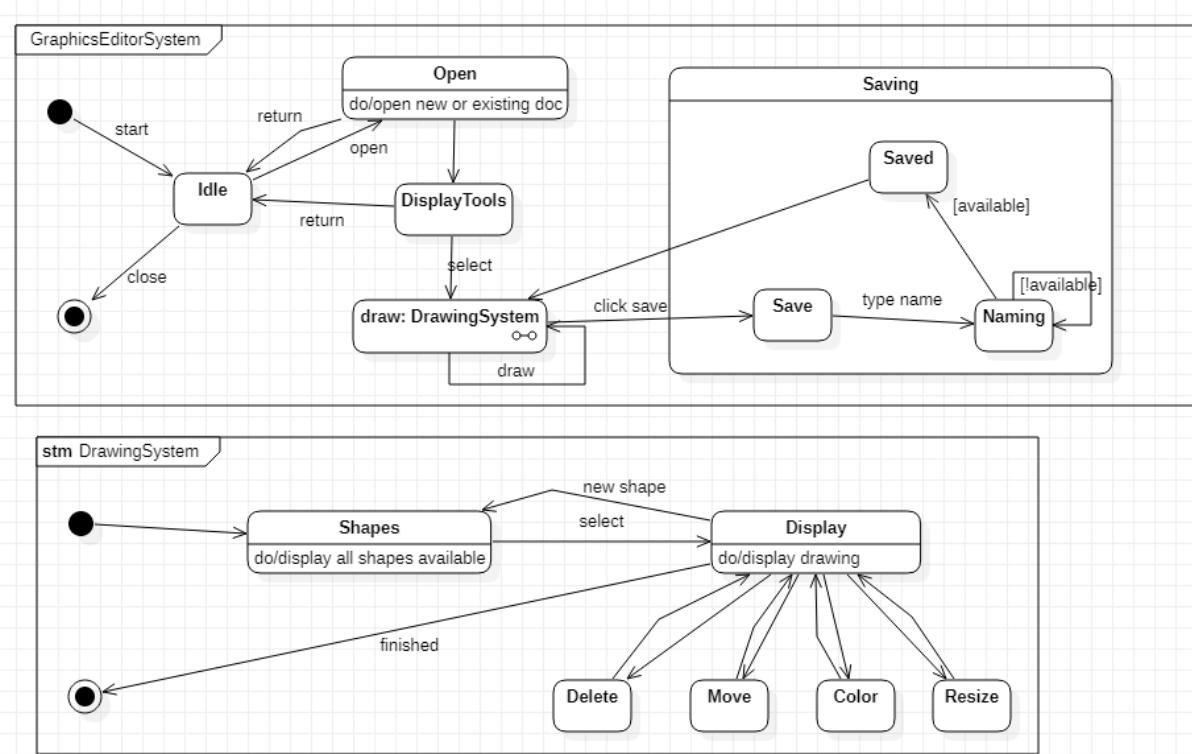
- The graphical editor consists of a graphical document editor which can be used to create new documents, delete documents, update or view the document.
- The graphical document editor consists of many documents, where each document can be saved, opened, printed or create a new one.
- A document is made up of many sheets which can have graphics included in them.
- Sheets have multiple numbers of drawing objects, which can be created, grouped or formatted.
- The programmer must provide implementations of functions that draw objects and their connections, as well as functions that add and remove connections. The latter function will be handled by a specific event listener. Any changes made in real-time to the underlying model will also be updated in the diagram through a separate event listener.
- The user can also add and remove connections between these objects as needed using the palette supplied, thus modifying the underlying model.
- Each sheet contains drawing objects, including text, geometrical objects and groups. A group is simply a set of drawing objects.
- A geometrical object includes circle, ellipse, rectangles, lines and squares, trapeziums which are identified by their respective constraints.

7.2 Advanced Class Diagram



- Graphics Editor class is made up of the composition of Document class as document class is there particularly for this class.
- And the Document class is made up of the composition of the Sheet class and sheet of group class and group of object class as one class cannot exist without another class.
- Object class is generalized into text, zeroDimension, One dimension and two dimension class.
- Zero dimension is generalized into point.one dimension class into line and arc.two dimension class into ellipse and rectangle class and further ellipse and rectangle into circle and square respectively.

7.3 Advanced State Diagram

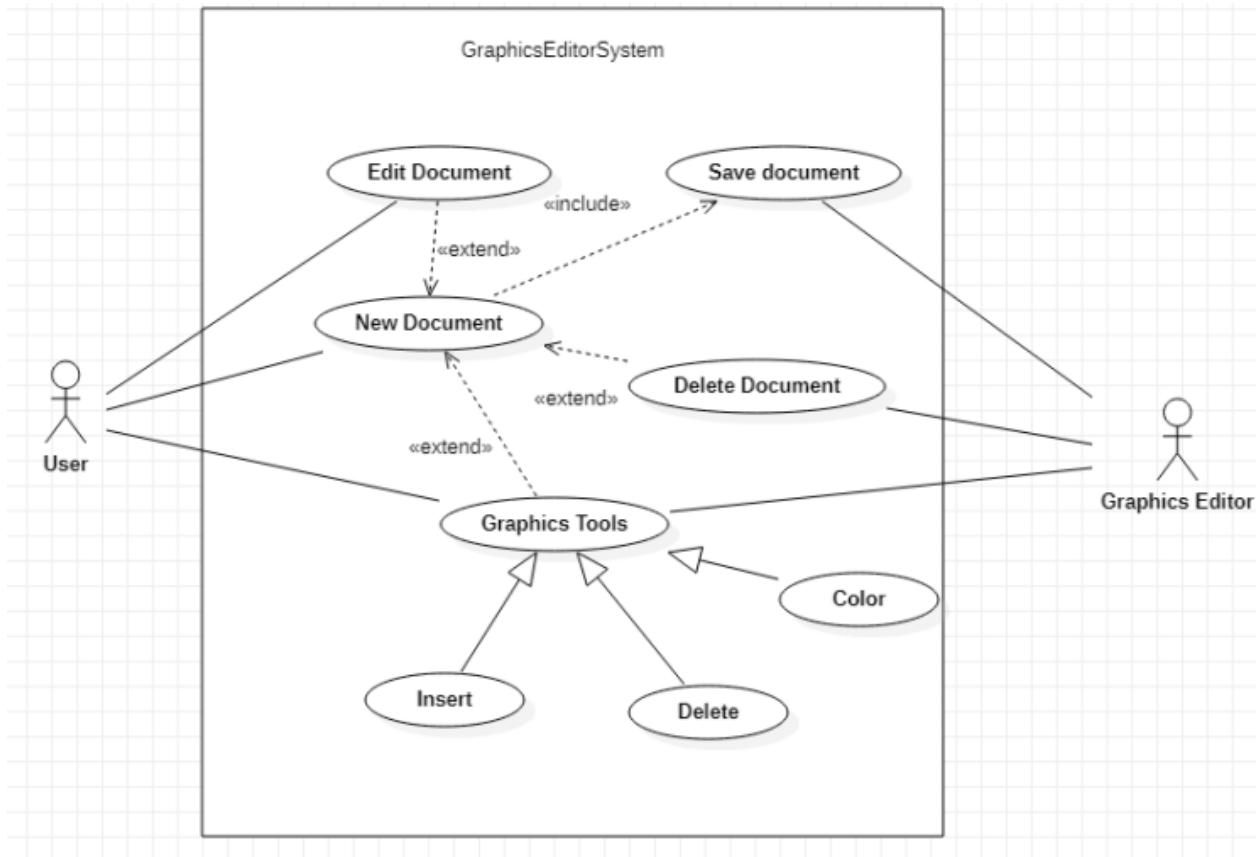


The state diagram contains one nested state and one submachine, which on successful login shows the Saving procedure and DrawingSystem procedure. It contains initial state and termination state with Saving as a nested state including the required simple states. It also has a submachine state named DrawingSystem with initial, termination state along with simple states; Shapes, Display and format each shape.

7.4 Advanced use case diagram

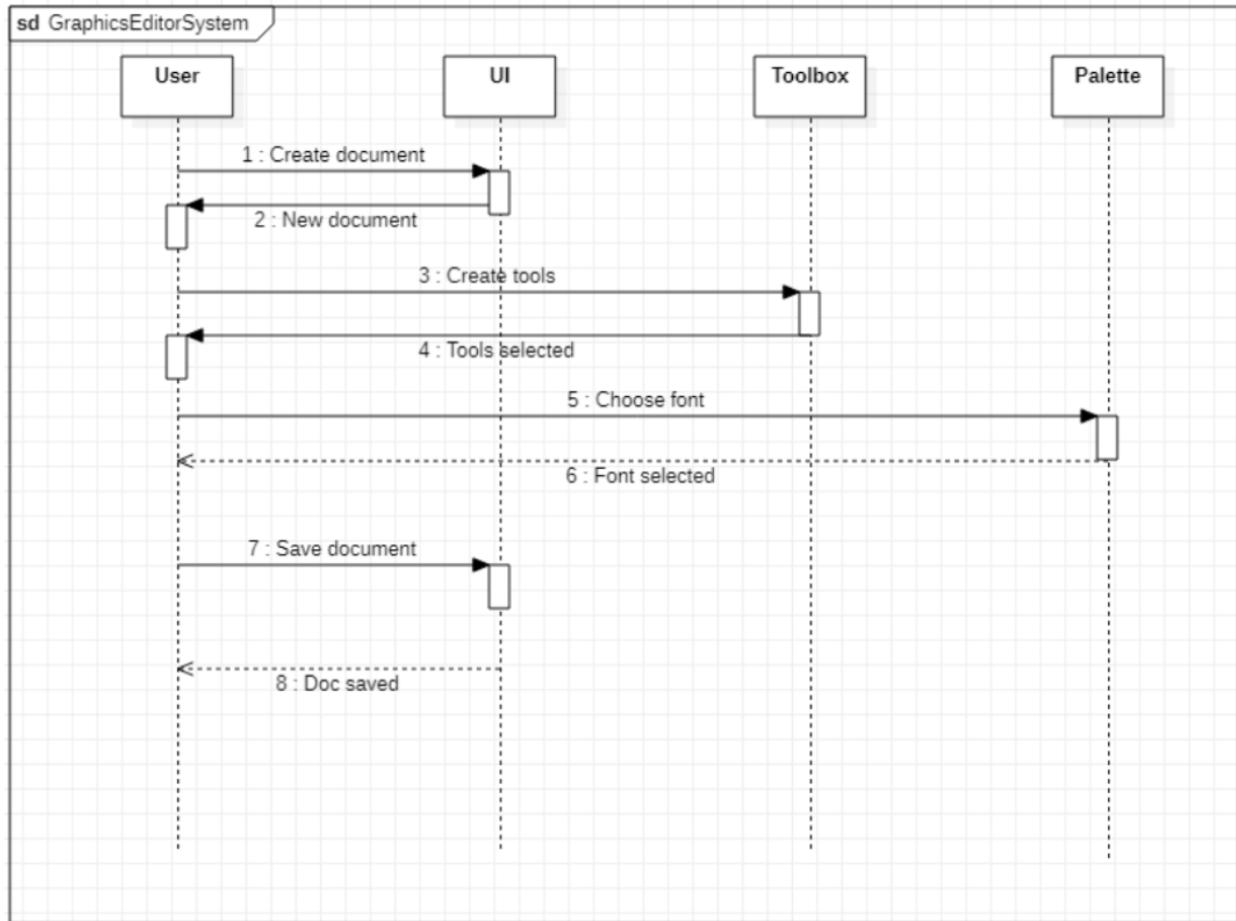
The advanced use case diagram has extra functionalities which includes extends, includes and generalization. The edit document use case extends new document use case, delete document use case extends new document use case, graphic tools use case extends new document use case,

new document use case includes save document use case. Insert, delete and color is generalized to super class graphics tools.



7.5 Advanced Sequence diagram

The lifeline is the dotted line and the rectangles represent the period of time the object is executing and is hence called activation. Reply message is used to return back to lifelines with the required message. The scenario shown here is user creating a document and saving it.



7.6 Advanced Activity diagram

The advanced activity diagram starts from initiation and in the user swimlane, the user login activity where a signal is sent to the network for request validation and upon confirmation the control flows to open file activity. There are two horizontal swimlanes namely user and editor where each one indicates the user operations and drawing a diagram respectively. Then the control flows to the close file activity and then termination activities.

