VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Analysis and Design

Submitted by

CHETHANA D (1BM20CS405)

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
April-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 "LAB COURSE **Object Oriented Analysis and Design**" carried out by **CHETHANA D (1BM20CS405),** 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 academic year 2021-2022. The Lab report has been approved as it satisfies the academic requirements in respect of an **Object Oriented Analysis and Design - (20CS6PCOMD)** work prescribed for the said degree.

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

Dr. Jyothi S Nayak
Professor and Head
Department of CSE
BMSCE, Bengaluru

`

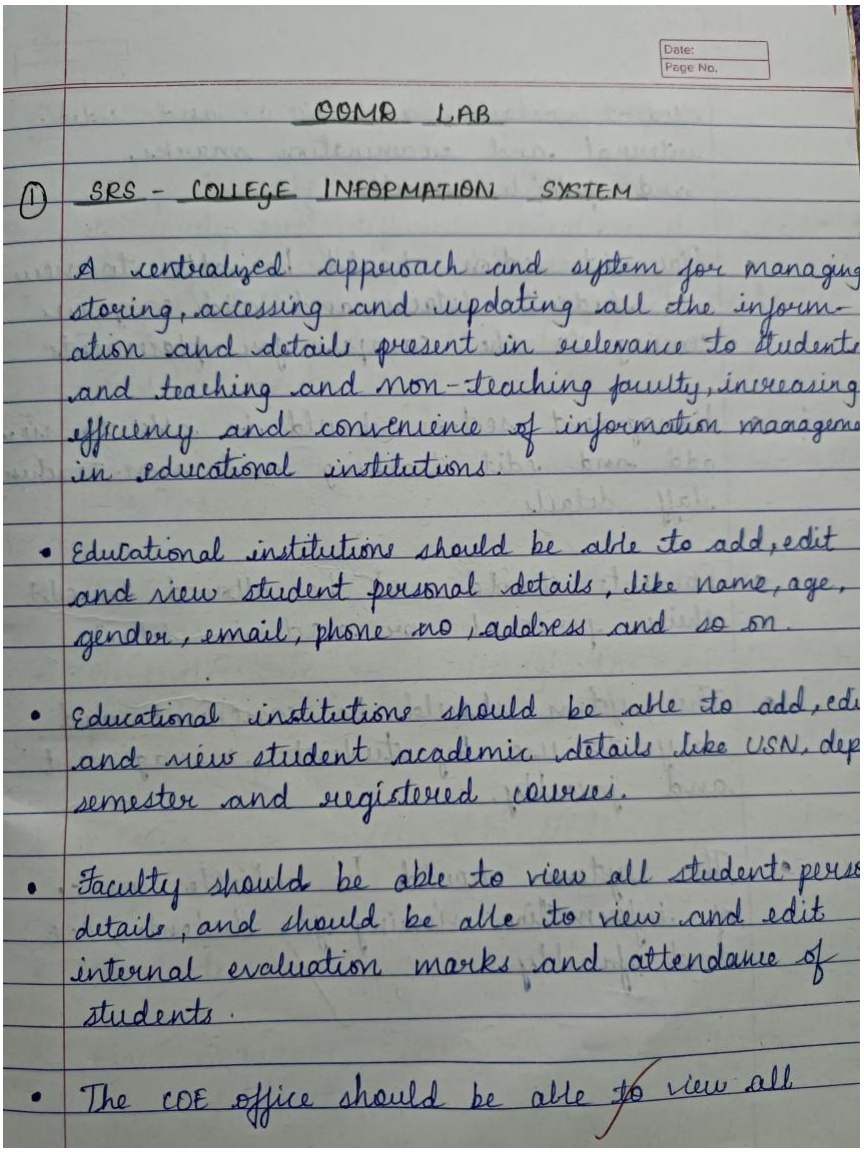
Index Sheet

Course Outcome

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

1. College Information System -

a) SRS:



Date:
Page No.

OOMD LAB

① SRS - COLLEGE INFORMATION SYSTEM

A centralized approach and system for managing storing, accessing and updating all the information and details present in relevance to students and teaching and non-teaching faculty, increasing efficiency and convenience of information management in educational institutions.

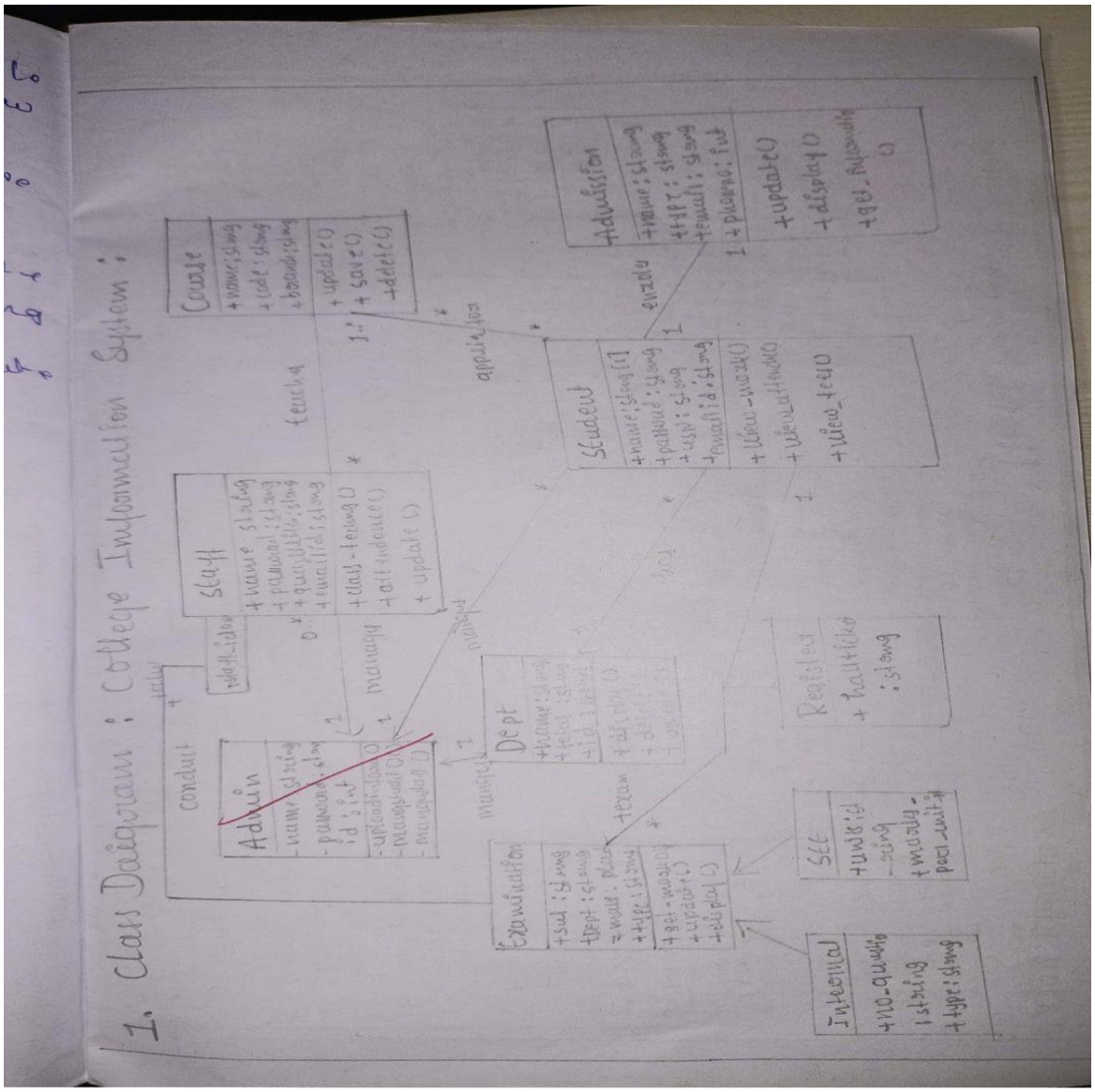
- Educational institutions should be able to add, edit and view student personal details, like name, age, gender, email, phone no, address and so on.
- Educational institutions should be able to add, edi and view student academic details like USN, dep semester and registered courses.
- Faculty should be able to view all student perso details, and should be able to view and edit internal evaluation marks and attendance of students.
- The COE office should be able to view all

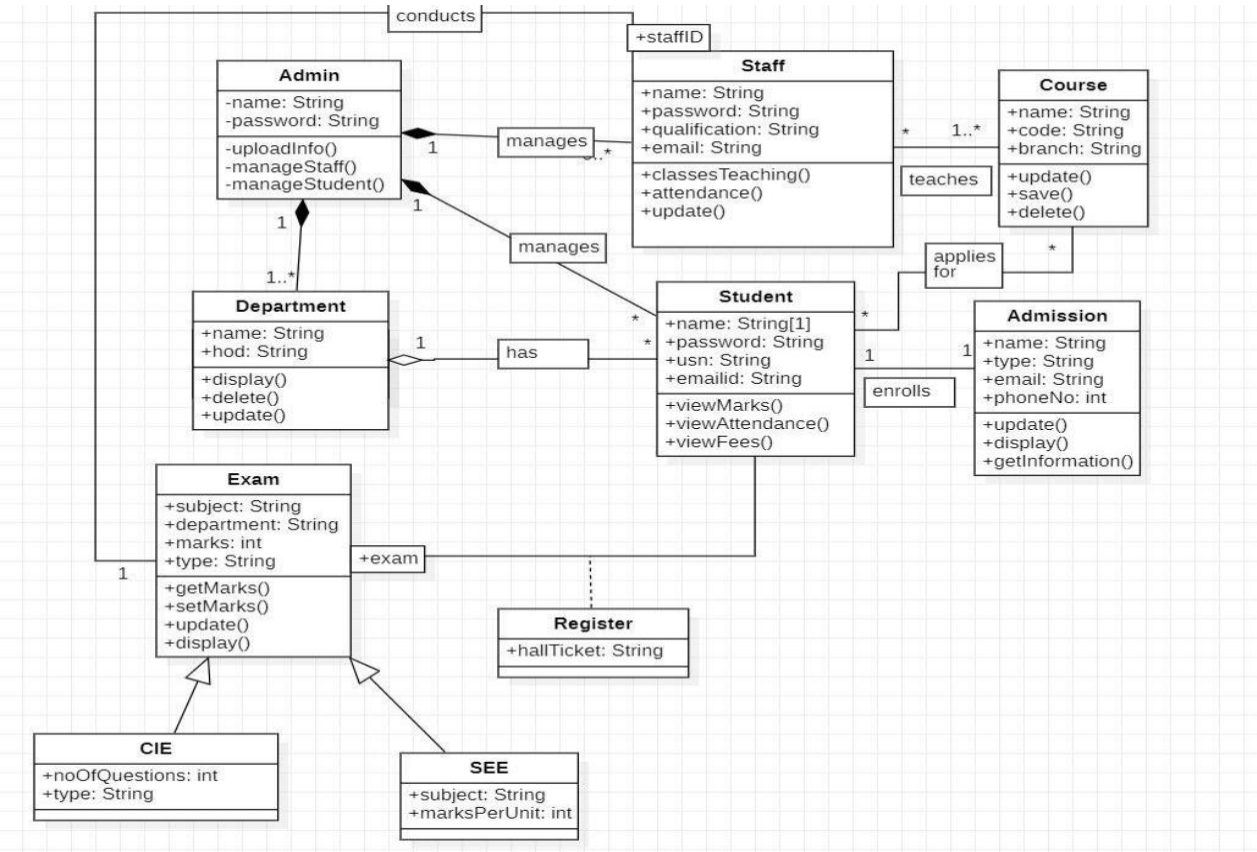
Date:
Page No.

student details, and view and edit. internal and examination marks, and publish results

- Placement section should be able to view all student details, and add companies coming to the campus for placements.
- Management section should be able to view, add and edit teaching and non-teaching staff details.
- Students should not be allowed to edit their personal or academic details.
- The system should be convenient and easy to use by students, management and faculty.
- The system should be reliable source of information viewing for students, COE and faculty.

b) Advanced Class Diagram

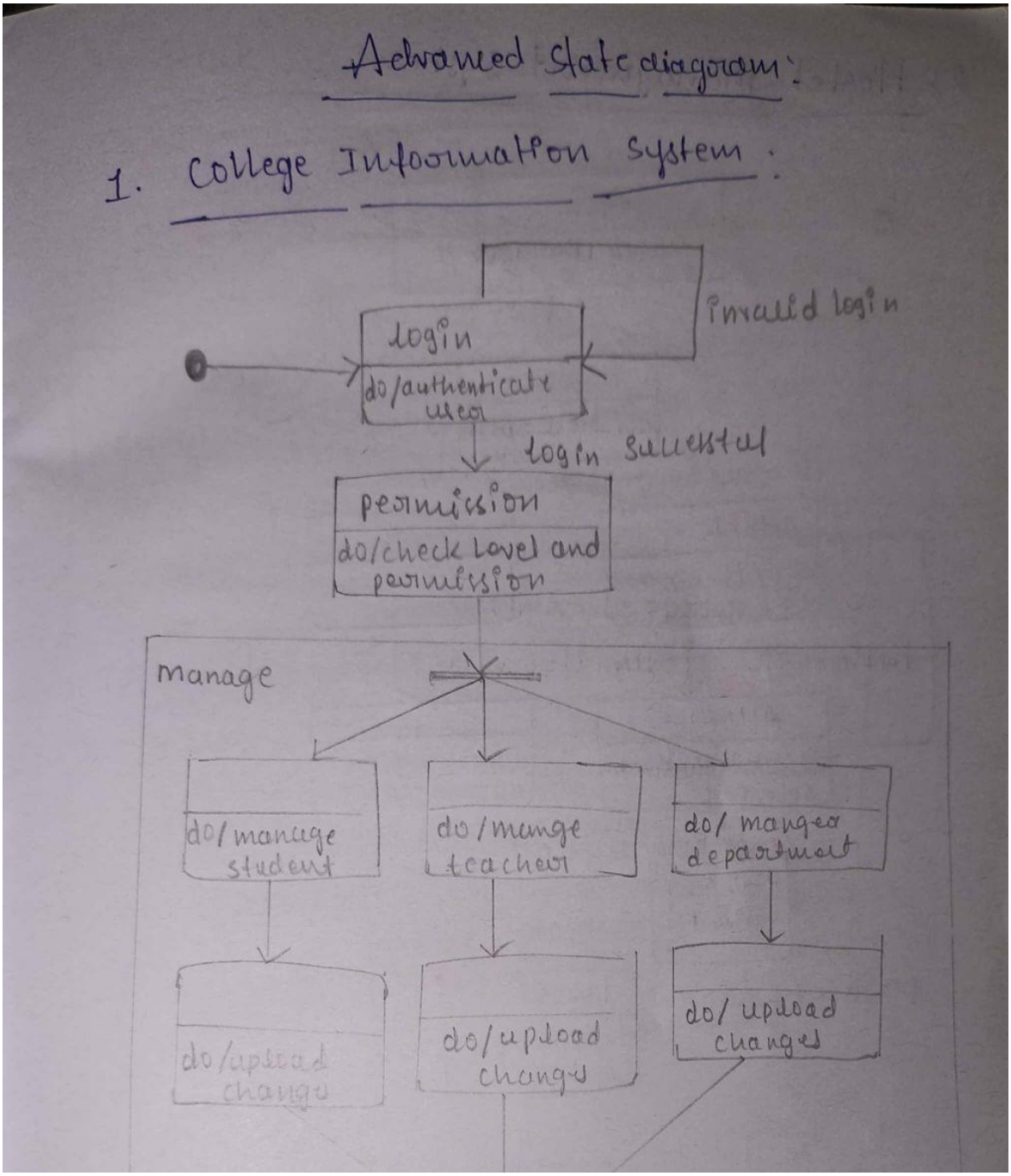


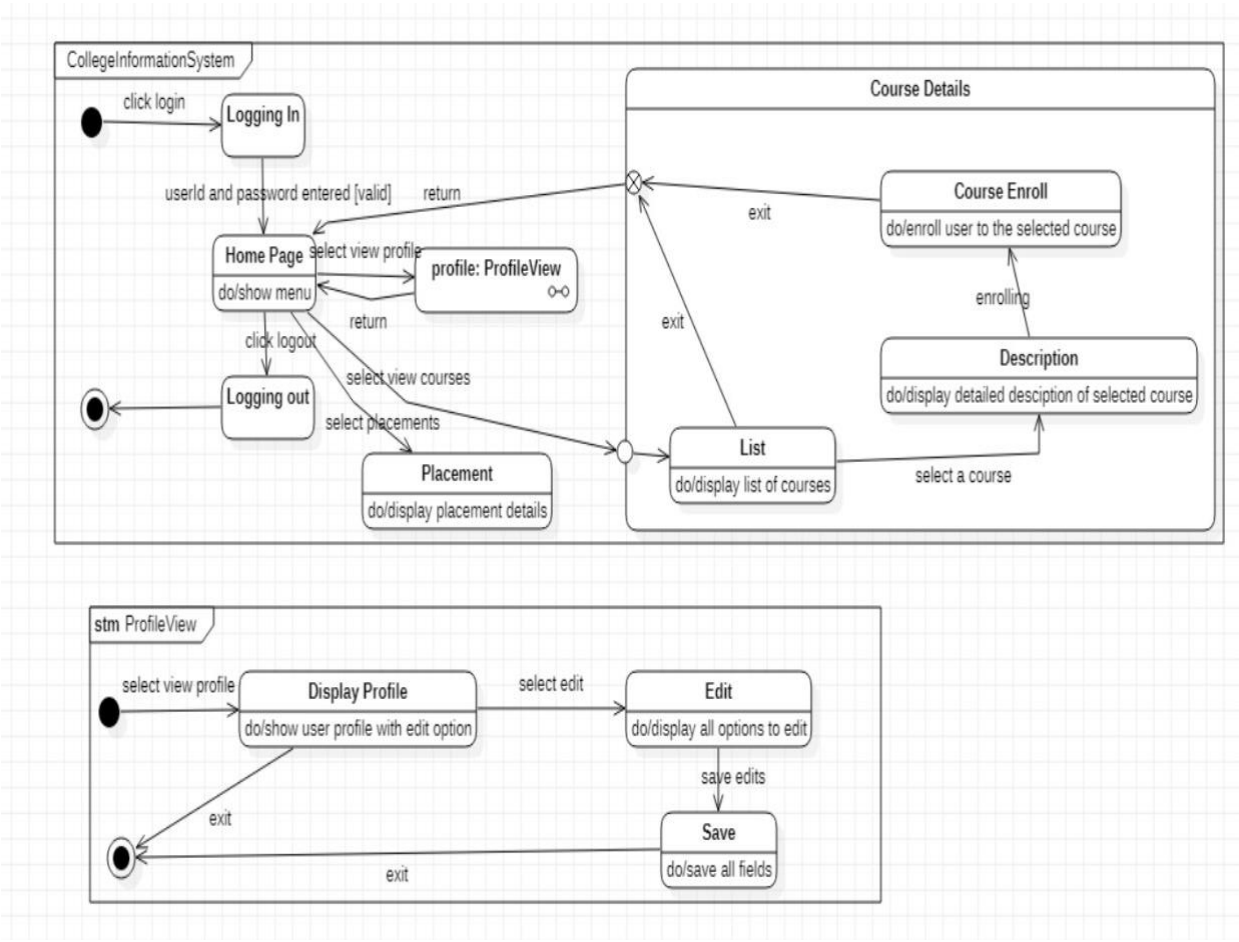


c) Advance State Diagram:

Advanced State diagram:

1. College Information System:

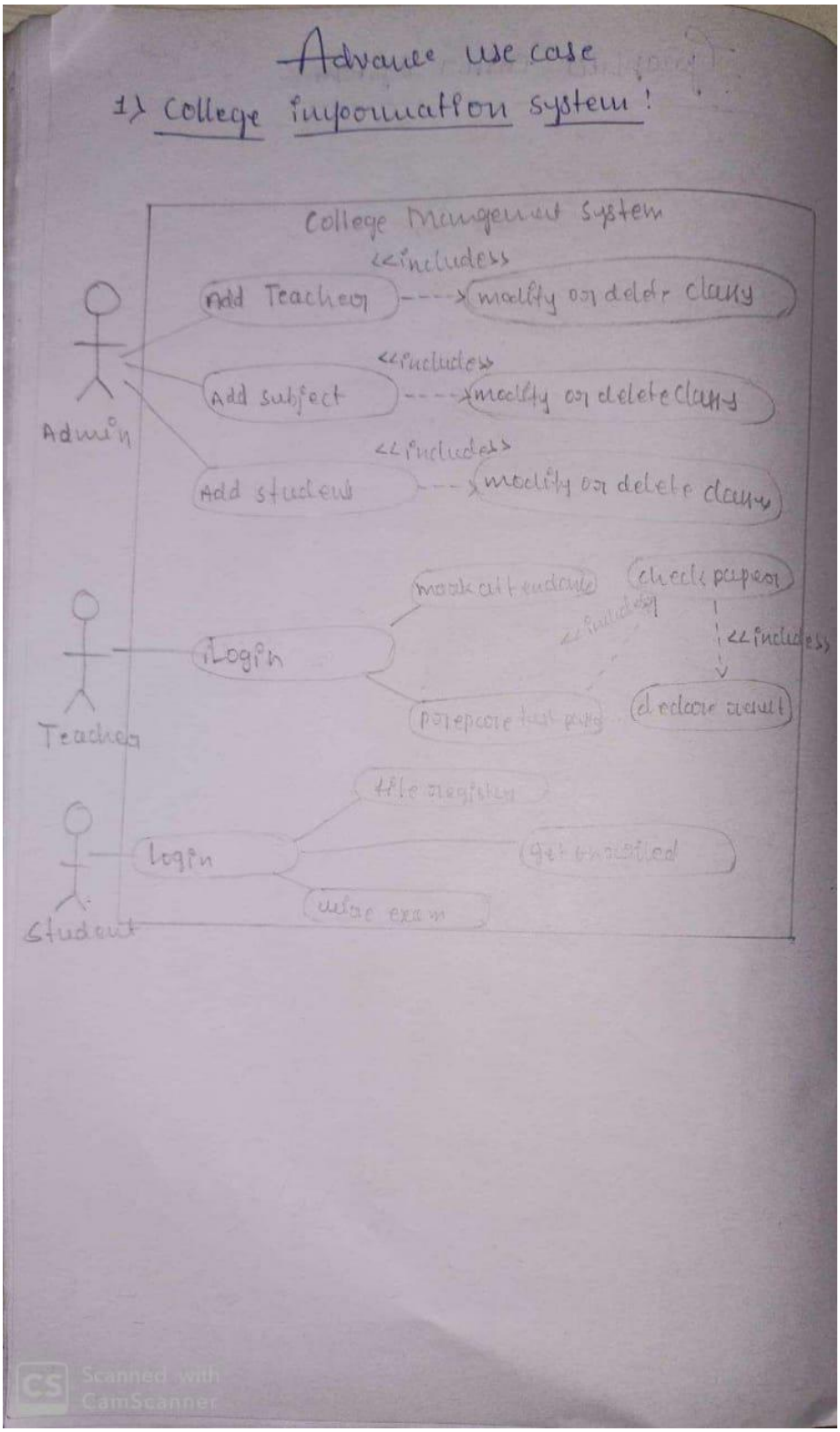


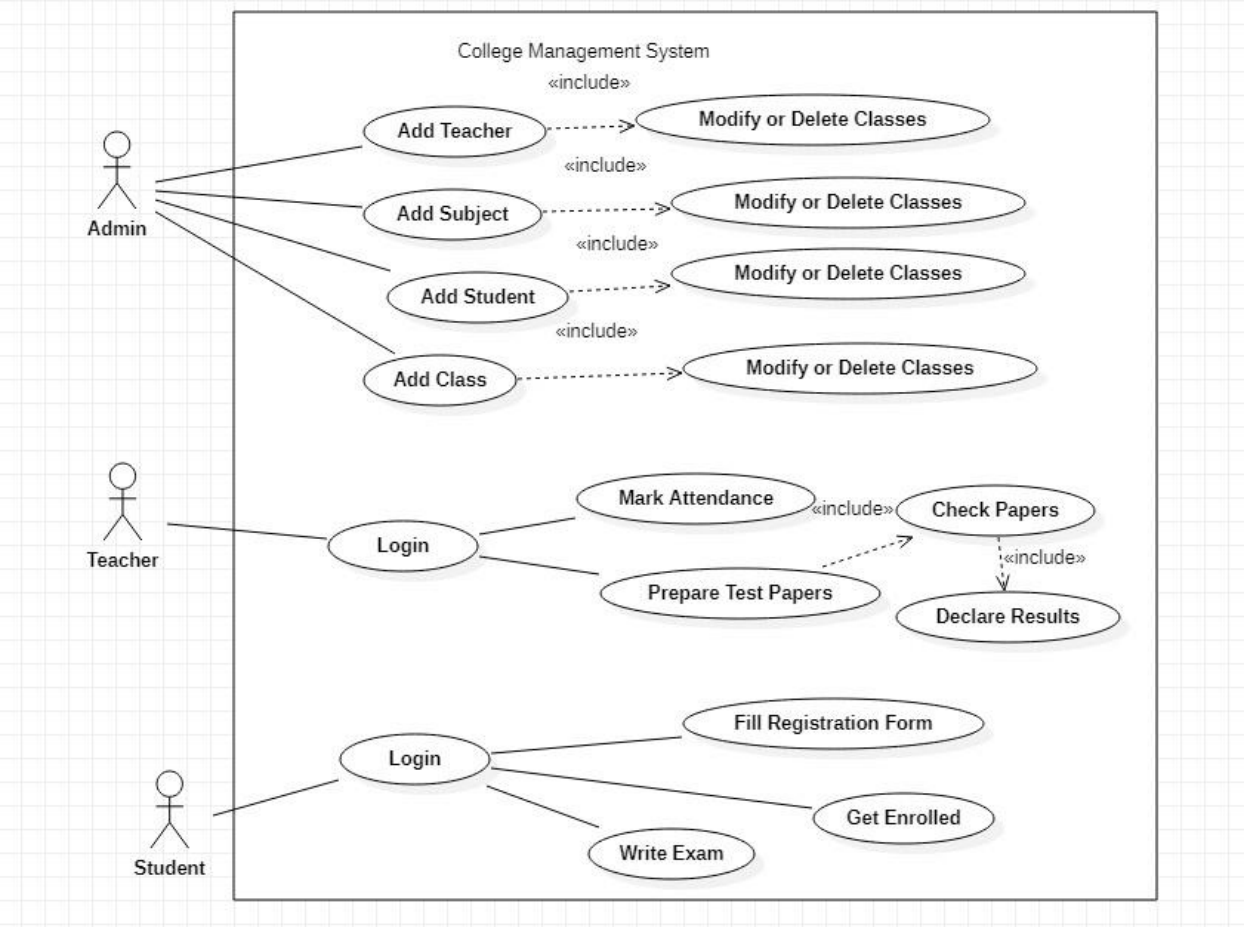


d) Advance Use Case Diagram:

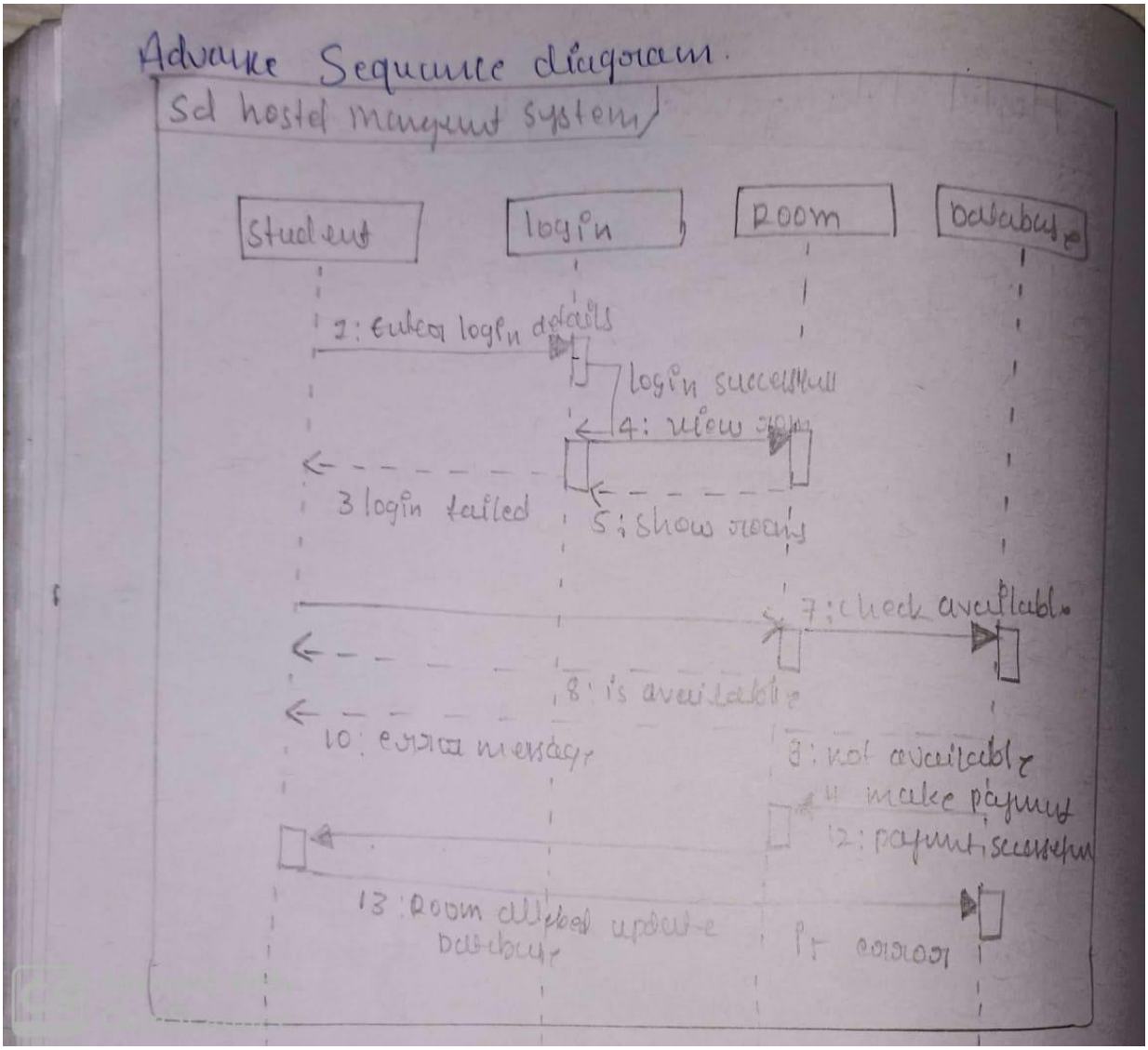
Advance use case

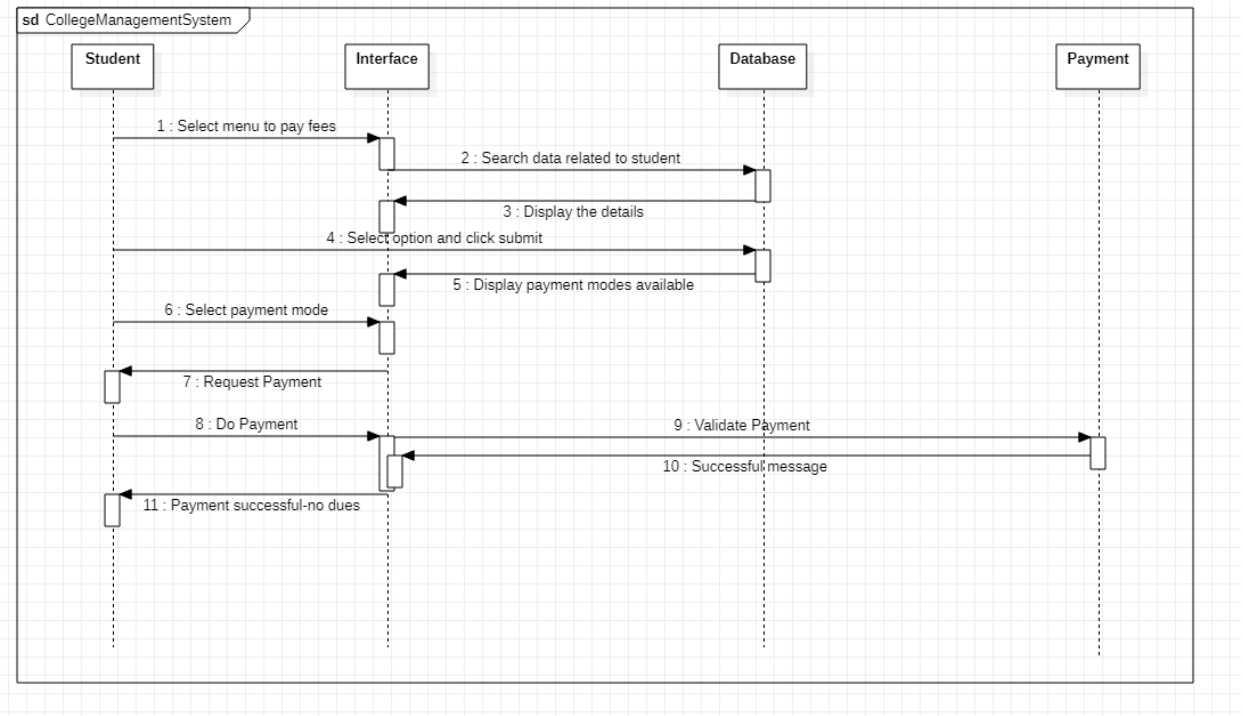
1) College information system :



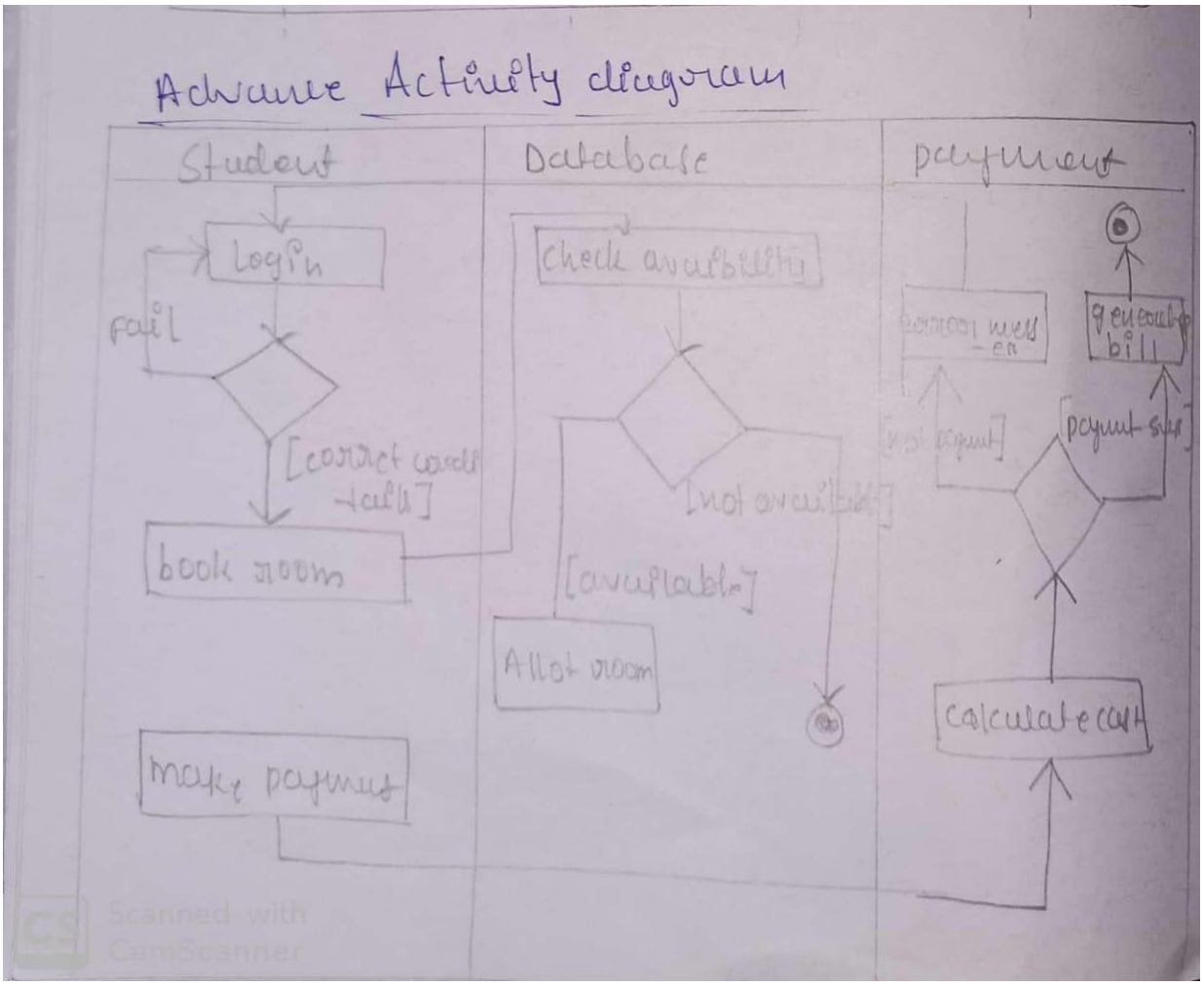


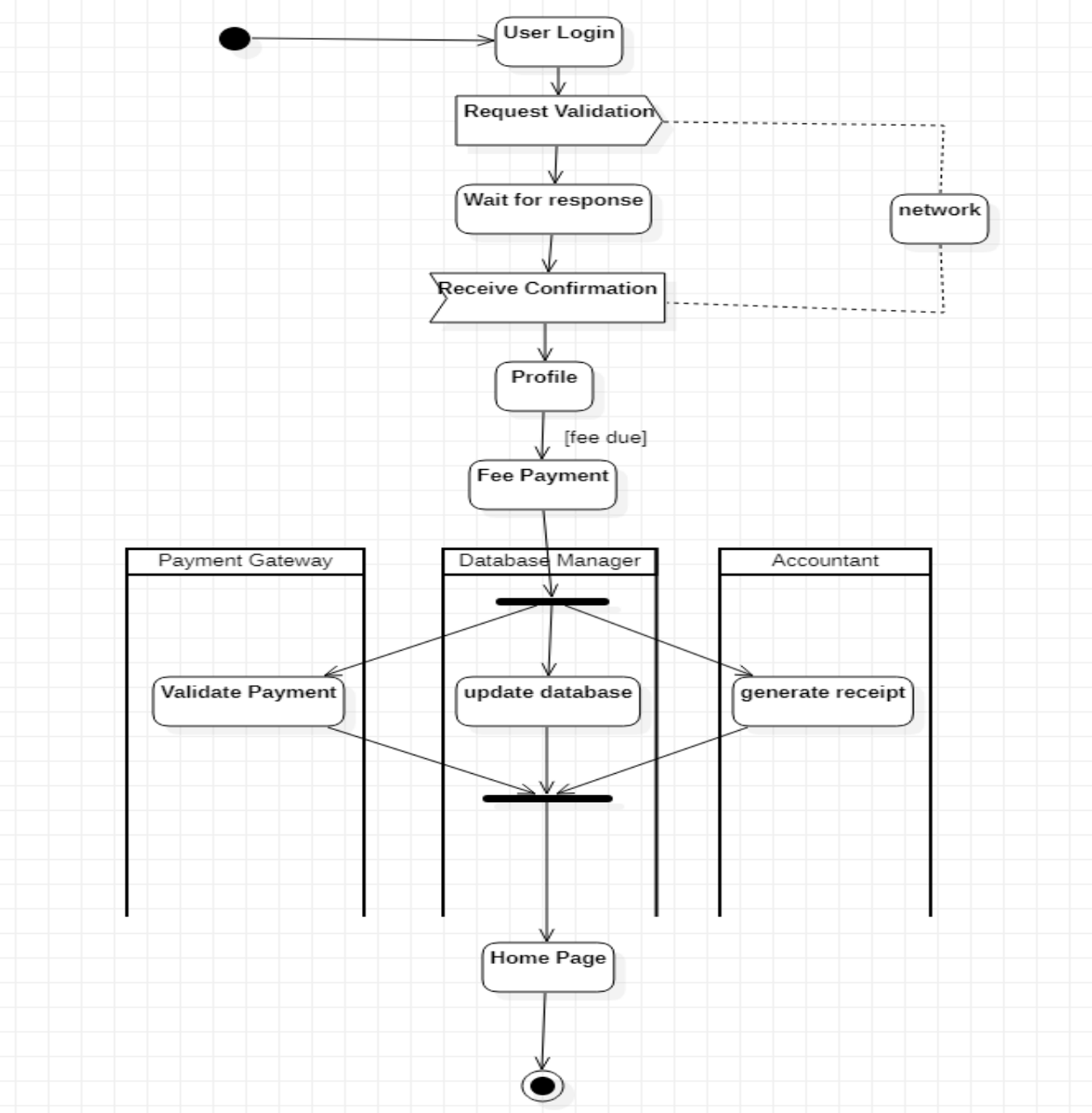
e) Sequence Diagram:


Advance Sequence diagram.
Sd hostel mangemt system
student
login
room
database
1: enter login details
login successful
4: view room
3 login failed
5: show rooms
7: check available
8: is available
10: error message
9: not available
11 make payment
12: payment success
13: Room allotted update database
14: confirm



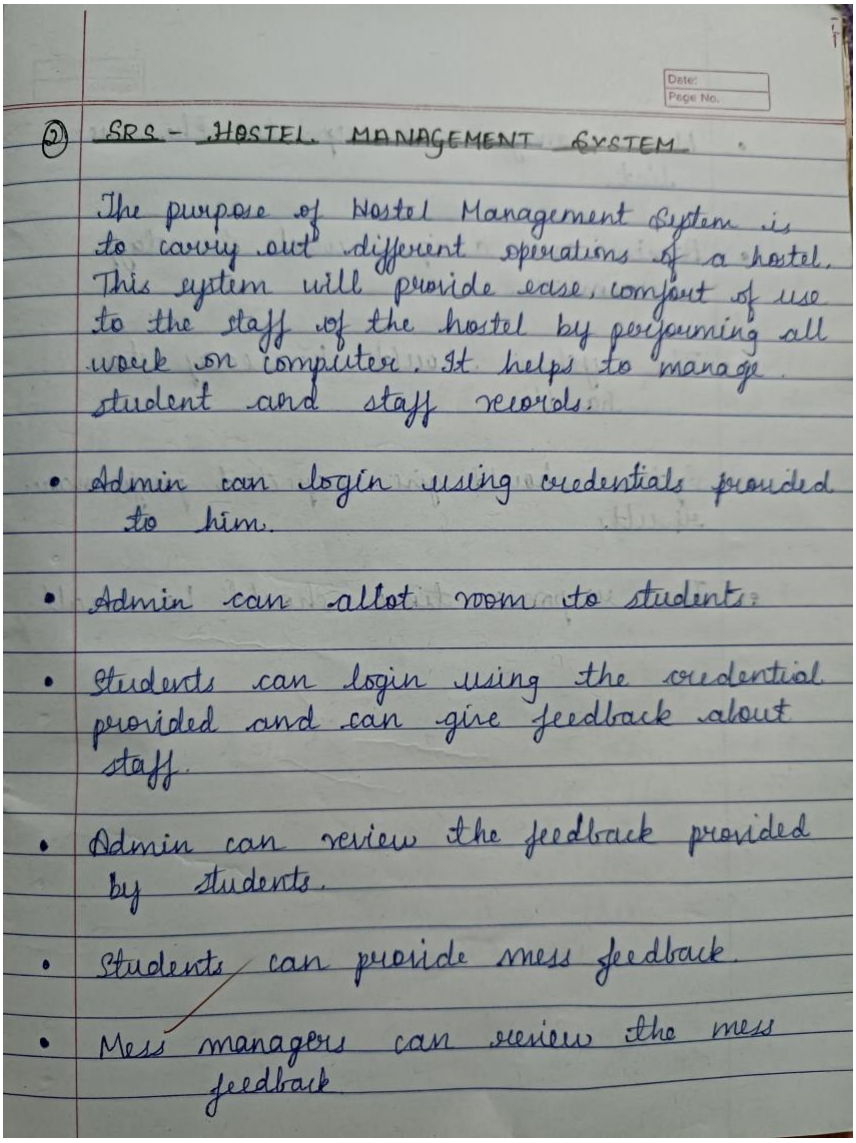
f) Activity Diagram:


Advance Activity diagram
Student
Database
payment
Login
fail
[correct cards
-fail]
book room
check availability
[not available]
[available]
Allot room
make payment
calculate cost
payment succ
generate bill


User Login
Request Validation
Wait for response
network
Receive Confirmation
Profile
[fee due]
Fee Payment
Payment Gateway
Database Manager
Accountant
Validate Payment
update database
generate receipt
Home Page

2. Hostel Management System-

a) SRS:

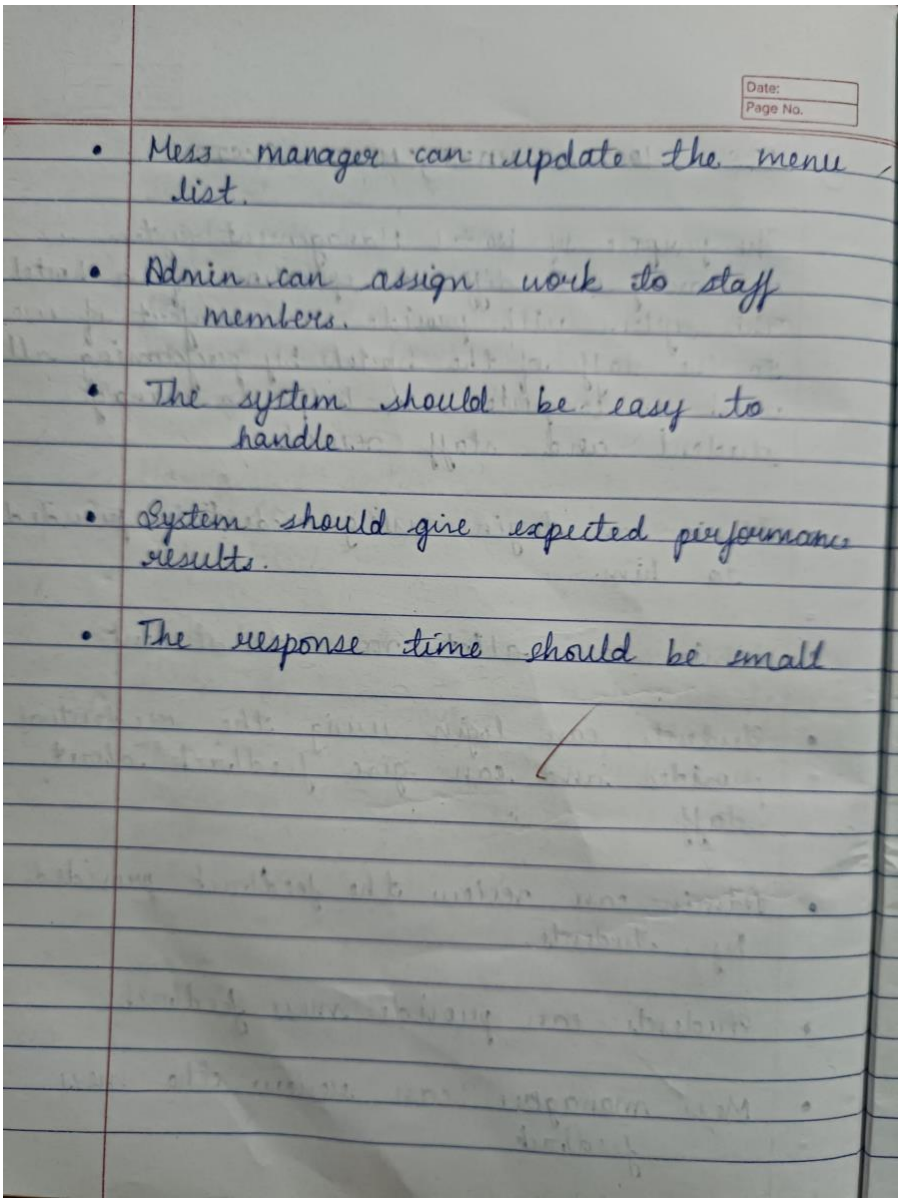


Date:
Page No.

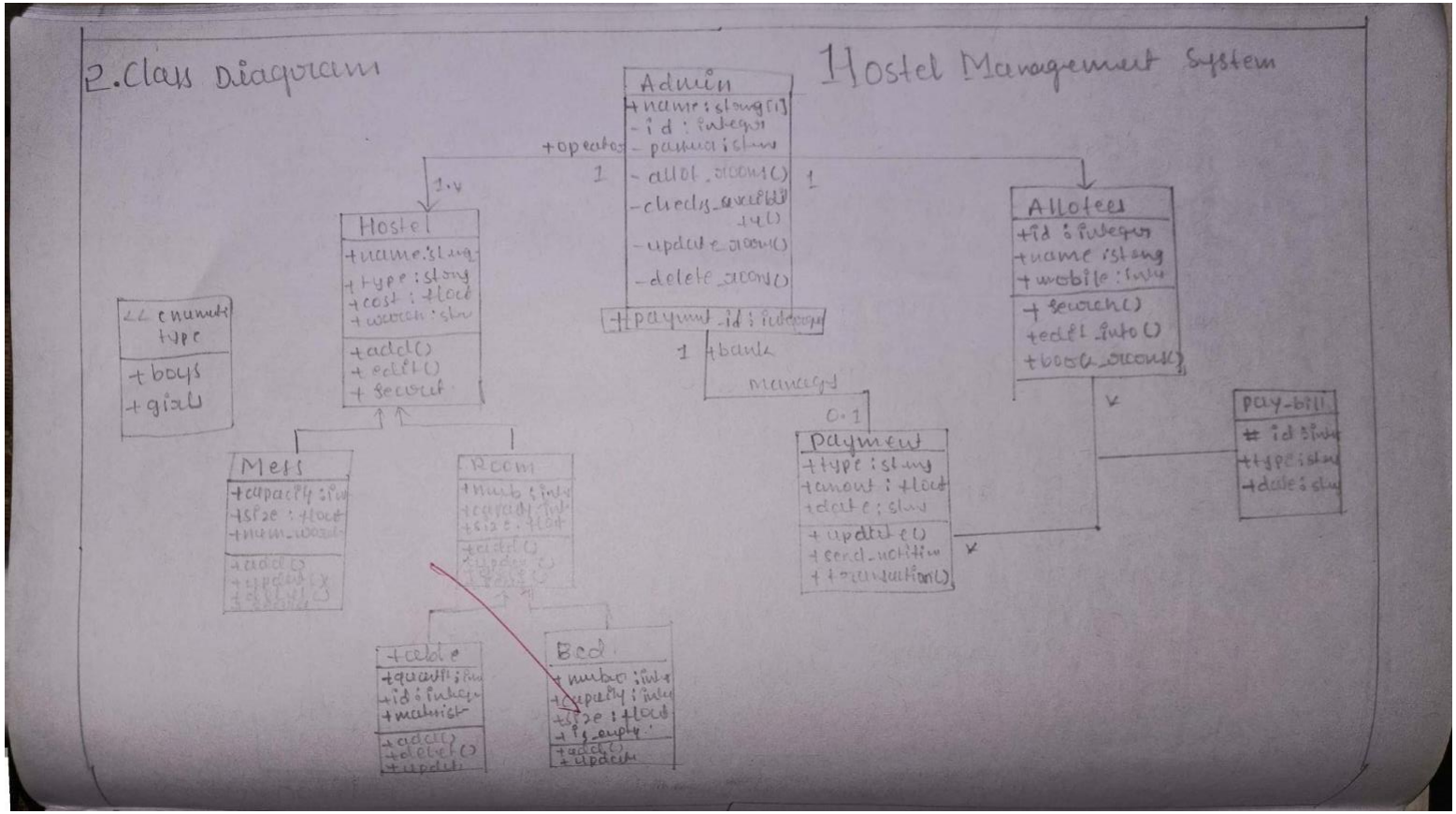
② SRS - HOSTEL MANAGEMENT SYSTEM

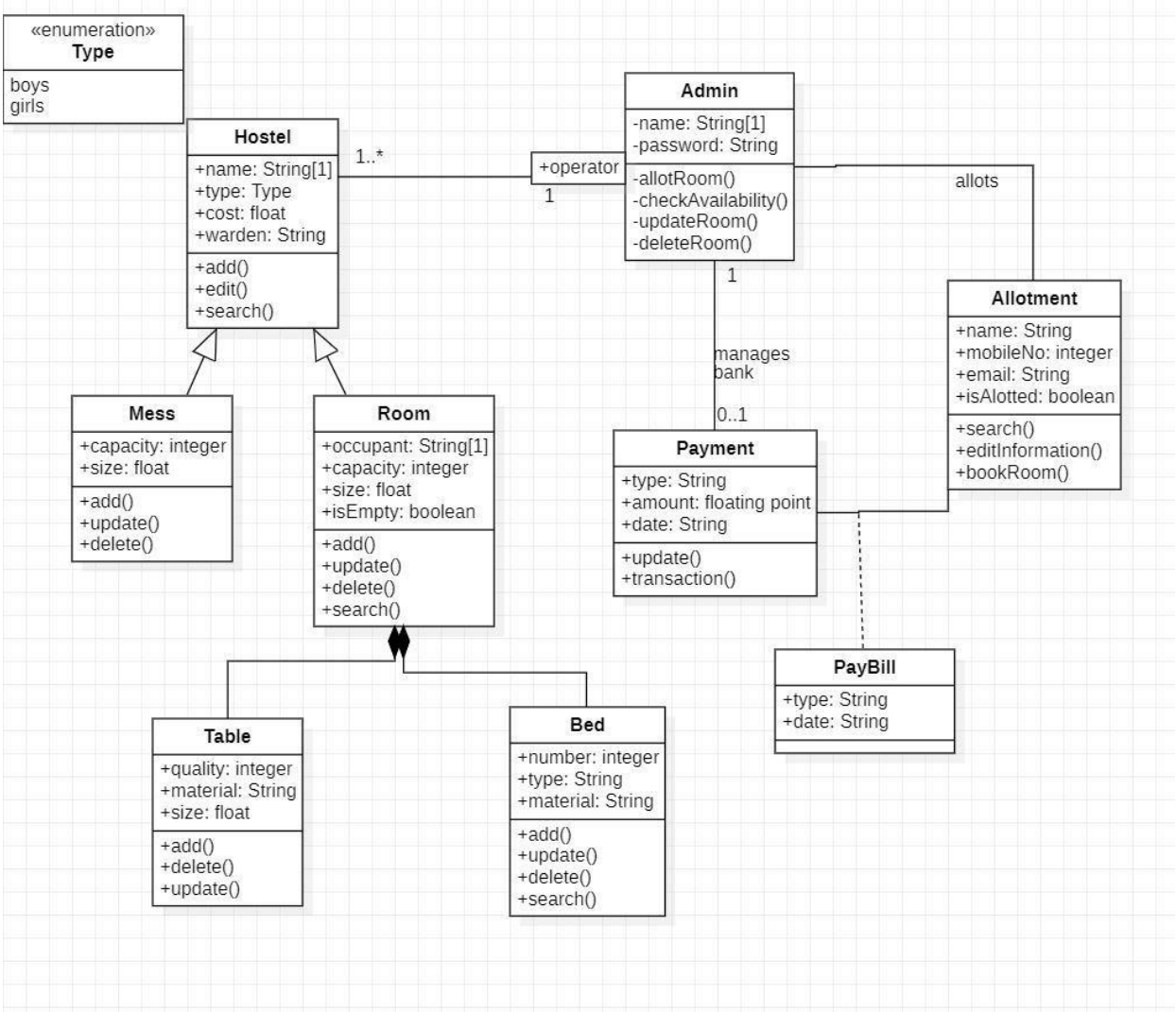
The purpose of Hostel Management System is to carry out different operations of a hostel. This system will provide ease, comfort of use to the staff of the hostel by performing all work on computer. It helps to manage student and staff records.

- Admin can login using credentials provided to him.
- Admin can allot room to students.
- Students can login using the credential provided and can give feedback about staff.
- Admin can review the feedback provided by students.
- Students can provide mess feedback.
- Mess managers can review the mess feedback.

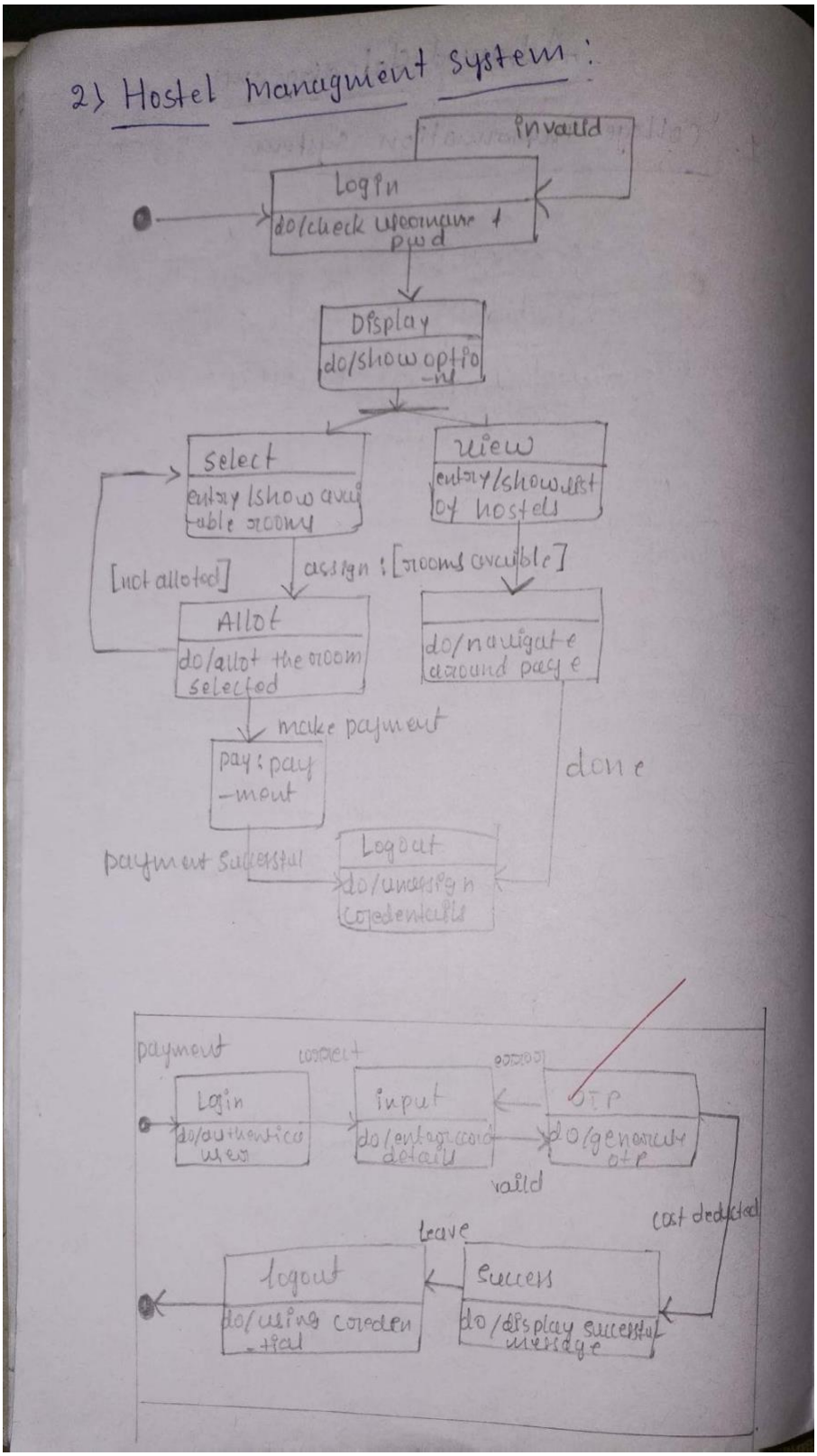


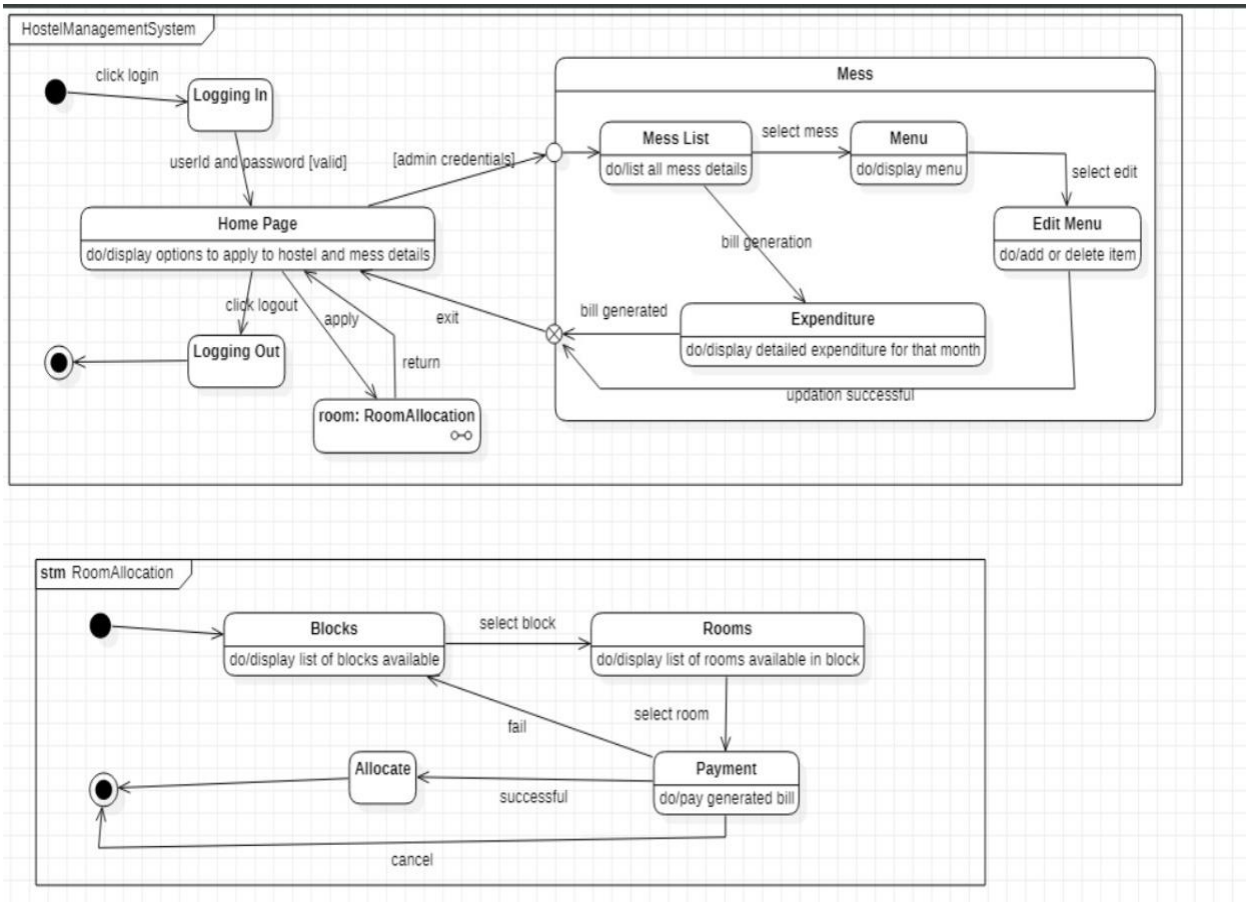
b) Advance Class Diagram:


2. Class Diagram
Hostel Management System
Admin
+name : string[]
- id : integer
- passwd : string
- allot_rooms()
- checks_available()
- update_room()
- delete_room()
+operates
1
1..*
1
Hostel
+name : string
+type : string
+cost : float
+address : string
+add()
+edit()
+search()
<<enum>>
type
+boys
+girls
Mess
+capacity : int
+size : float
+num_waiters
+add()
+update()
+display()
Room
+numb : int
+capacity : int
+size : float
+add()
+update()
+display()
Table
+quantity : int
+id : integer
+material
+add()
+delete()
+update()
Bed
+numbo : int
+capacity : int
+size : float
+is_empty
+add()
+update
+payment_id : integer
1
+bank
manages
0..1
Payment
+type : string
+amount : float
+date : string
+update()
+send_notification()
+transaction()
Allotees
+id : integer
+name : string
+mobile : int
+search()
+edit_info()
+book_room()
pay-bill
id : int
+type : string
+date : string



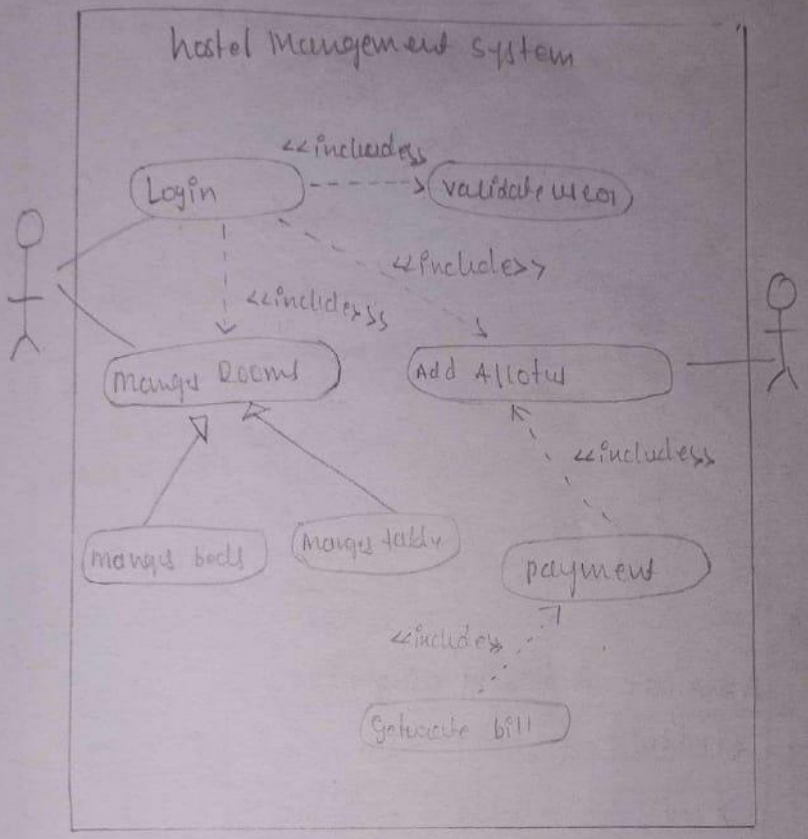
c) Advance State Diagram:

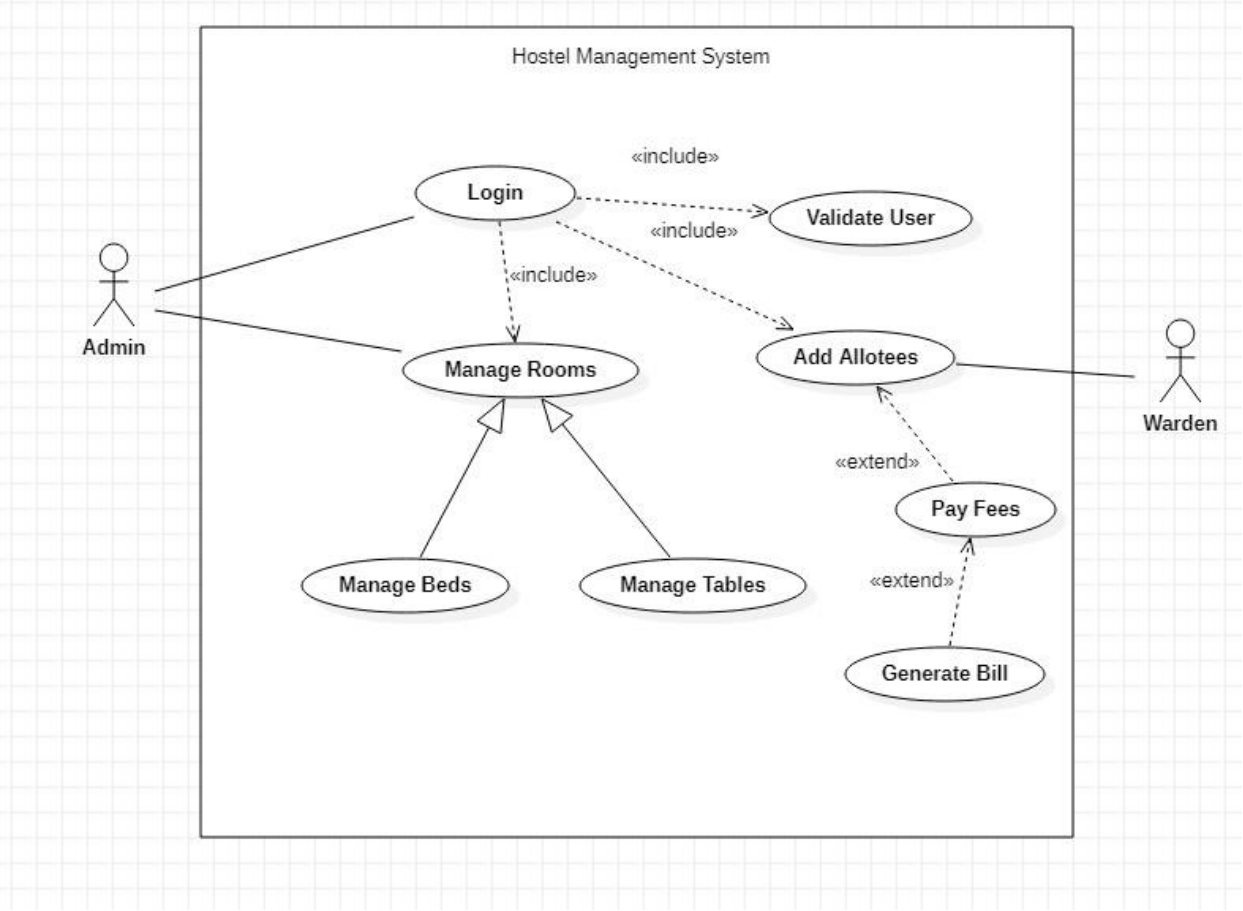

2) Hostel Managment System :
Invalid
Login
do/check username & pwd
Display
do/show options
select
entry/show available rooms
view
entry/show list of hostels
[not alloted]
assign : [rooms available]
Allot
do/allot the room selected
do/navigate around page
make payment
pay : payment
done
payment successful
Logout
do/unassign credentials
payment
correct
Login
do/authenticate user
input
do/enter card details
OTP
do/generate otp
valid
cost deducted
leave
logout
do/using credential
Success
do/display success message



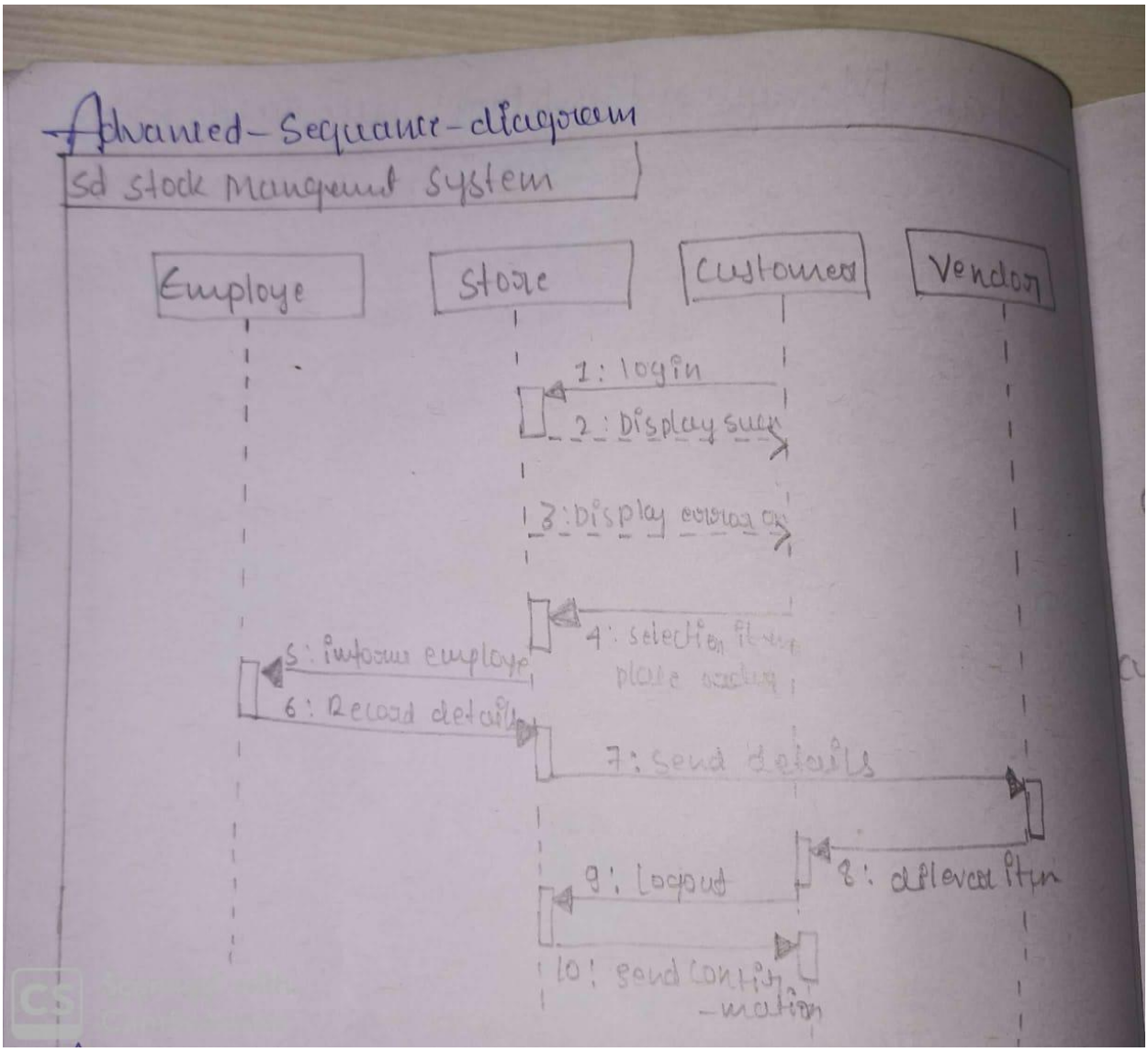
d) Advance Use Case Diagram:

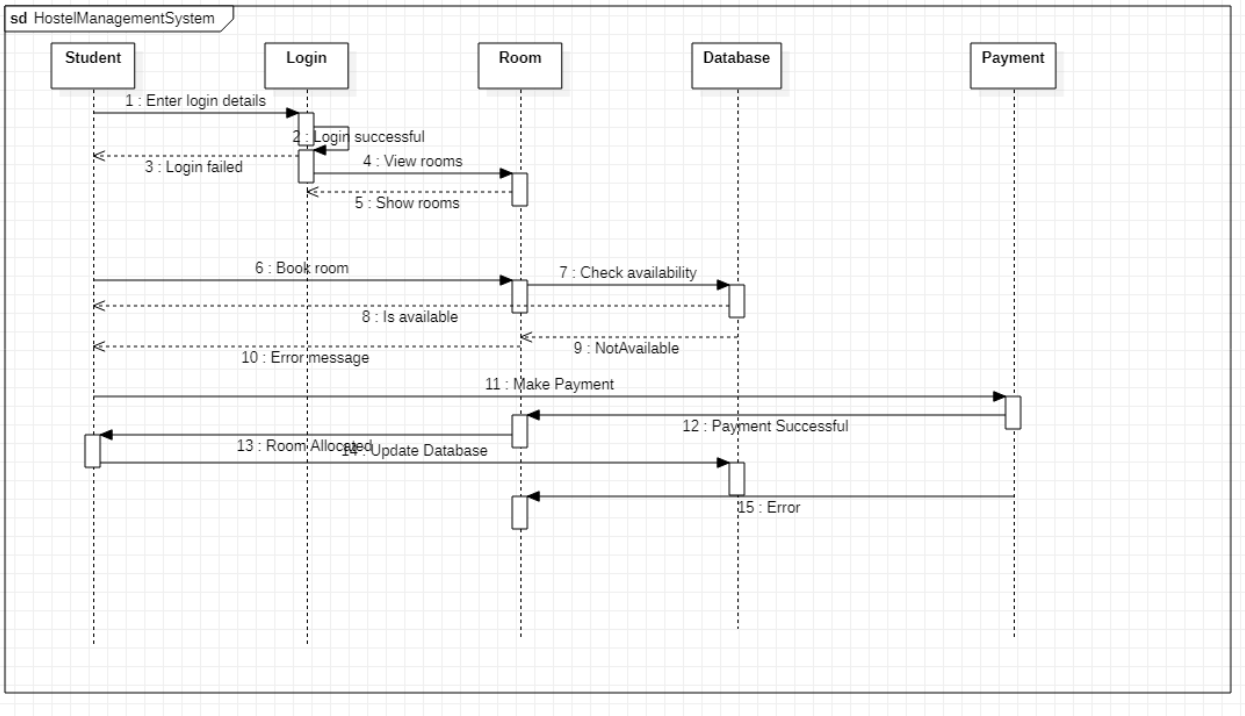
2) Hostel Mangament System:



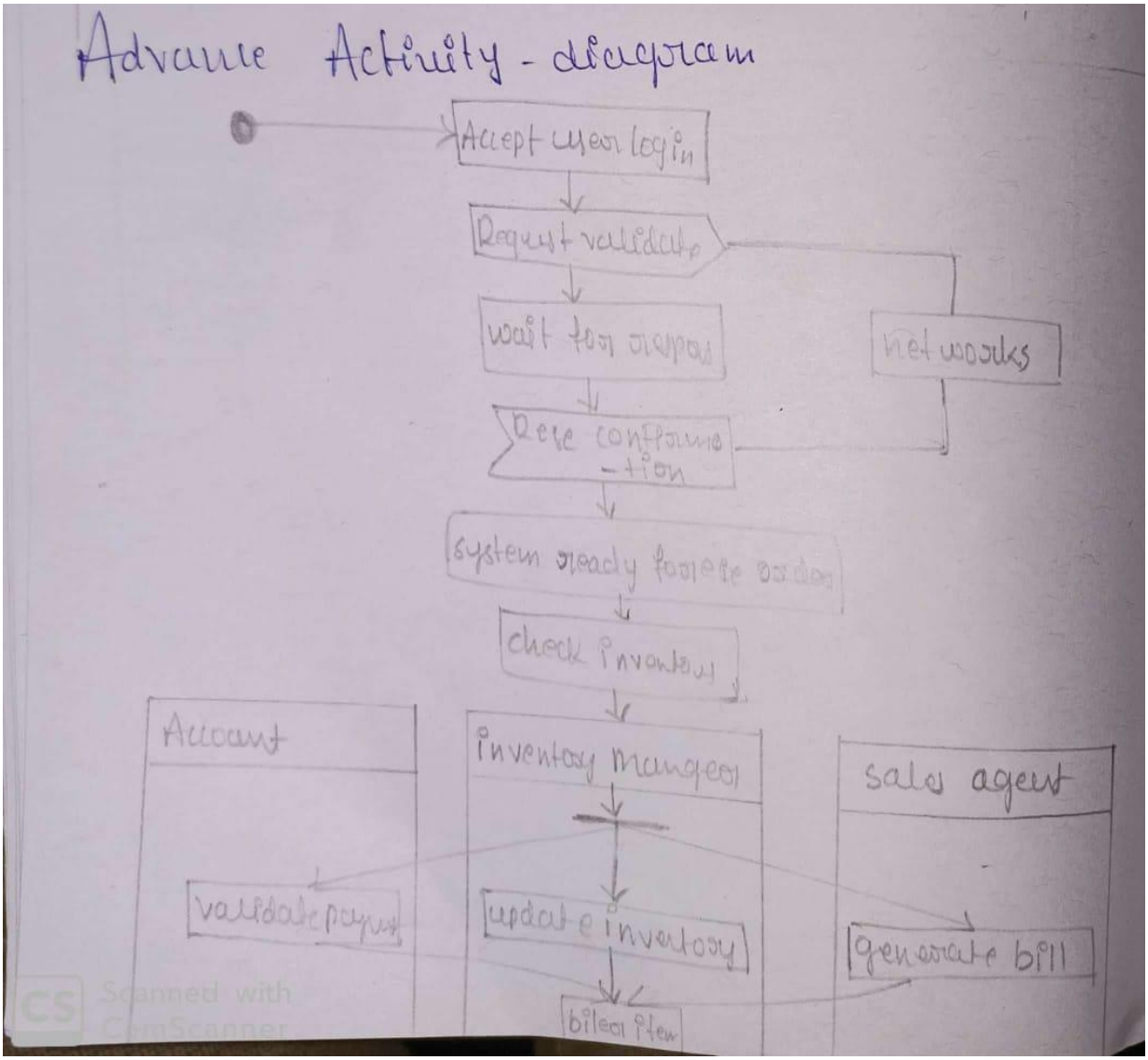


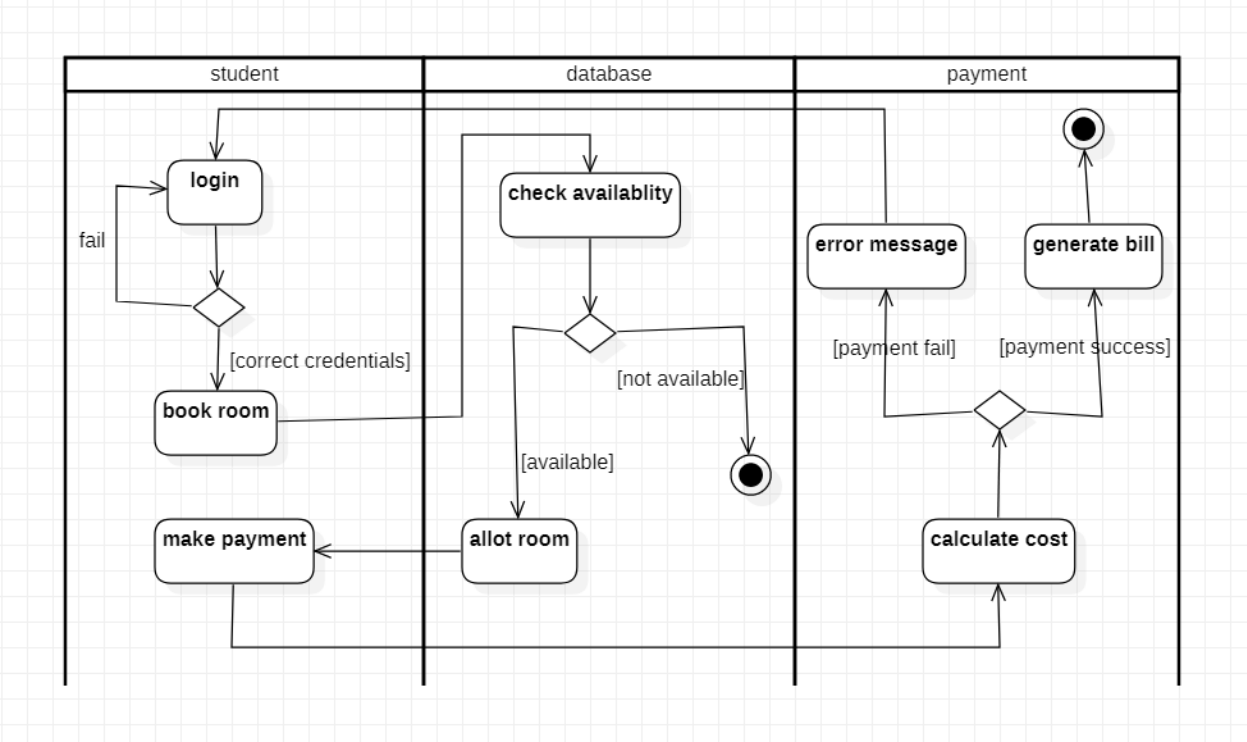
e) Sequence Diagram:


Advanced-Sequance-diagram
sd stock Management System
Employe
Store
Customer
Vendor
1: login
2: Display succ
3: Display
4: selection
place
5: Inform employe
6: Reload details
7: Send details
8: deliver item
9: Logout
10: send confir
-mation



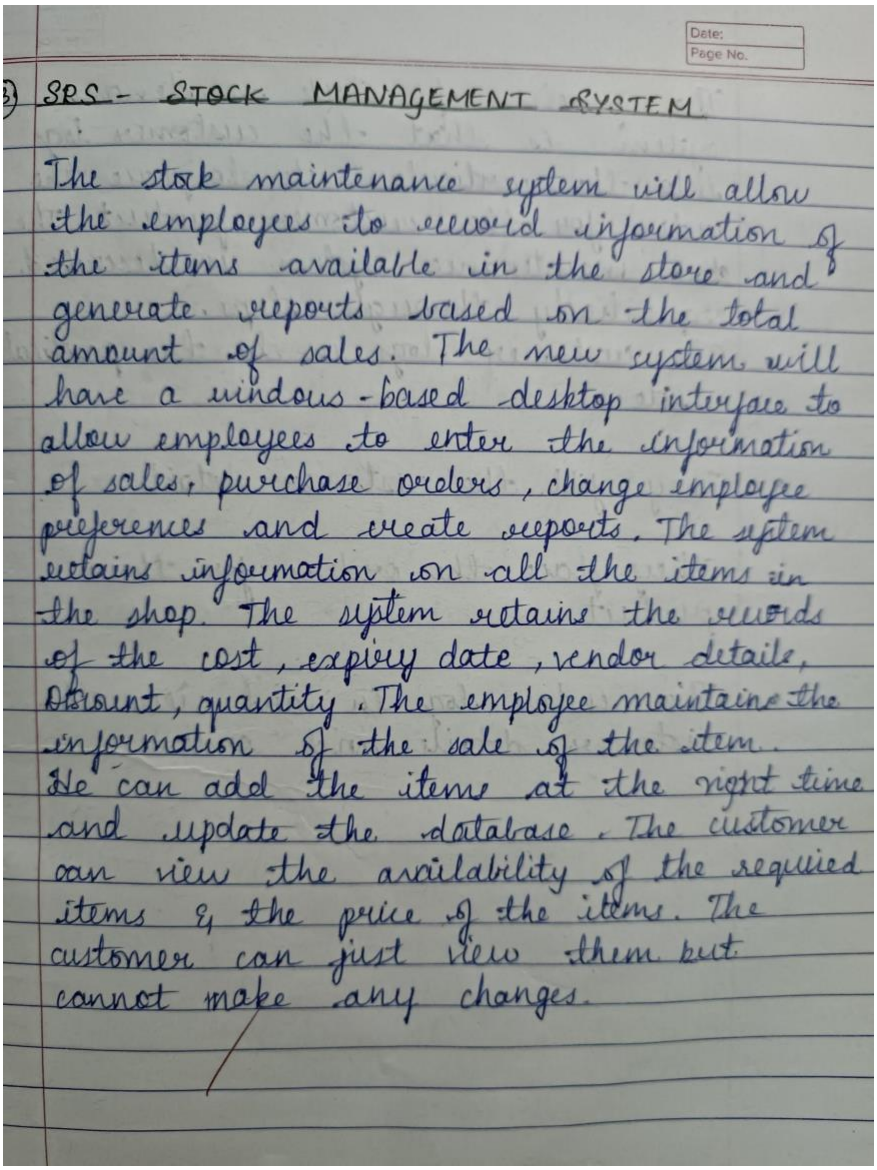
f) Activity Diagram:


Advance Activity - diagram
Accept user login
Request validate
wait for response
network
Role conformation
system ready for role order
check inventory
Account
inventory manager
sale agent
validate payment
update inventory
generate bill
bill item


student
database
payment
login
fail
[correct credentials]
book room
check availablity
[not available]
[available]
allot room
make payment
error message
generate bill
[payment fail]
[payment success]
calculate cost

3. Stock Maintenance System-

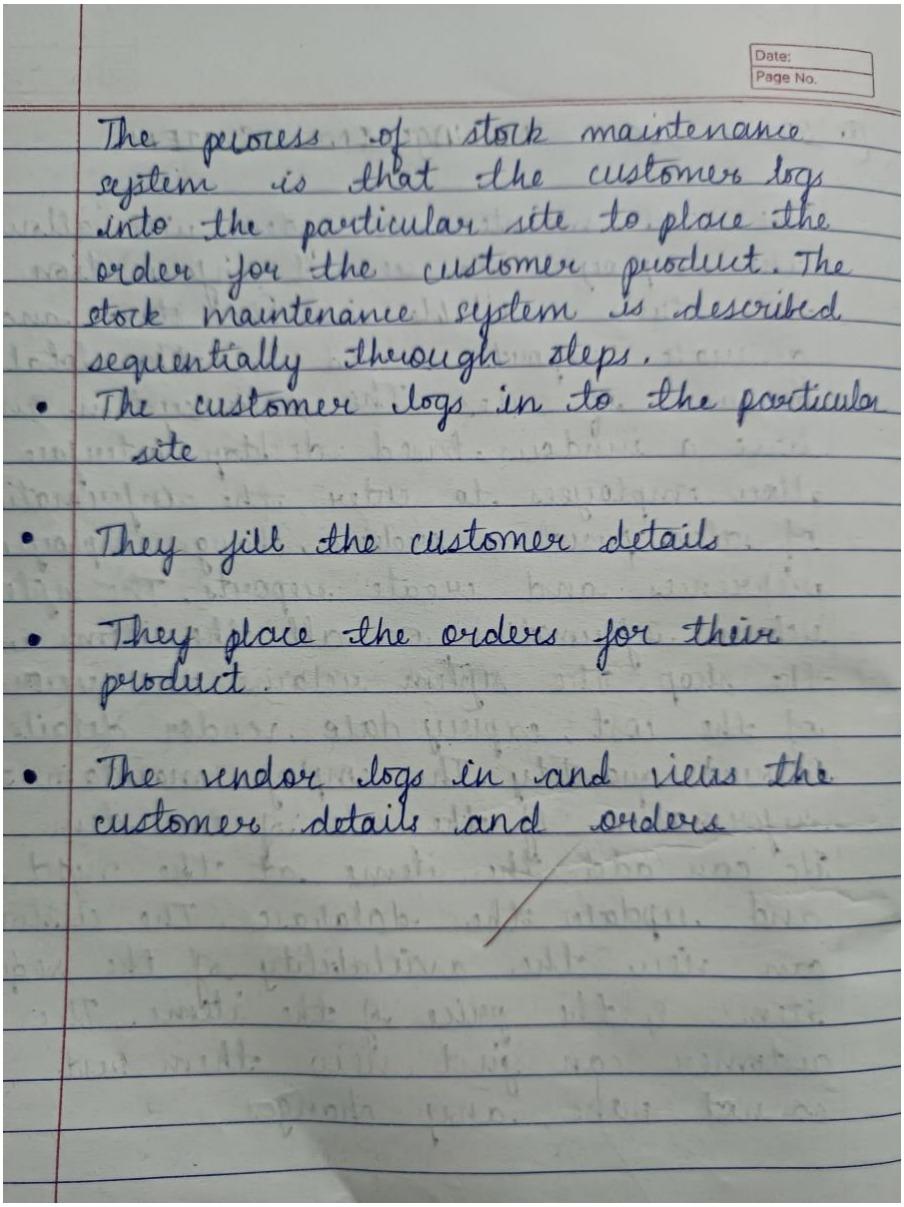
a) SRS:



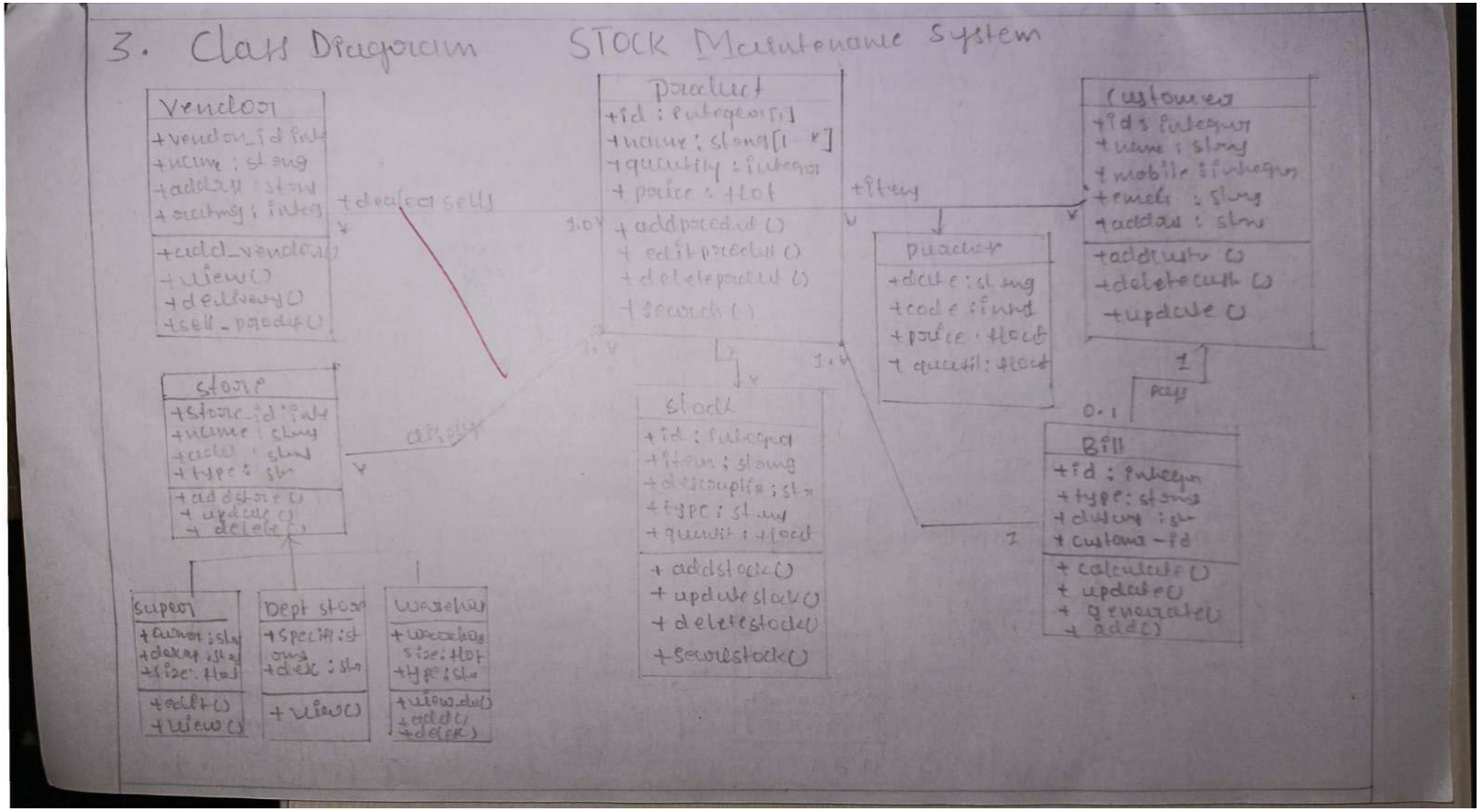
Date:
Page No.

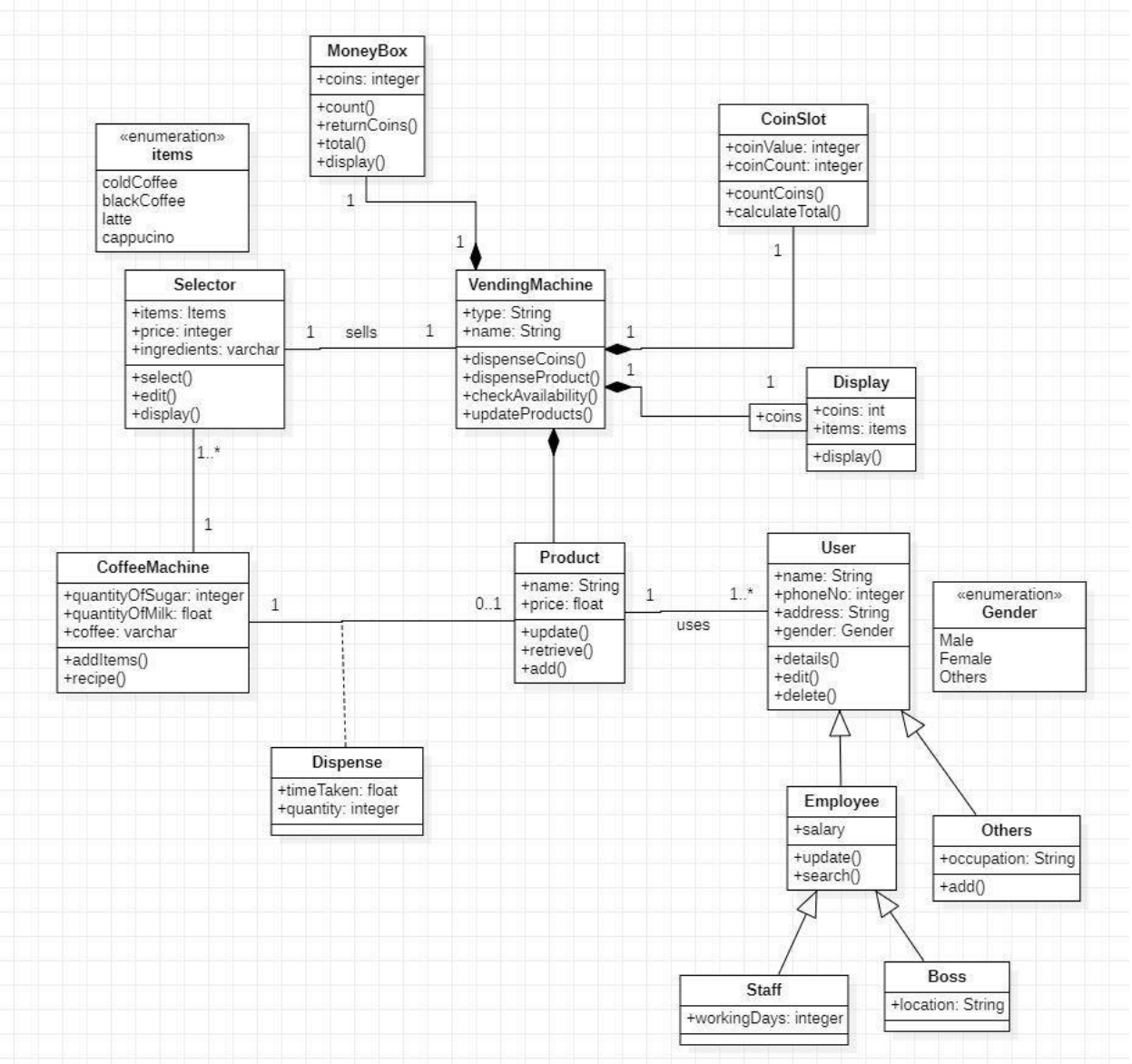
3) SRS - STOCK MANAGEMENT SYSTEM

The stock maintenance system will allow the employees to record information of the items available in the store and generate reports based on the total amount of sales. The new system will have a windows-based desktop interface to allow employees to enter the information of sales, purchase orders, change employee preferences and create reports. The system retains information on all the items in the shop. The system retains the records of the cost, expiry date, vendor details, discount, quantity. The employee maintains the information of the sale of the item. He can add the items at the right time and update the database. The customer can view the availability of the required items & the price of the items. The customer can just view them but cannot make any changes.



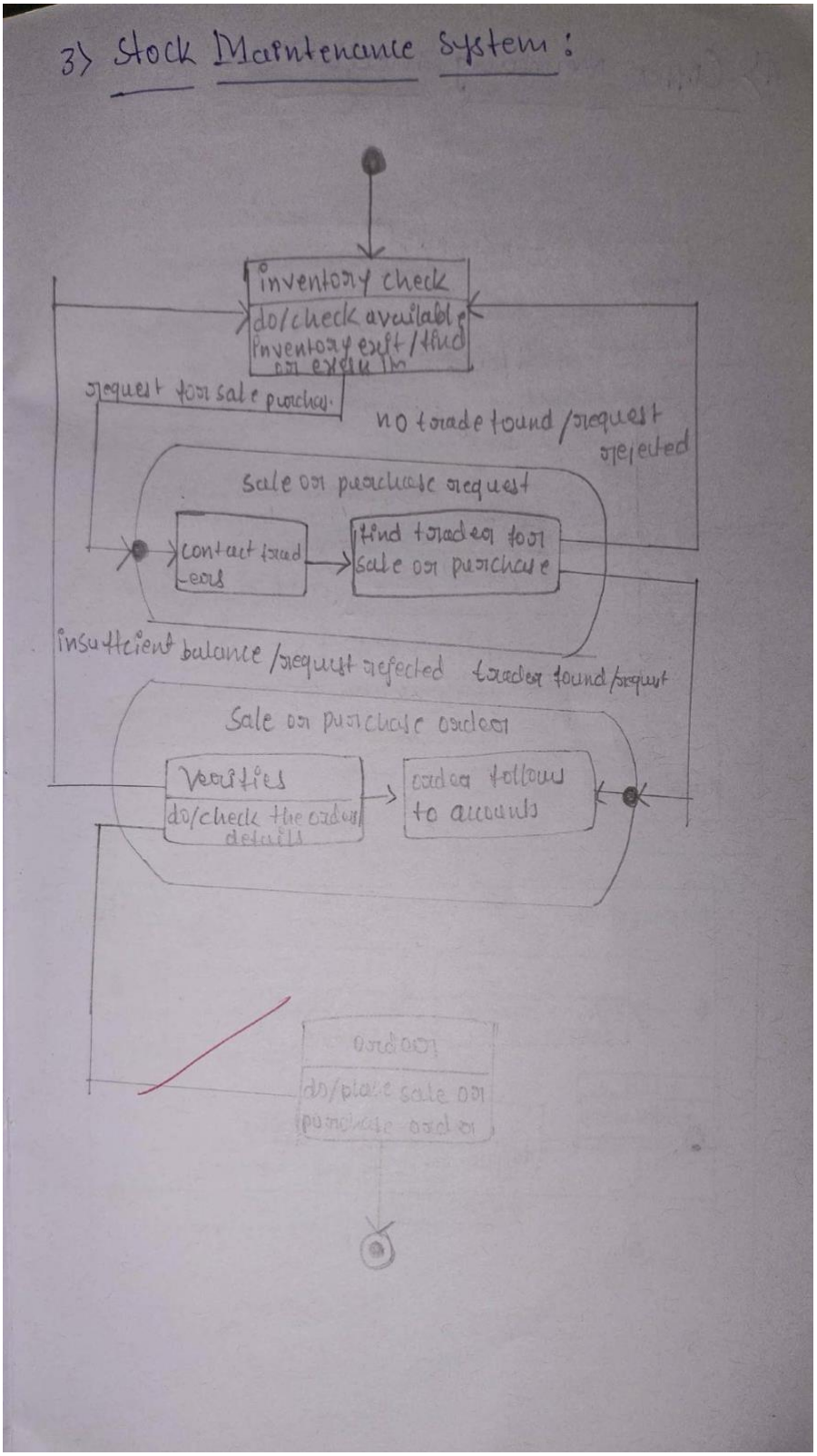
b) Advance Class Diagram:

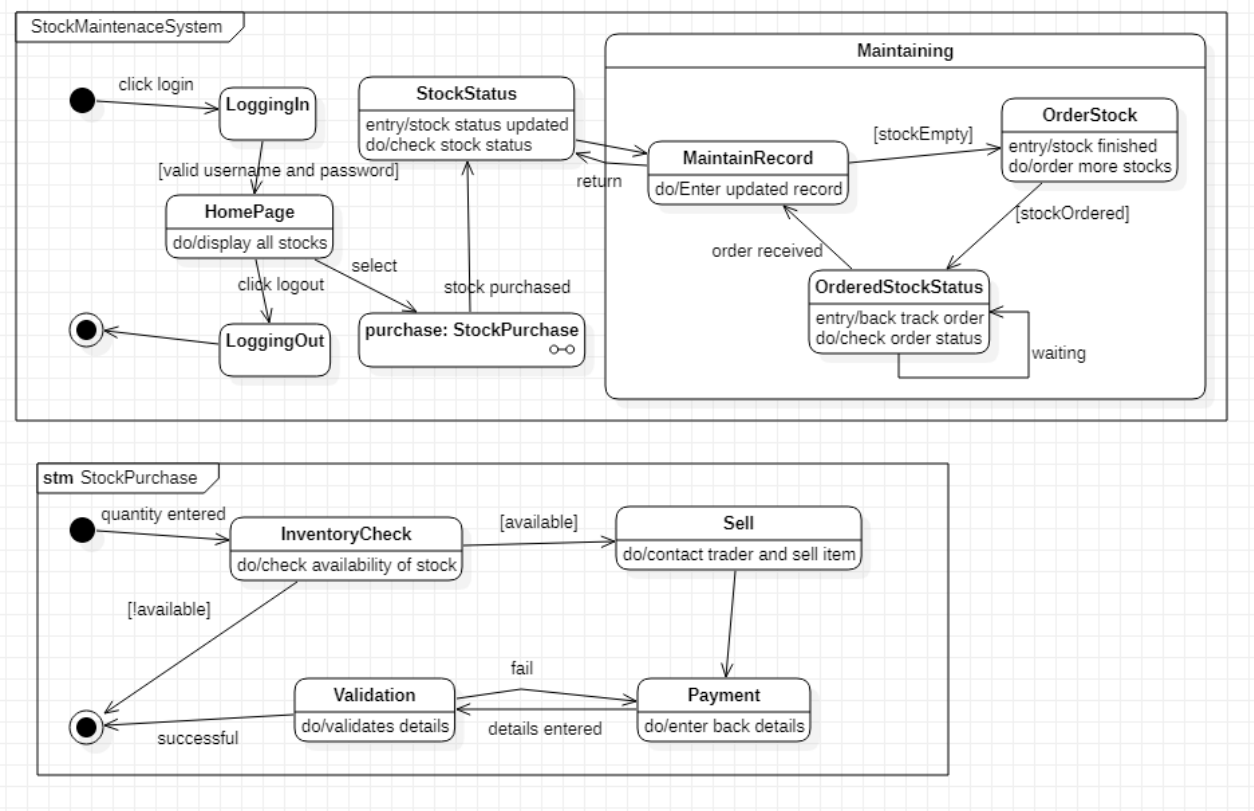

3. Class Diagram STOCK Maintenance System
Vendor
+vendor_id int
+name : string
+address : string
+contact : integer
+add_vendor()
+view()
+delivery()
+sell_prod()
+deal sell
Product
+id : integer[1]
+name : string[1-*]
+quantity : integer
+price : float
+addproduct()
+ editproduct()
+deleteproduct()
+search()
+item
Customer
+id : integer
+name : string
+mobile : integer
+email : string
+addr : string
+addcust()
+deletecust()
+update()
Store
+store_id : int
+name : string
+city : string
+type : str
+addstore()
+update()
+delete()
display
Stock
+id : integer
+item : string
+description : str
+type : string
+quantity : float
+addstock()
+updatestock()
+deletestock()
+securestock()
Super
+owner : str
+area : str
+size : float
+edit()
+view()
Dept store
+specify : string
+desc : str
+view()
Warehouse
+warehouse size : float
+type : str
+view_del()
+add()
+del()
Purchase
+date : string
+code : int
+price : float
+quantity : float
Bill
+id : integer
+type : string
+date : str
+customer_id
+calculate()
+update()
+ generate()
+ add()
pays
1
0..1
1..*
1..*
1..*
1..*
1
1



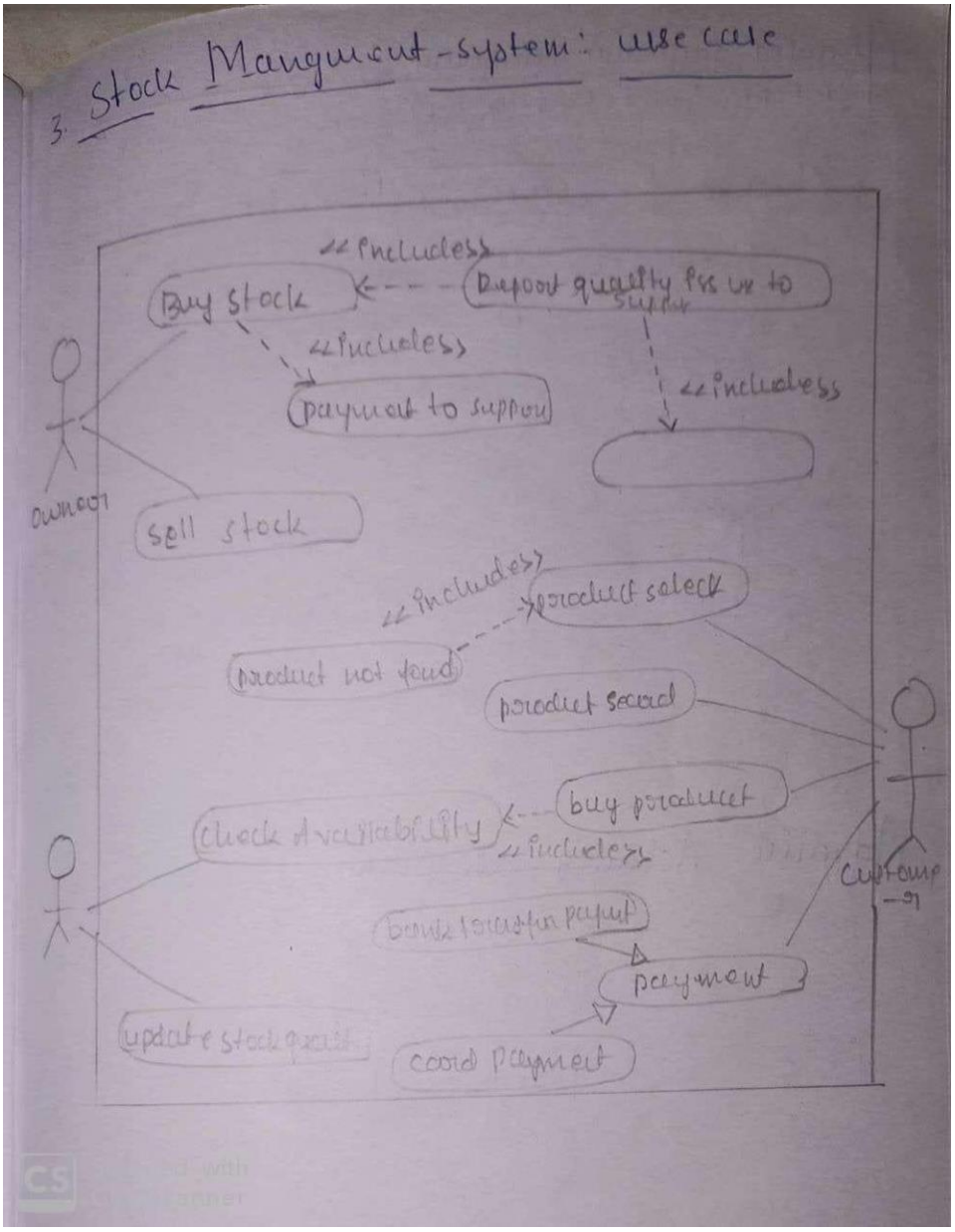
c) Advance State Diagram:

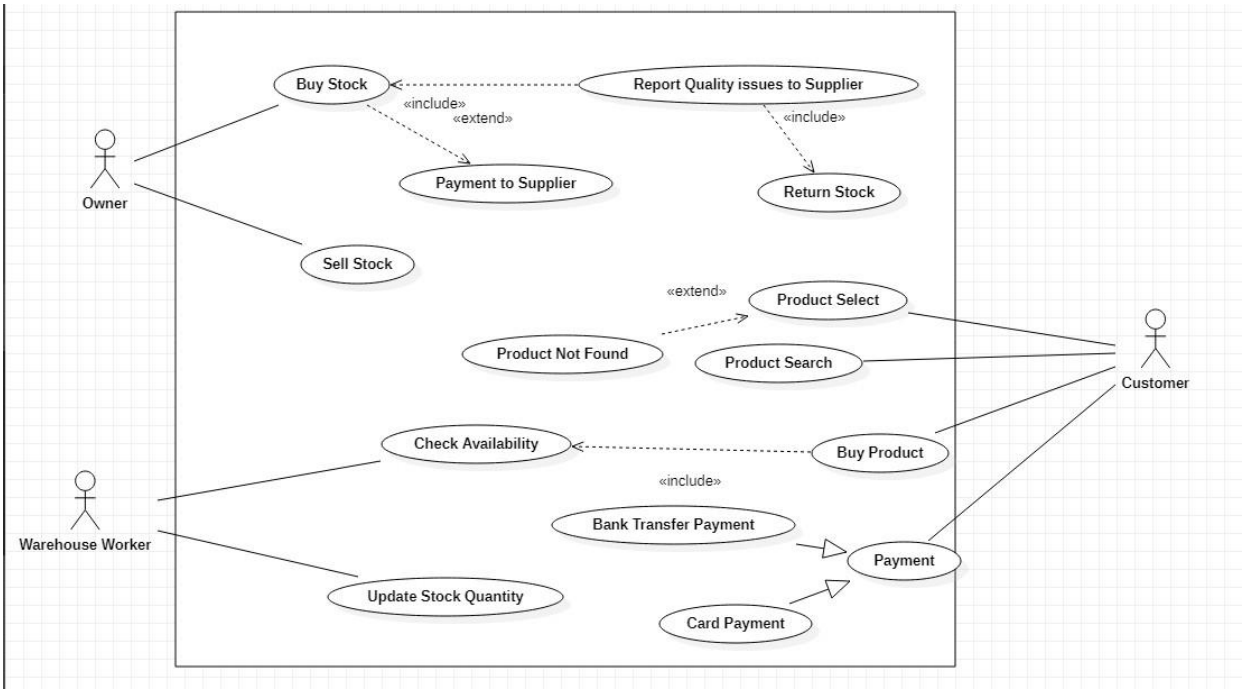
3) Stock Maintenance System :



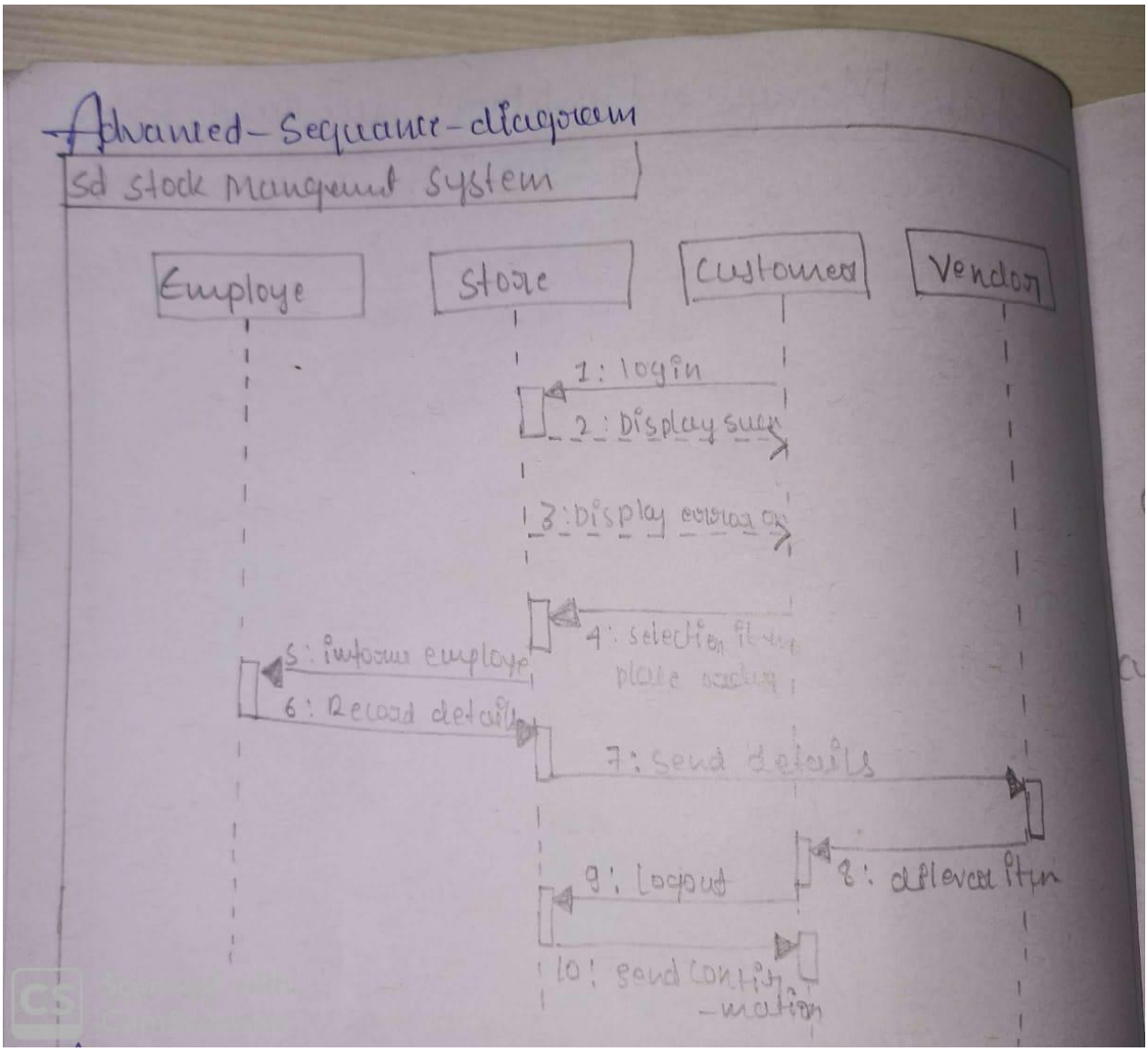


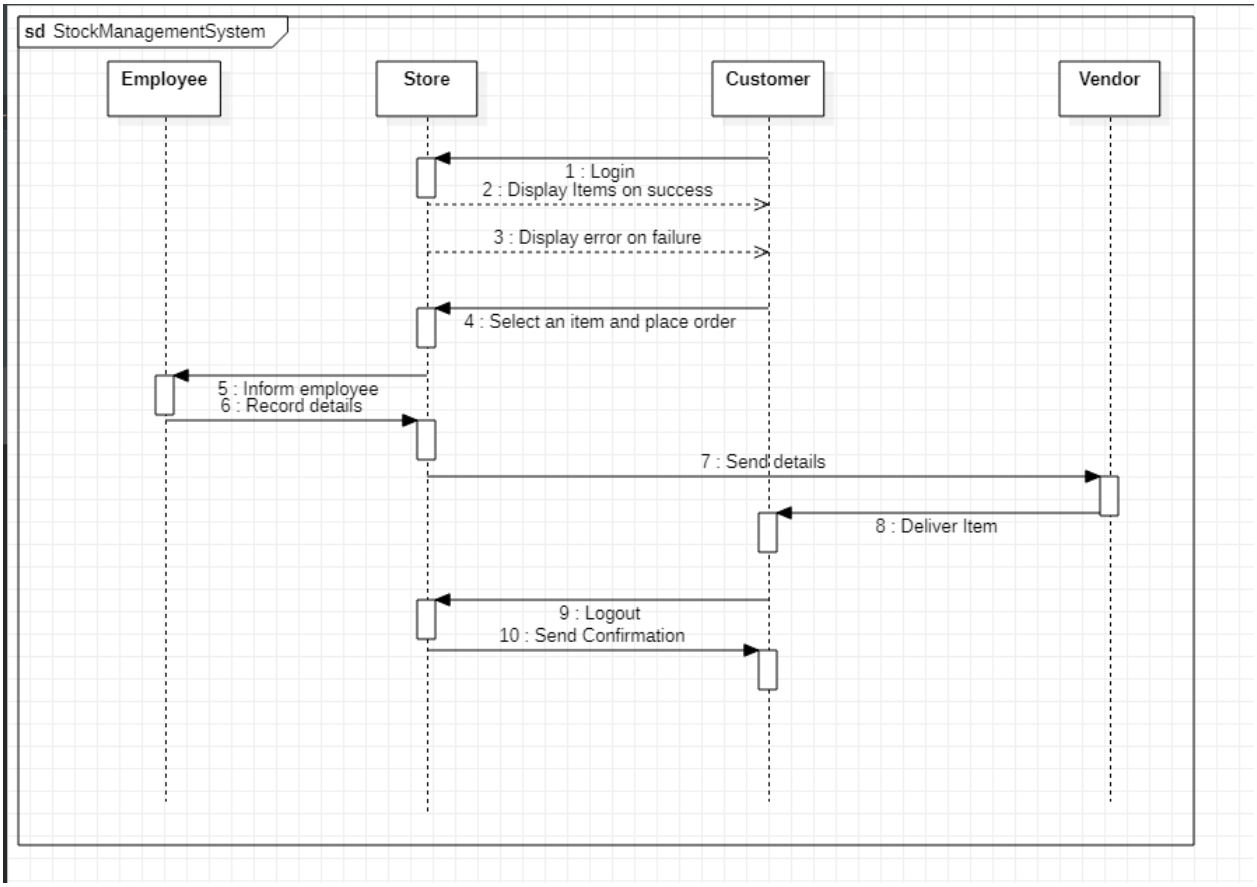
d) Advance Use Case Diagram:


3. Stock Manguicut -system: use case
<<Includes>>
Buy stock
Deposit quality prs ur to
<<Includes>>
payment to suppou
<<Includes>>
owner
sell stock
<<Includes>>
product select
product not found
product secard
buy product
check Availability
<<Includes>>
customer
payment
update stock quait
coord payment

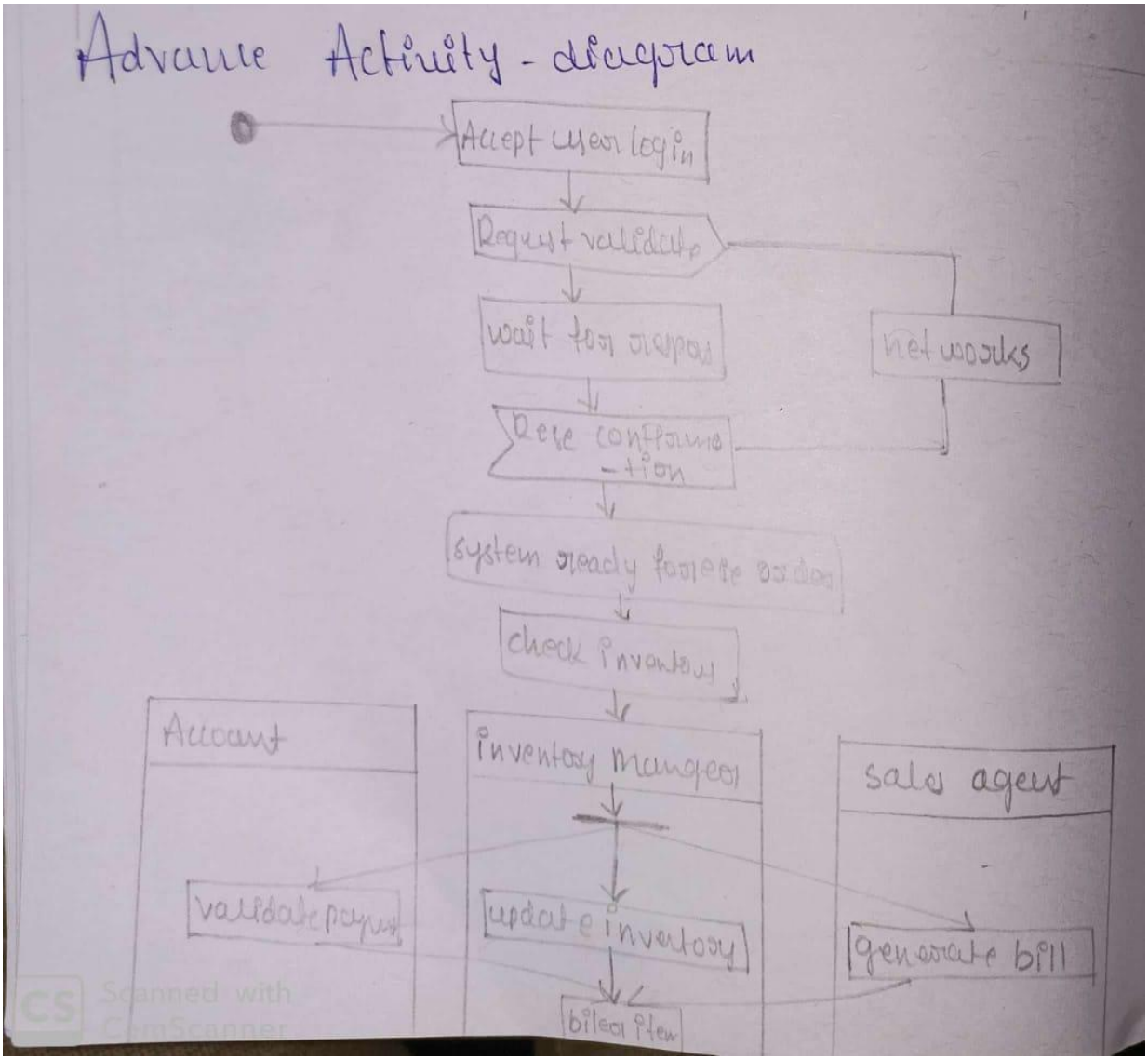


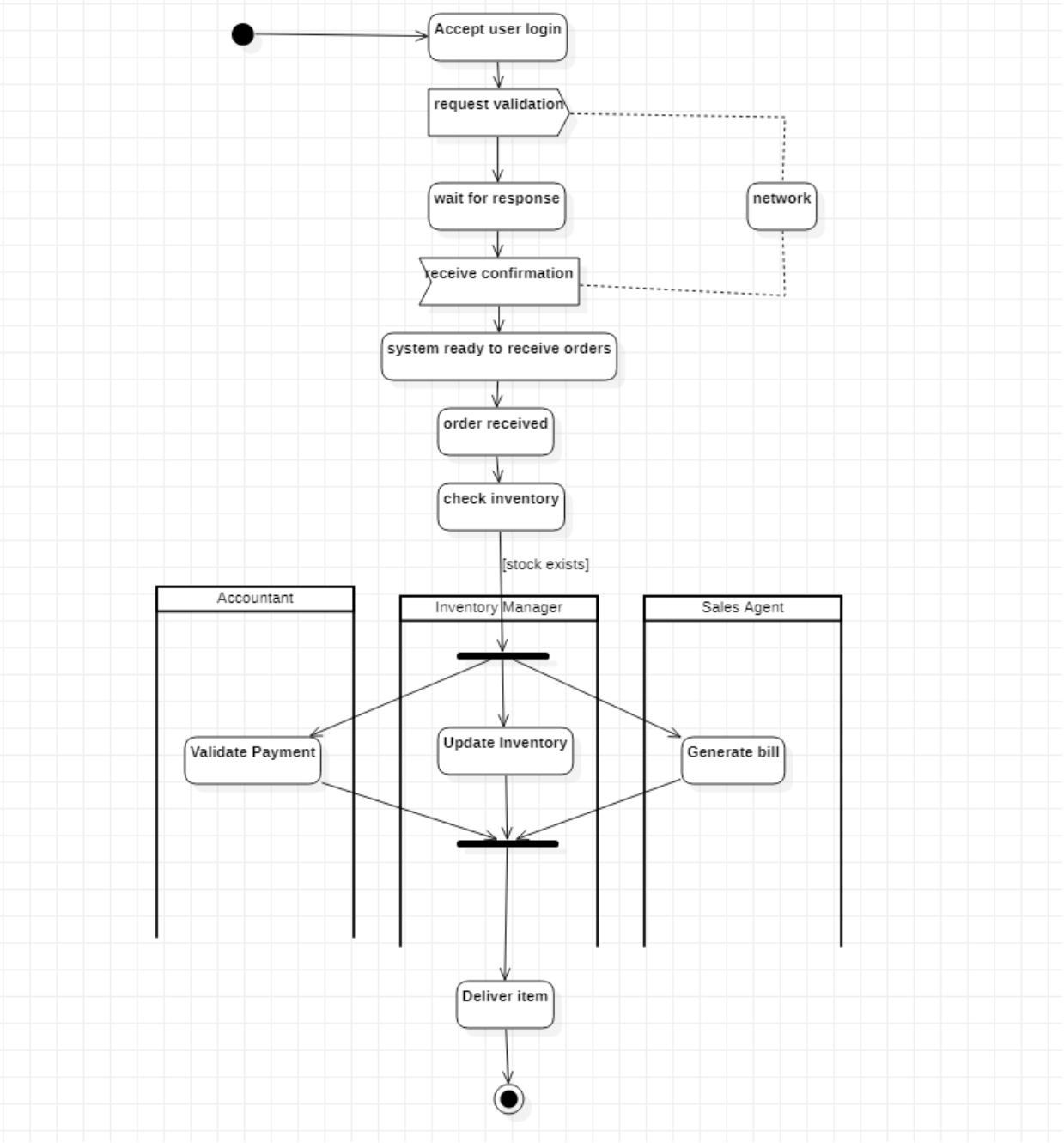
e) Sequence Diagram:


Advanced-Sequance-diagoream
sd stock Manageent System
Employe
Store
Customer
Vendor
1: login
2: Display succ
3: Display
4: selection
5: Inform employe
6: Reload details
7: Send details
8: deliver item
9: Logout
10: send confir
-mation



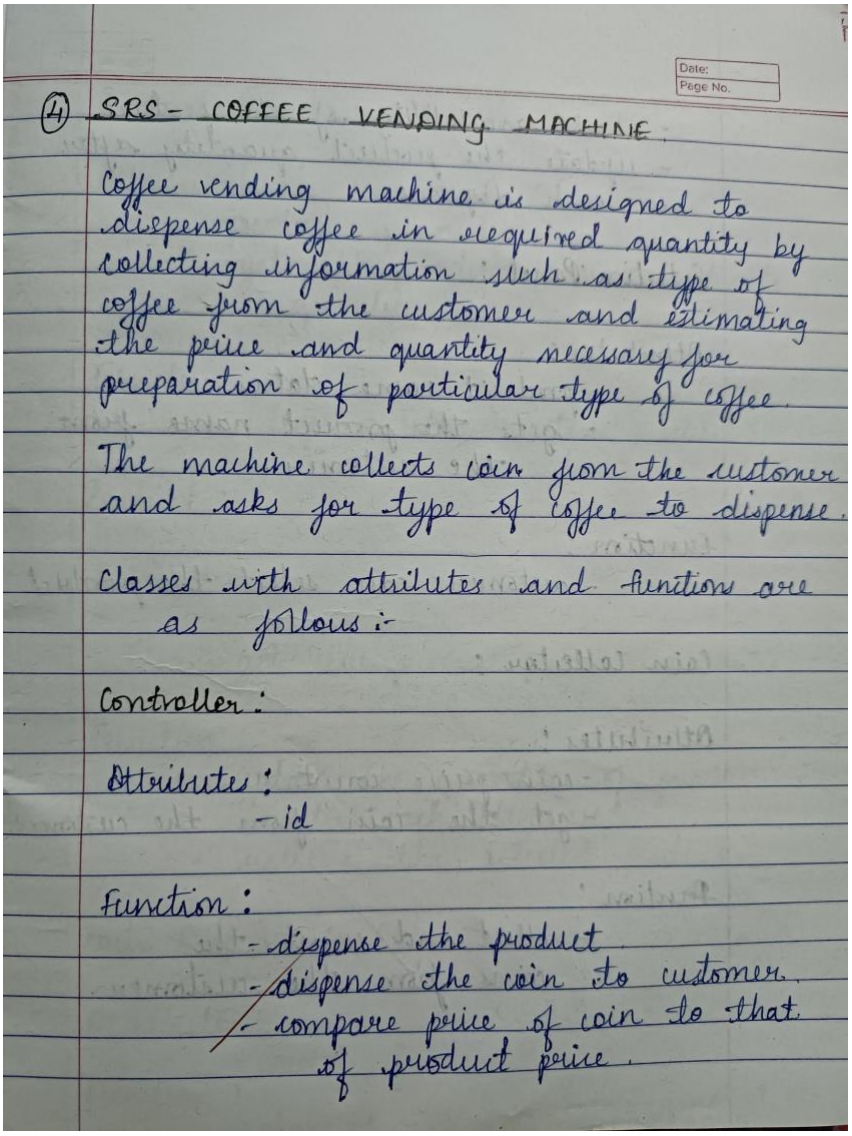
f) Activity Diagram:


Advance Activity - diagram
Accept user login
Request validate
wait for response
network
Role confirmation
system ready for role action
check inventory
Account
inventory manager
sales agent
validate payment
update inventory
generate bill
billed item


Accept user login
request validation
wait for response
network
receive confirmation
system ready to receive orders
order received
check inventory
[stock exists]
Accountant
Inventory Manager
Sales Agent
Validate Payment
Update Inventory
Generate bill
Deliver item

4. Coffee Vending Machine-

a) SRS:



④ SRS - COFFEE VENDING MACHINE.

Coffee vending machine is designed to dispense coffee in required quantity by collecting information such as type of coffee from the customer and estimating the price and quantity necessary for preparation of particular type of coffee.

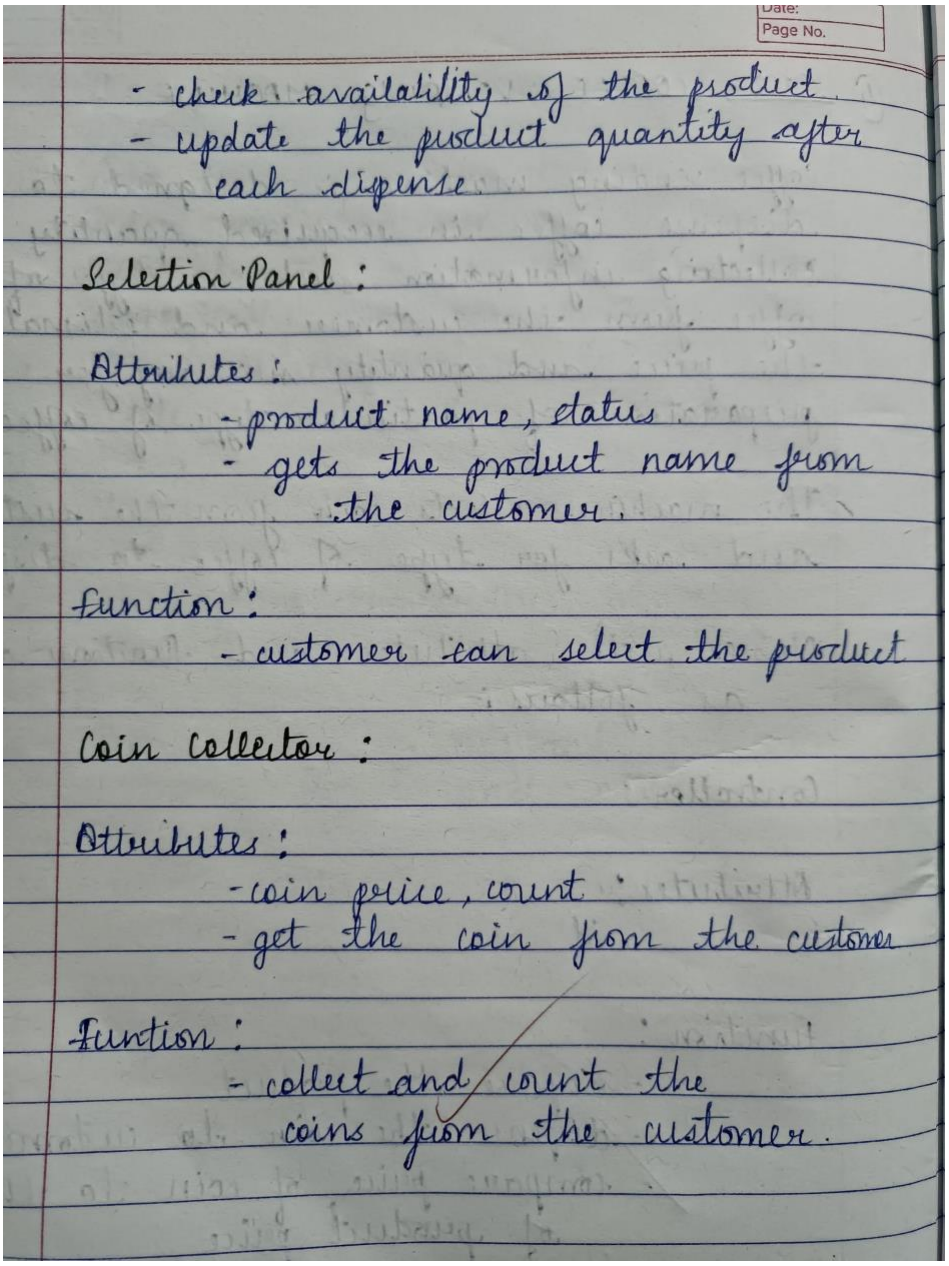
The machine collects coin from the customer and asks for type of coffee to dispense.

Classes with attributes and functions are as follows :-

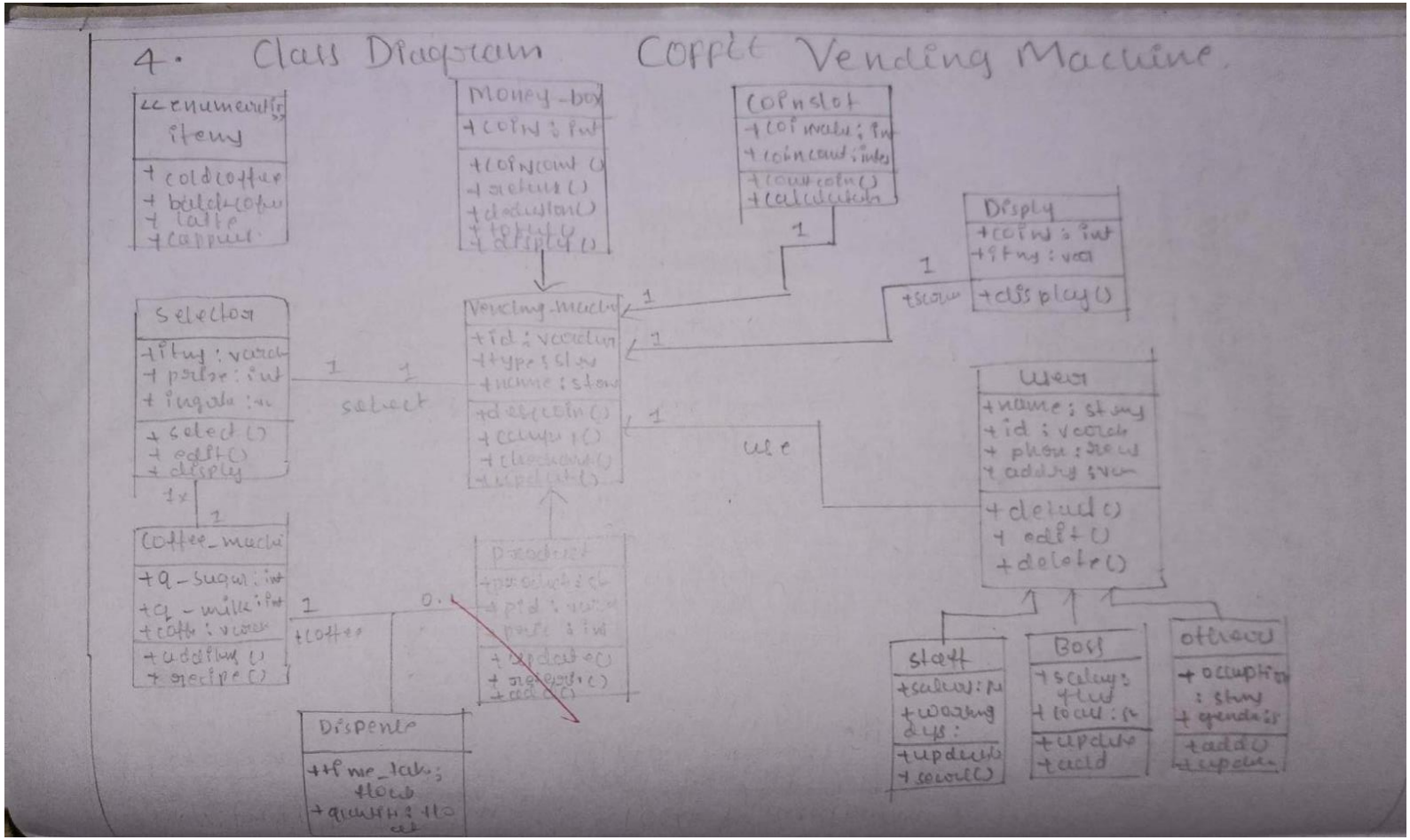
Controller :

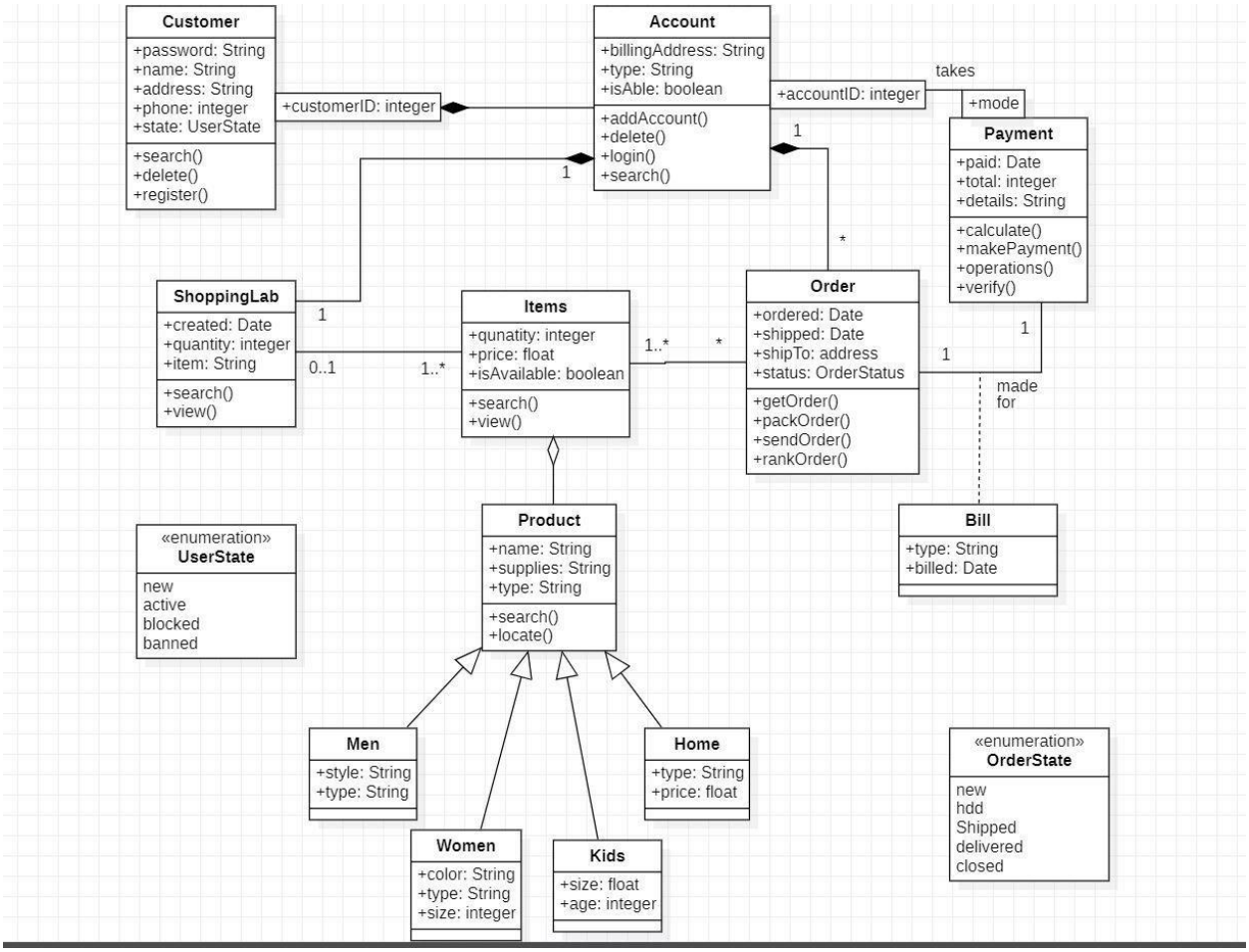
Attributes :
- id

Function :
- dispense the product
- dispense the coin to customer.
- compare price of coin to that of product price.



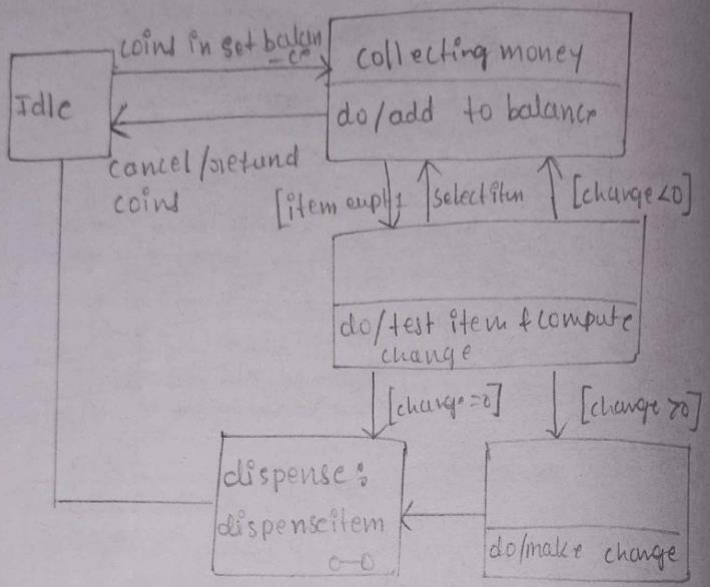
b) Advance Class Diagram:

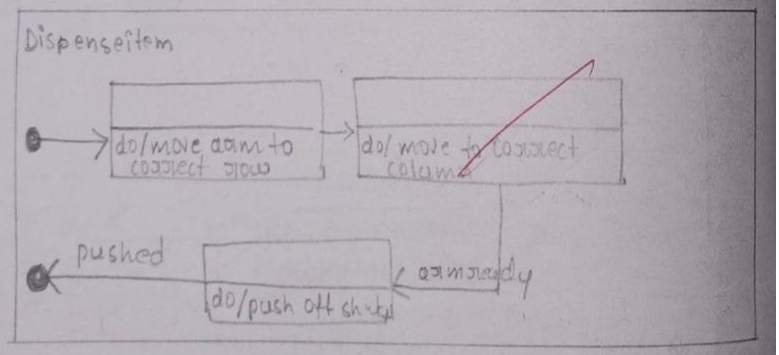

4. Class Diagram Coffee Vending Machine
Money_box
CoinSlot
Disply
Selector
Vending_machine
User
Coffee_machine
Product
Dispense
Staff
Boss
others
select
use
1
1..*
0..*

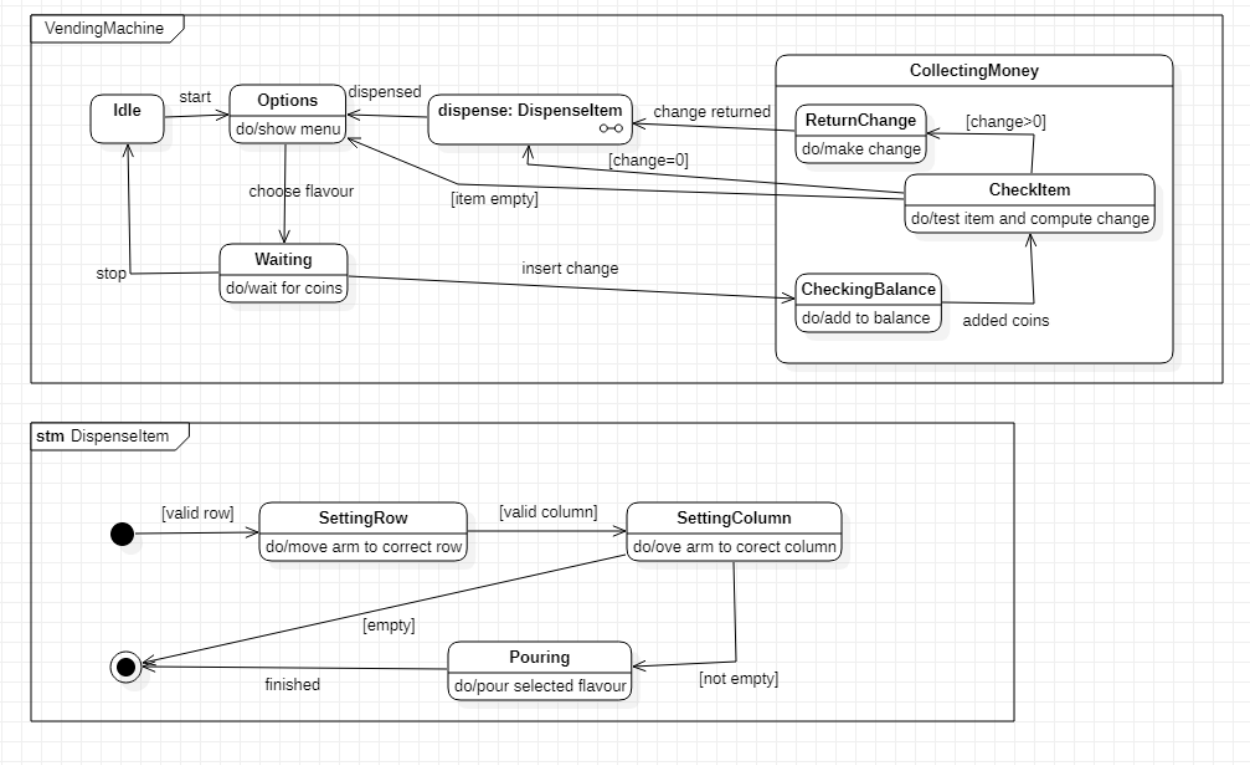


c) Advance State Diagram:

4) Coffee vending machine :

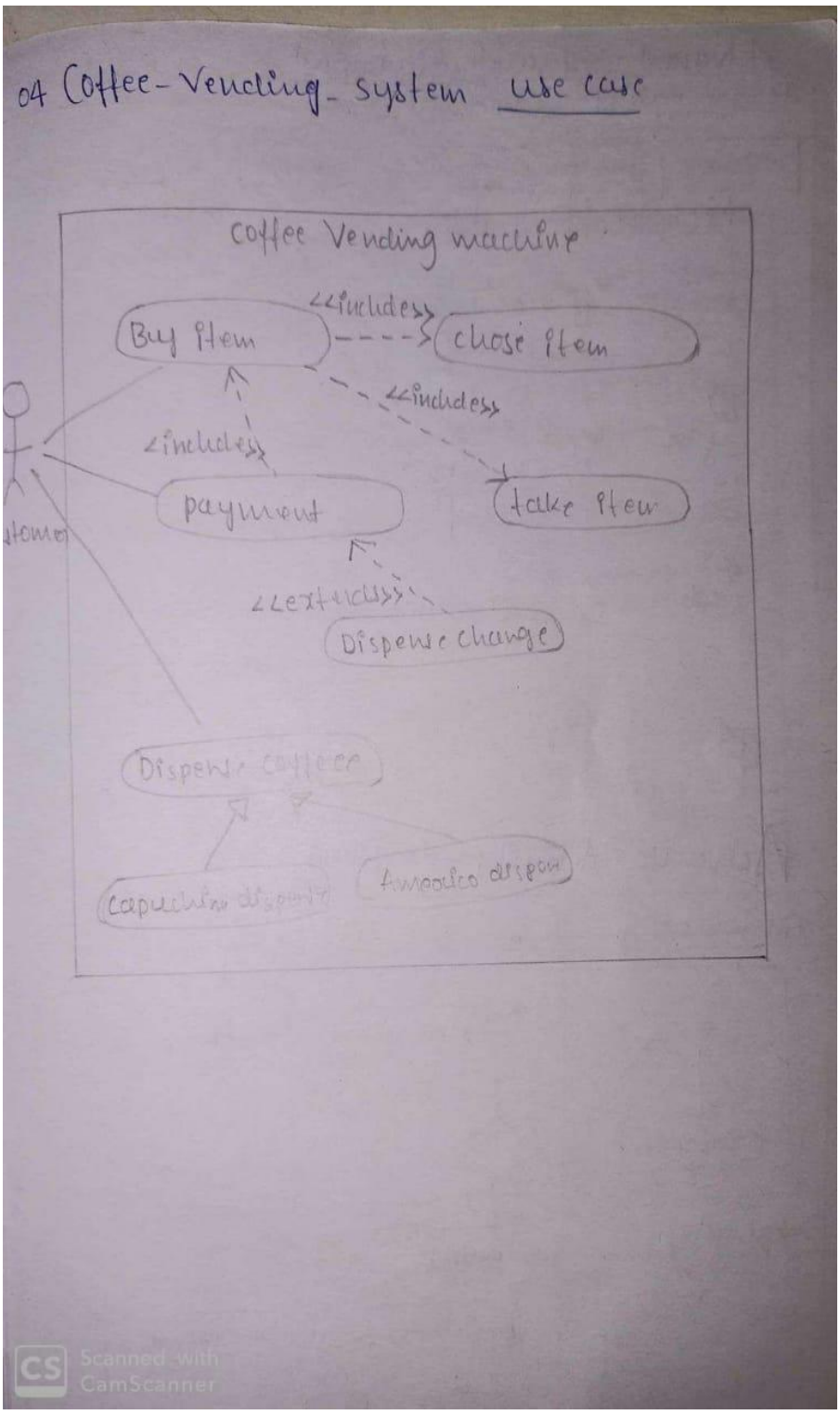


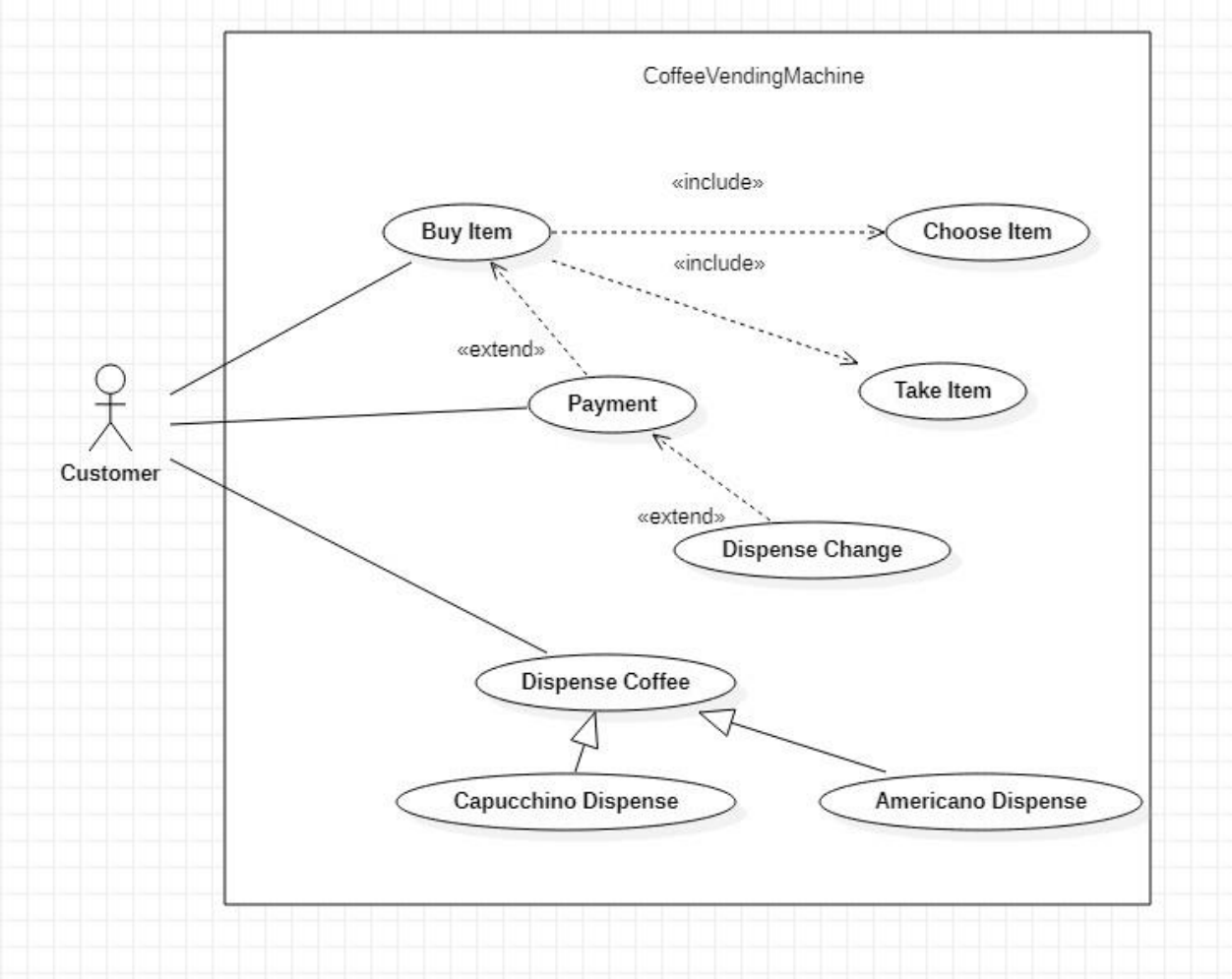




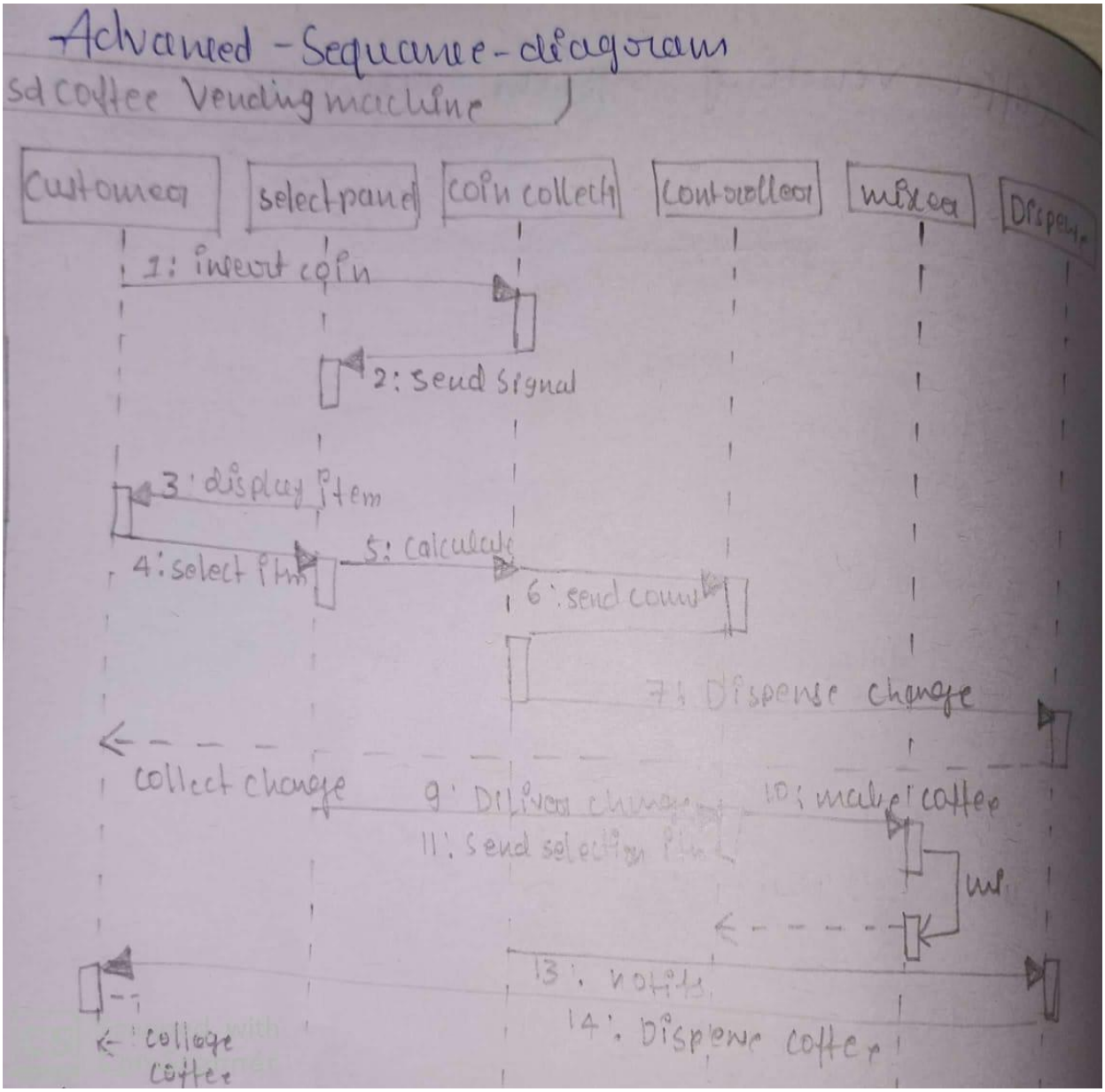
d) Advance Use Case Diagram:

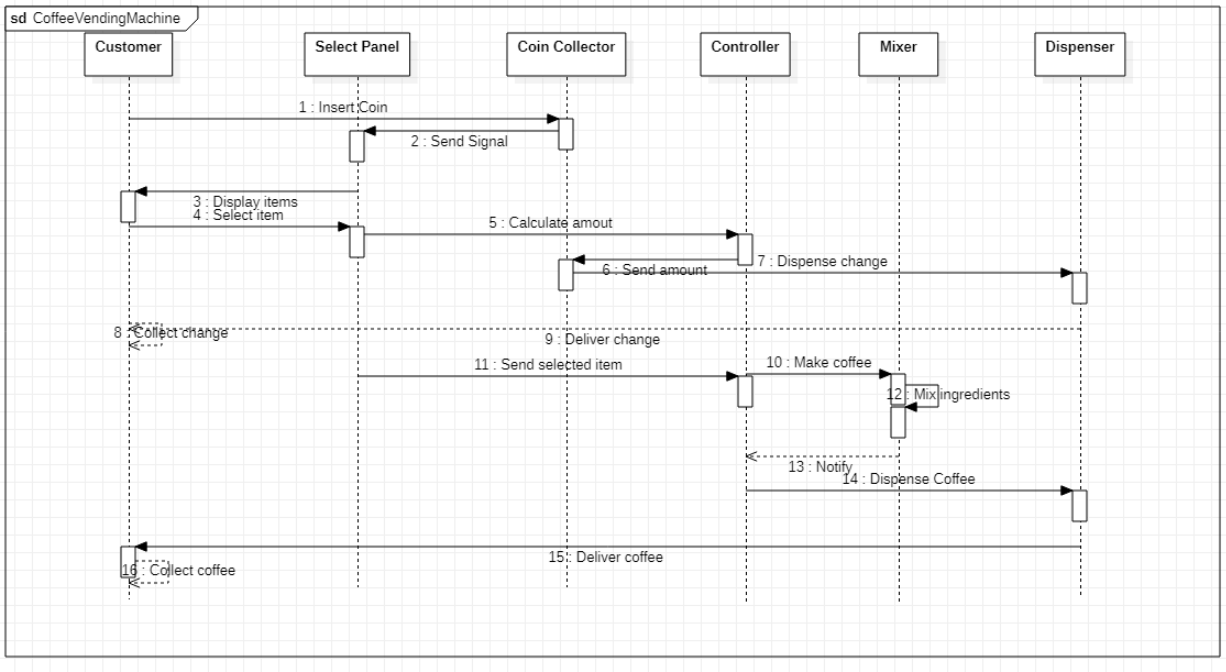
04 Coffee-Vending-system use case



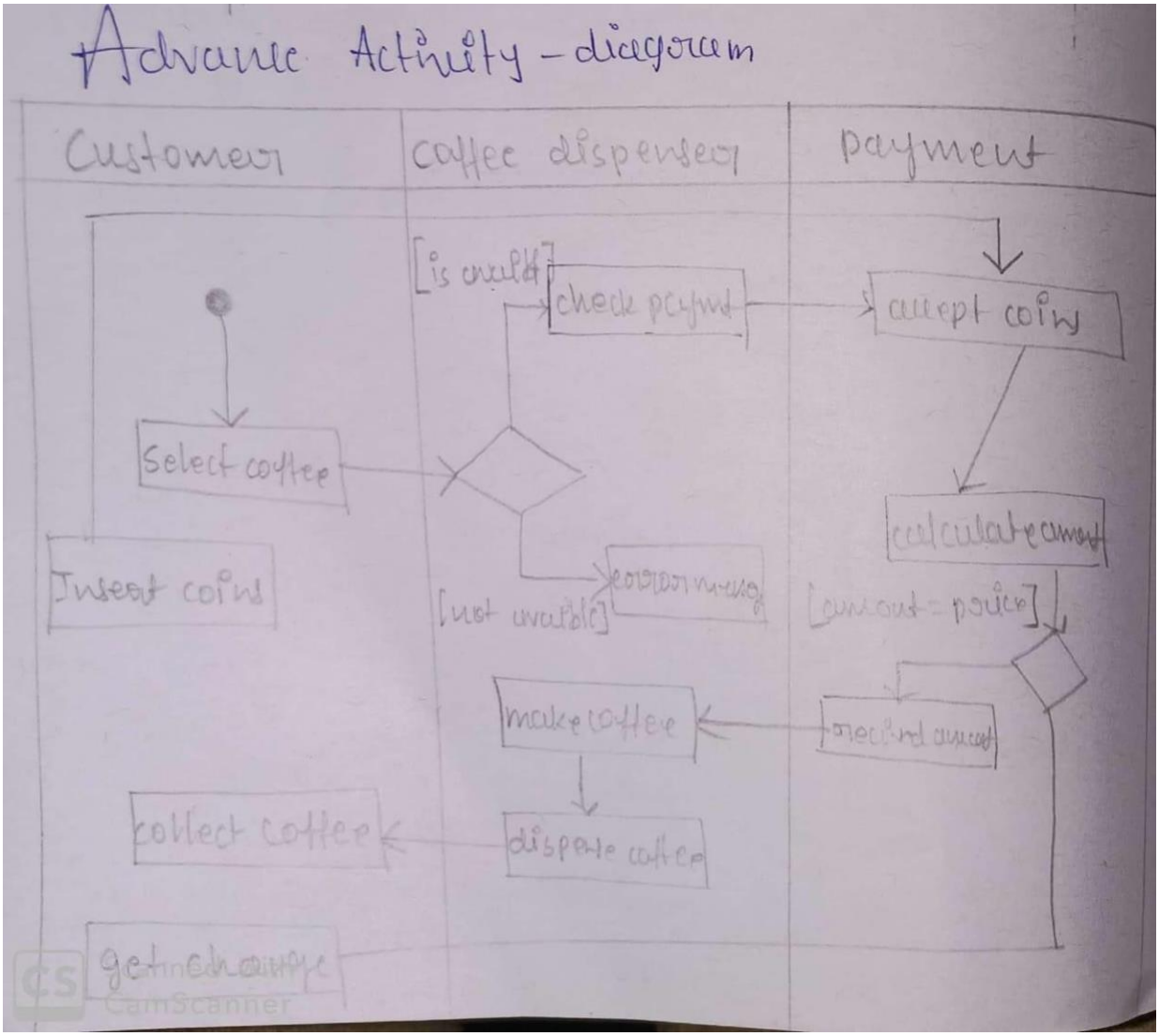


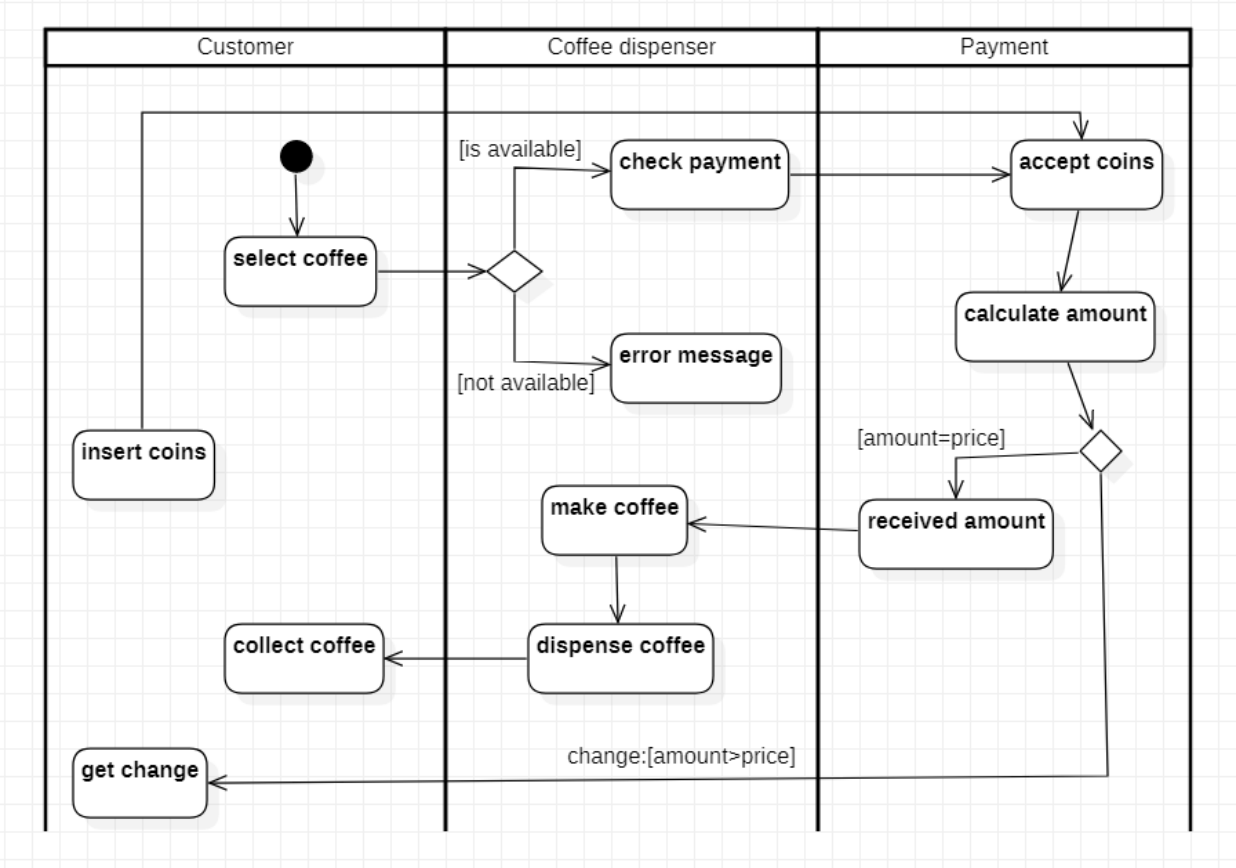
e) Sequence Diagram:


Advanced - Sequence - diagram
sd coffee Vending machine
Customer
selectpanel
coin collect
controller
mixer
Dispenser
1: insert coin
2: send signal
3: display item
4: select item
5: calculate
6: send coin
7: Dispense change
collect change
9: Dispense change
10: make coffee
11: send selection
wait
13: notify
14: Dispense coffee
collect coffee


sd CoffeeVendingMachine
Customer
Select Panel
Coin Collector
Controller
Mixer
Dispenser
1 : Insert Coin
2 : Send Signal
3 : Display items
4 : Select item
5 : Calculate amout
6 : Send amount
7 : Dispense change
8 : Collect change
9 : Deliver change
11 : Send selected item
10 : Make coffee
12 : Mix ingredients
13 : Notify
14 : Dispense Coffee
15 : Deliver coffee
16 : Collect coffee

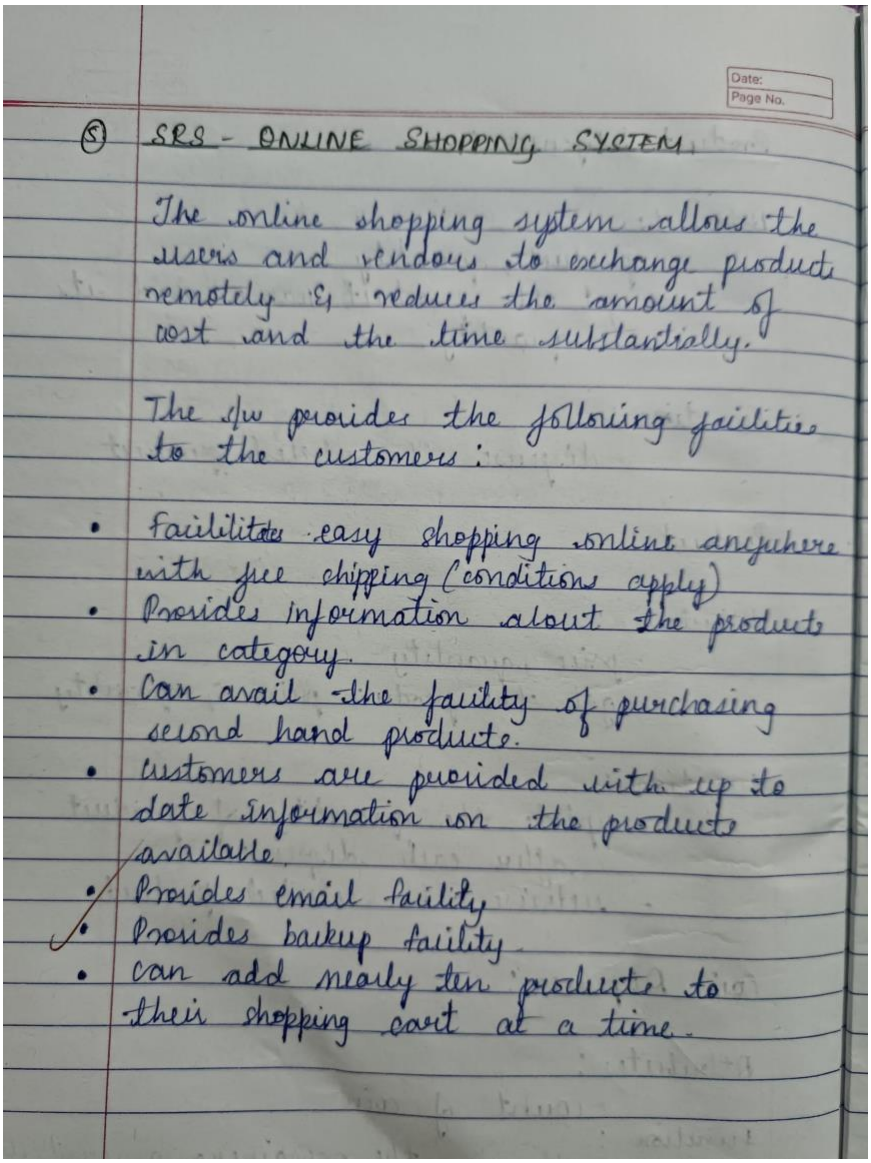
f) Activity Diagram:




Customer
Coffee dispenser
Payment
[is available]
check payment
accept coins
select coffee
calculate amount
error message
[not available]
insert coins
[amount=price]
make coffee
received amount
collect coffee
dispense coffee
change:[amount>price]
get change

5. Online Shopping System-

a) SRS:



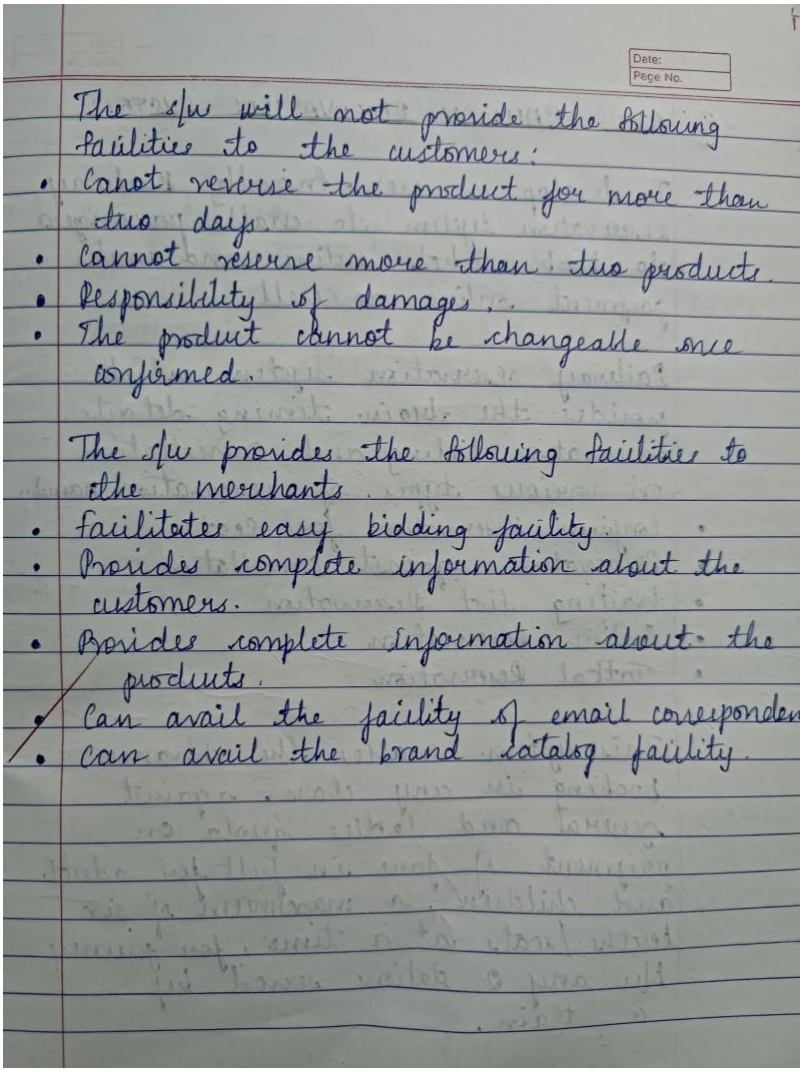
Date:
Page No.

(5) SRS - ONLINE SHOPPING SYSTEM

The online shopping system allows the users and vendors to exchange products remotely & reduces the amount of cost and the time substantially.

The sw provides the following facilities to the customers:

- Facilitates easy shopping online anywhere with free shipping (conditions apply)
- Provides information about the products in category.
- Can avail the facility of purchasing second hand products.
- Customers are provided with up to date information on the products available.
- Provides email facility.
- Provides backup facility.
- Can add nearly ten products to their shopping cart at a time.



Date:
Page No.

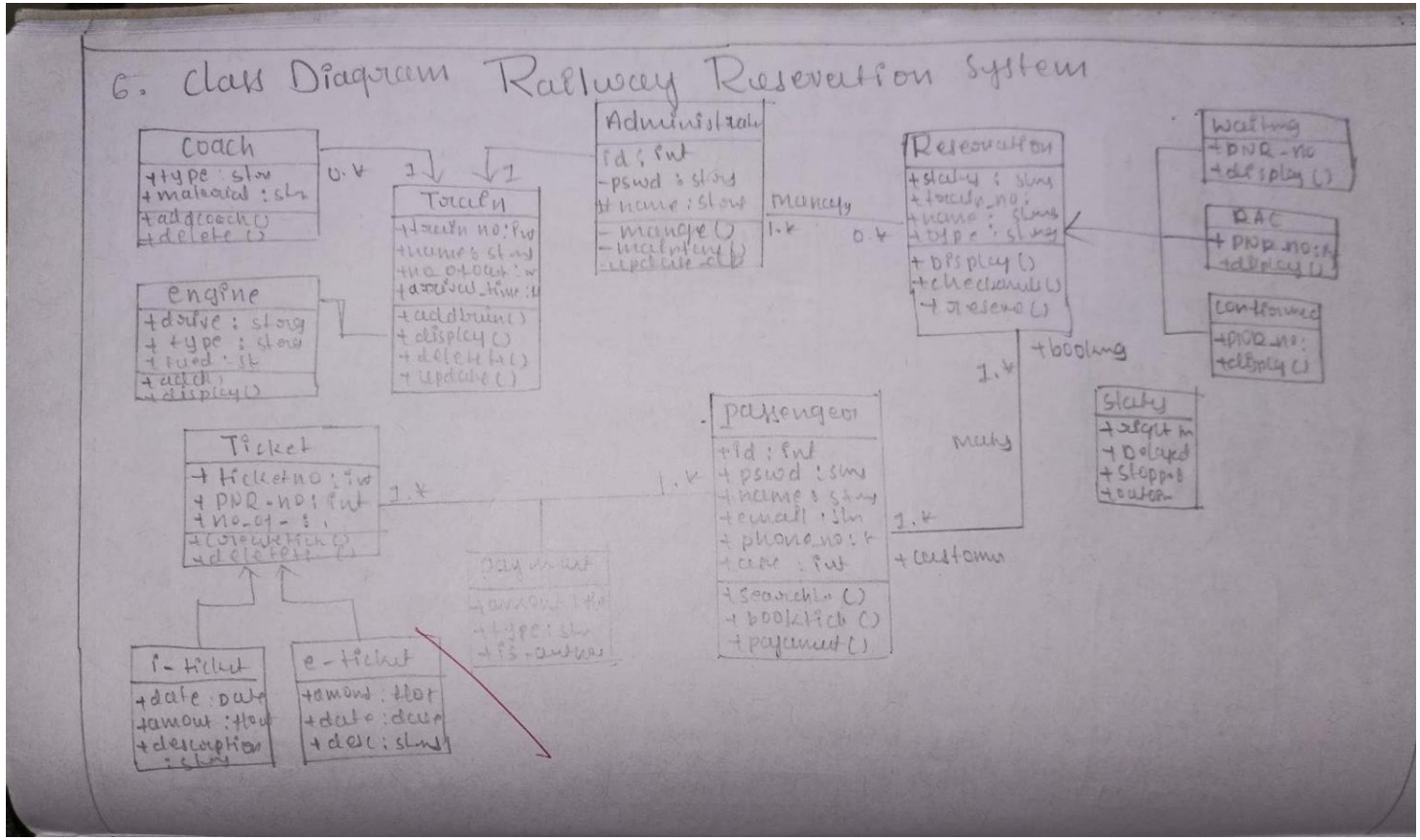
The s/w will not provide the following facilities to the customers:

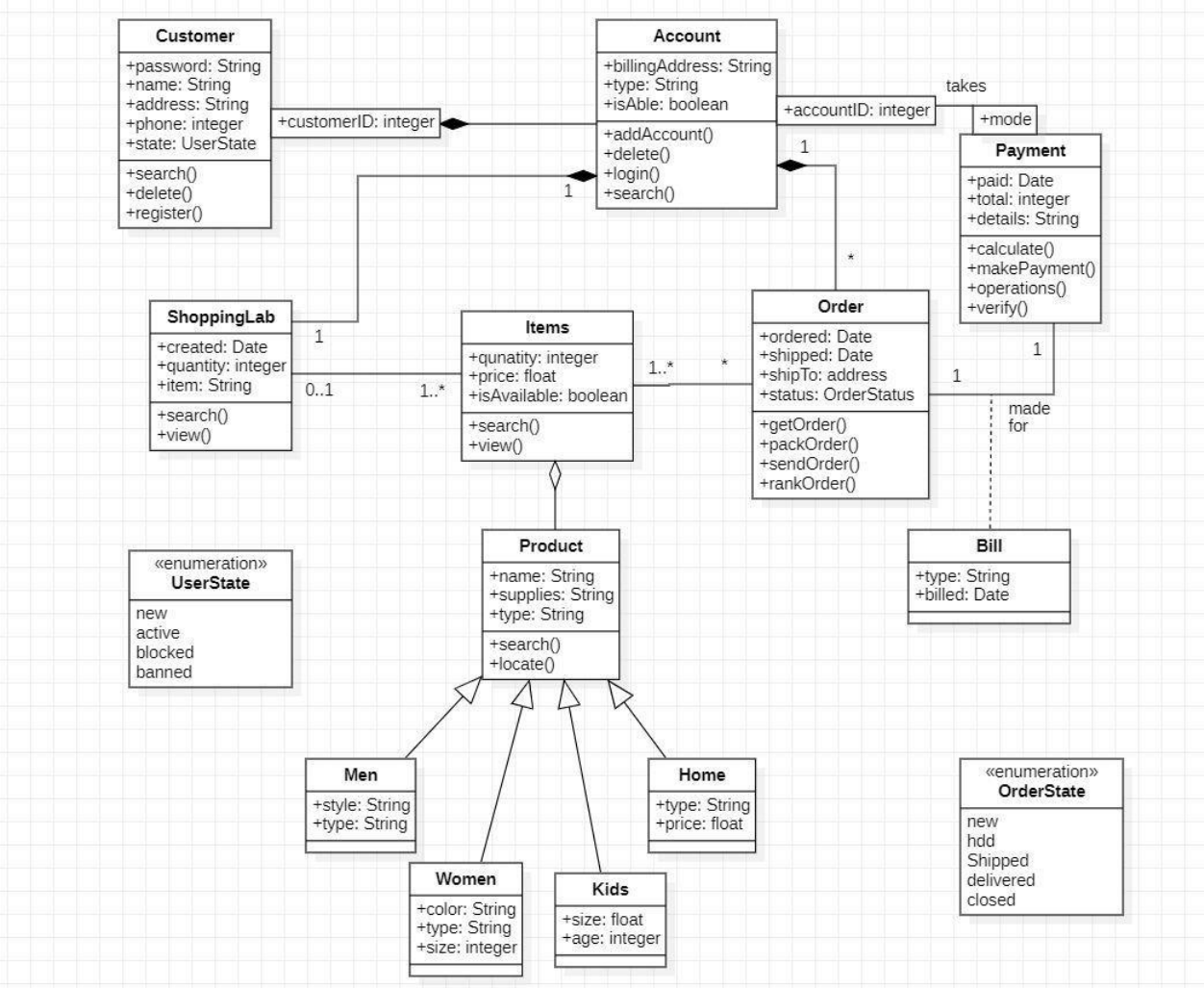
- Cannot reverse the product for more than two days.
- Cannot reserve more than two products.
- Responsibility of damages.
- The product cannot be changeable once confirmed.

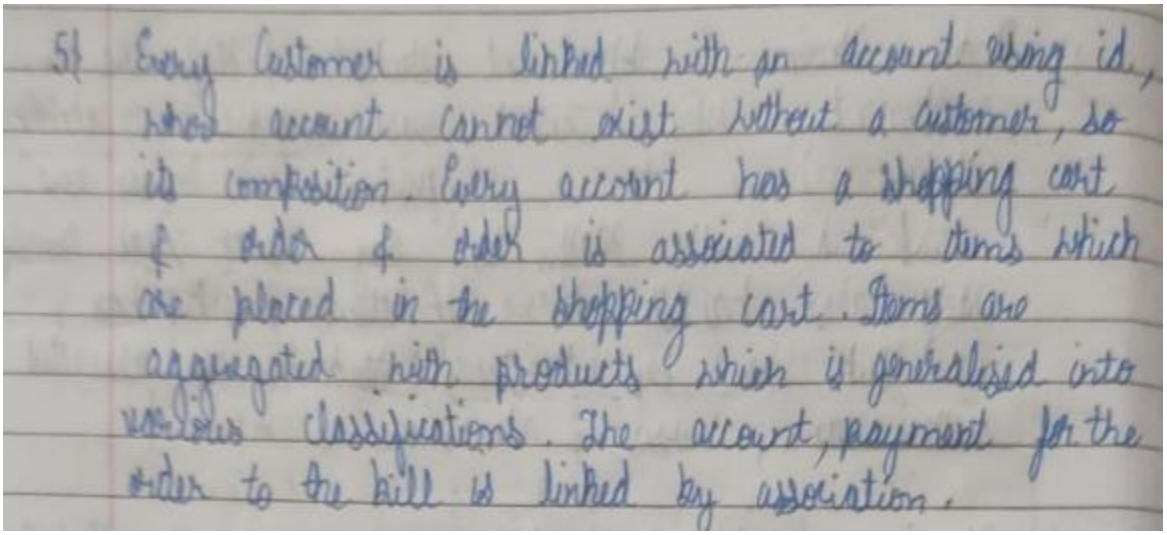
The s/w provides the following facilities to the merchants.

- Facilitates easy bidding facility.
- Provides complete information about the customers.
- Provides complete information about the products.
- Can avail the facility of email correspondence
- Can avail the brand catalog facility.

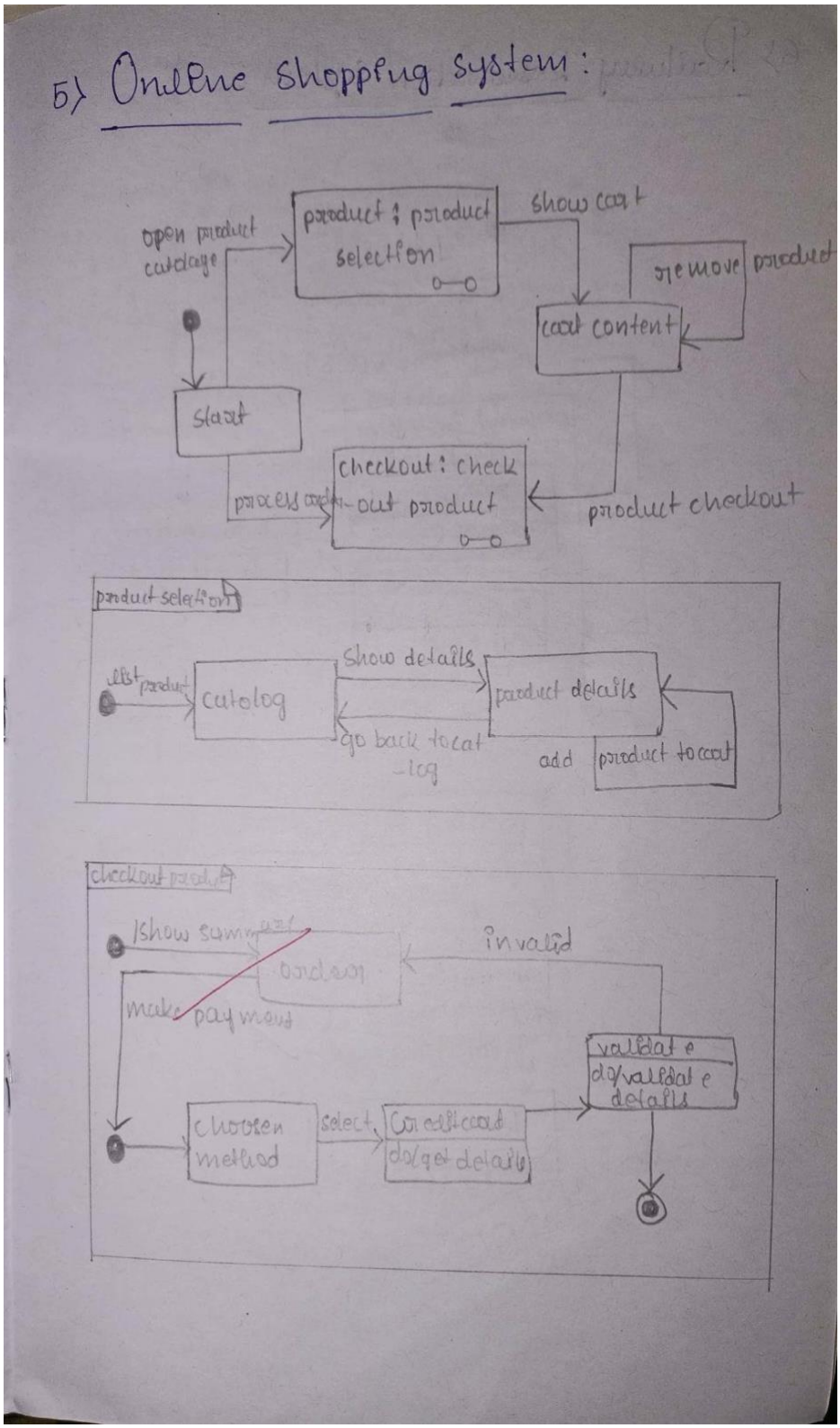
b) Advance Class Diagram:

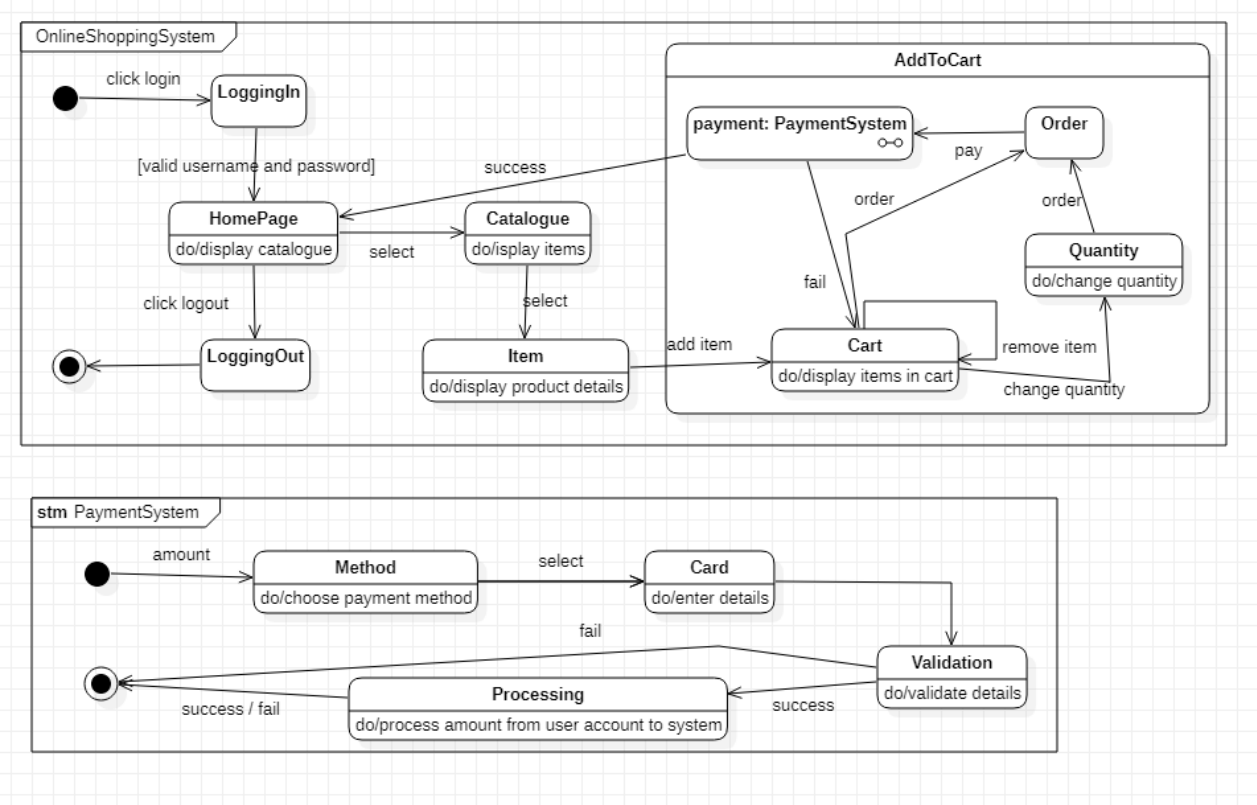

6. Class Diagram Railway Reservation System
Coach
+type : str
+material : str
+addcoach()
+delete()
0..*
1
1
Train
+train no : int
+name : str
+no of coach : int
+arrival_time : t
+addtrain()
+display()
+delete tr()
+update()
engine
+drive : str
+type : str
+used : it
+attach()
+display()
Administrator
id : int
-pswd : str
+name : str
- manage()
- maintain()
- inspect()
manage
1..*
0..*
Reservation
+status : str
+train_no :
+name : str
+type : str
+Display()
+checkavail()
+reserve()
+boolang
1..*
manage
1..*
+customs
Waiting
+PNR_no
+display()
RAC
+PNR_no:
+display()
confirmed
+PNR_no:
+display()
passengers
+id : int
+pswd : str
+name : str
+email : str
+phone_no : i
+age : int
+searchin()
+bookticket()
+payment()
Ticket
+ticketno : int
+PNR_no : int
+no_of_s :
+storeticket()
+deleteit()
1..*
1..*
Payment
+amount : flo
+type : str
+is_auth
i-ticket
+date : Date
+amount : float
+description : str
e-ticket
+amount : float
+date : date
+desc : str
Status
+right n
+Delayed
+Stopped
+outer



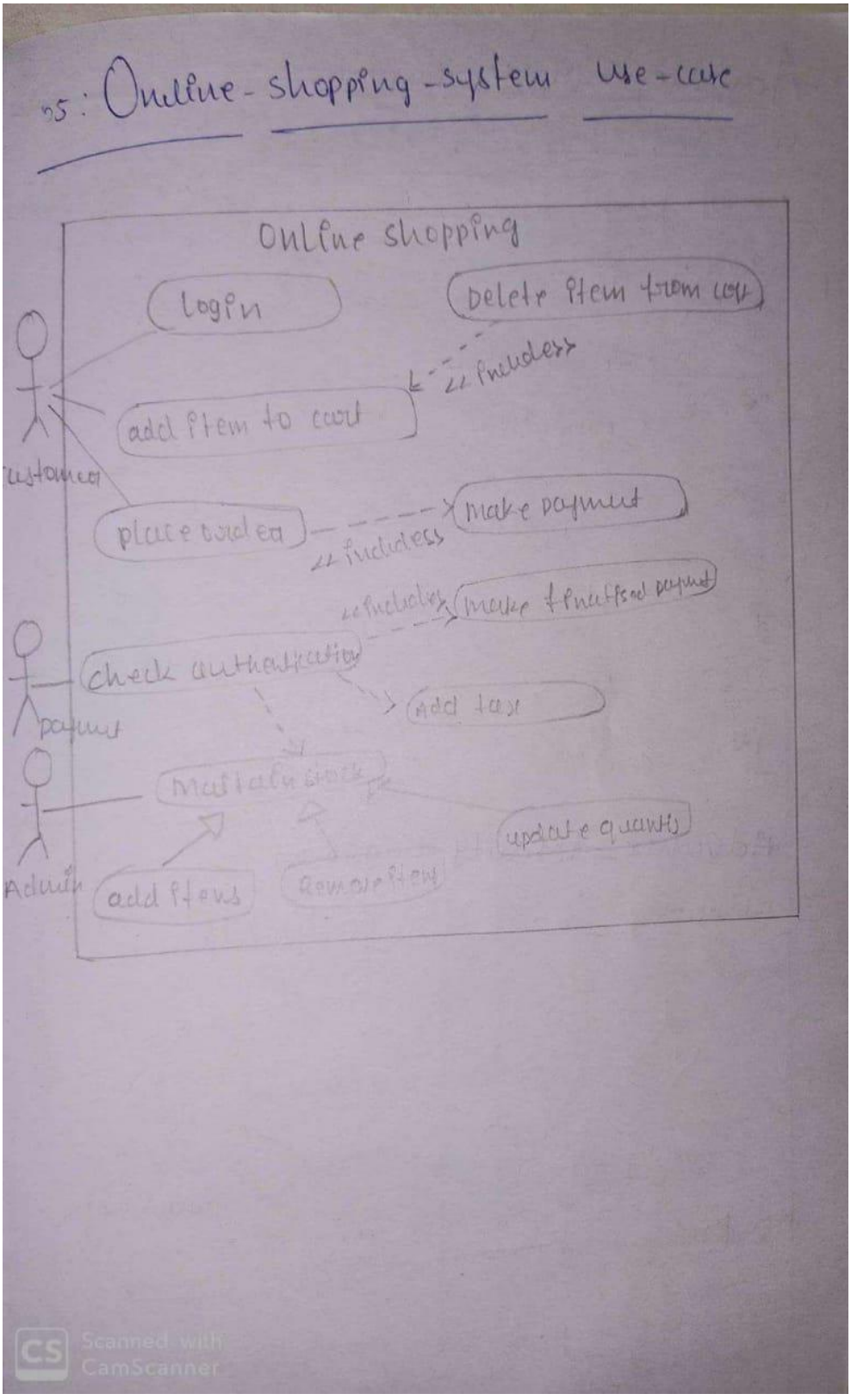

5) Every Customer is linked with an account using id, whose account cannot exist without a customer, so its composition. Every account has a shopping cart & order & order is associated to items which are placed in the shopping cart. Items are aggregated with products which is generalised into various classifications. The account, payment for the order to the bill is linked by association.

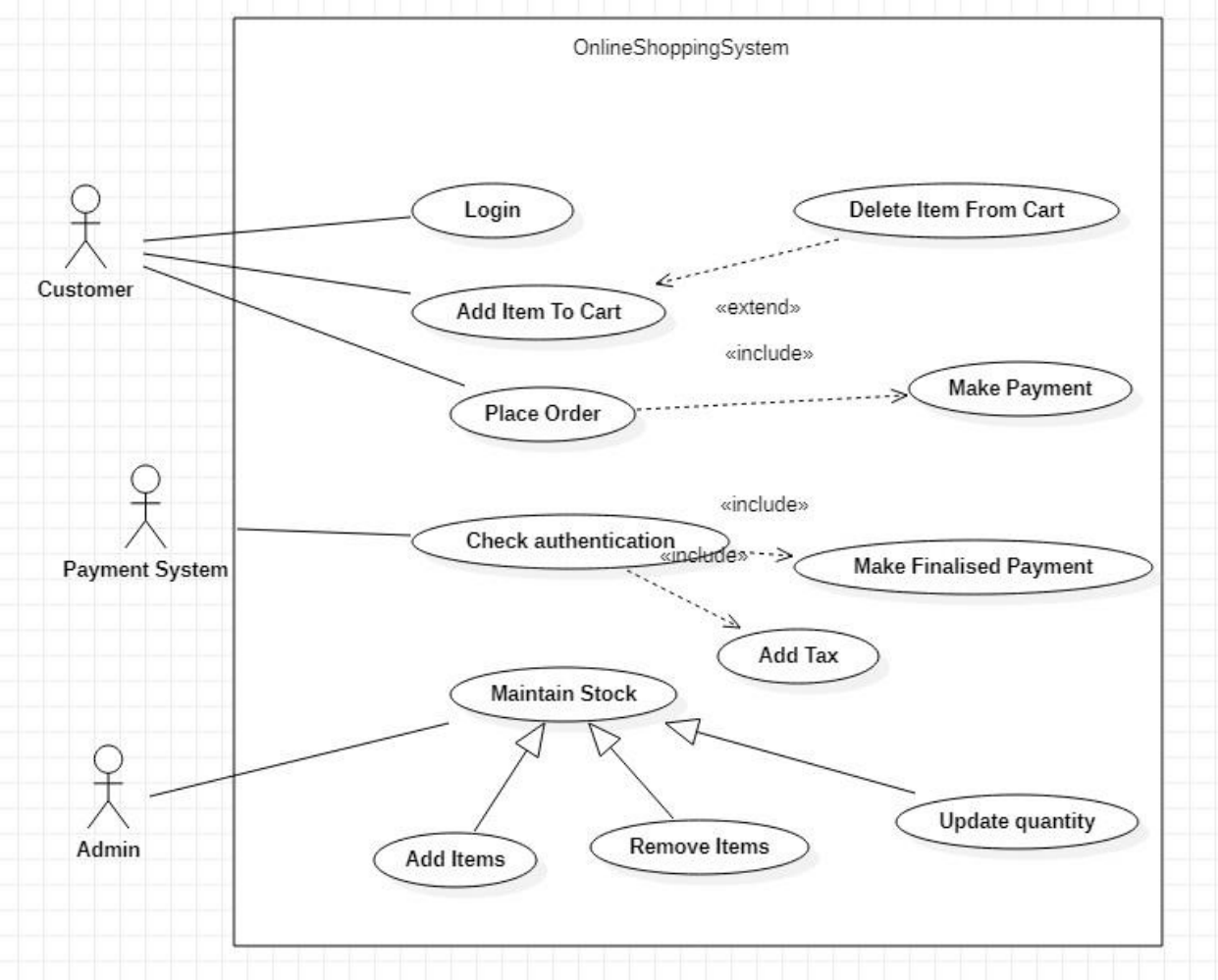
c) Advance State Diagram:



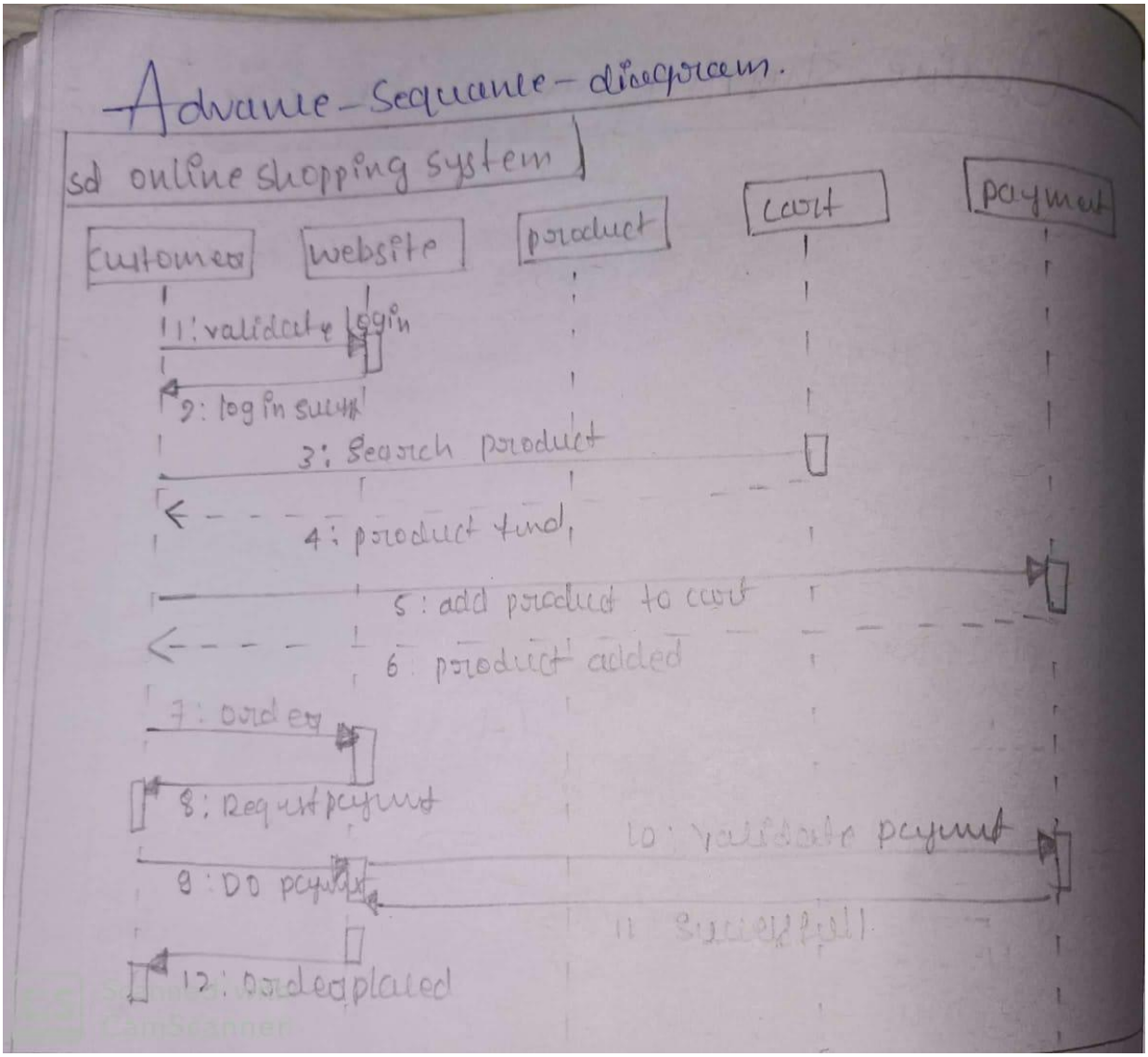


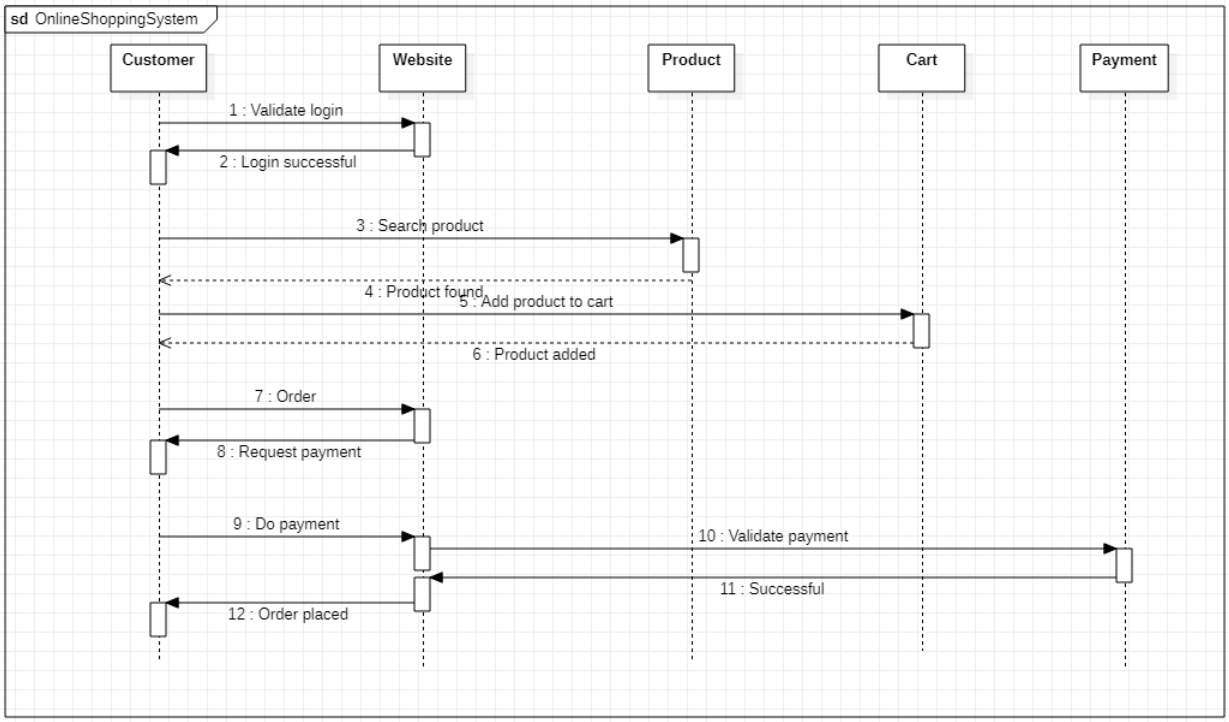
d) Advance Use Case Diagram:


05 : Online - shopping - system Use - case
Online shopping
login
delete item from cart
<<includes>>
add item to cart
customer
place order
make payment
<<includes>>
<<includes>>
make financial payment
check authentication
add tax
payment
maintain stock
update quantity
add items
remove item
Admin
Scanned with CamScanner

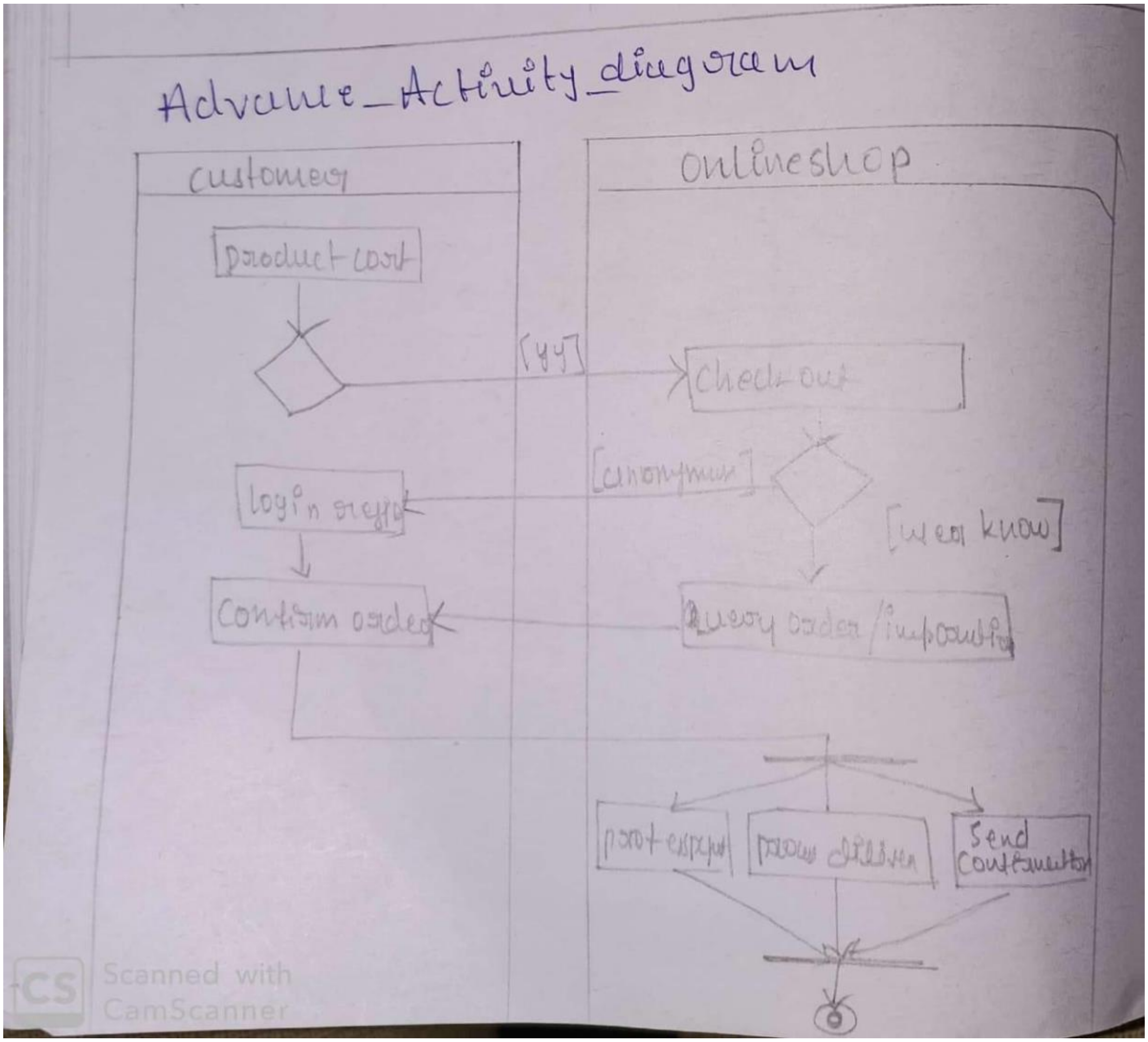


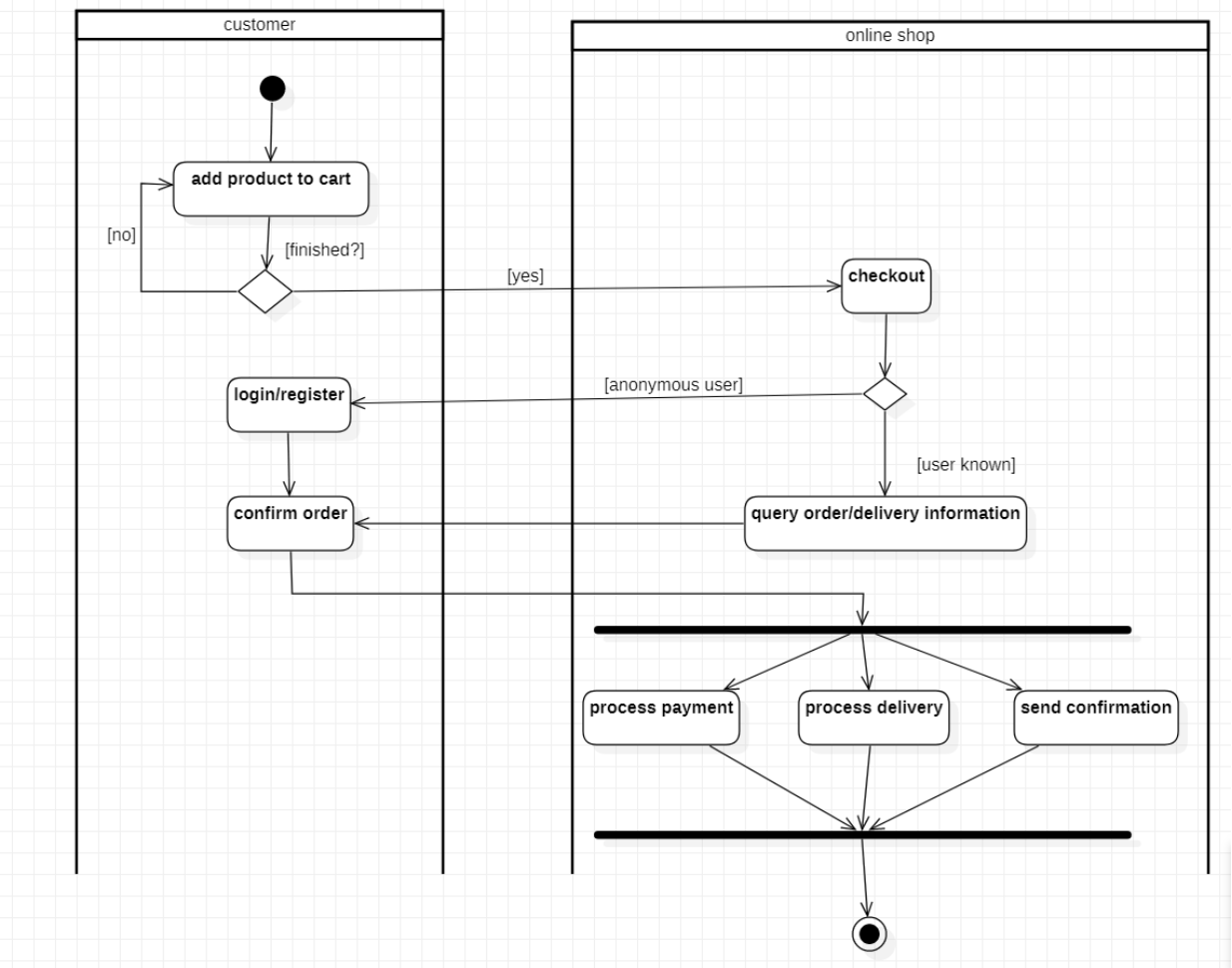
e) Sequence Diagram:


Advance - Sequence - diagram.
sd online shopping system
customer
website
product
cart
payment
1: validate login
2: login succ
3: Search product
4: product find
5: add product to cart
6: product added
7: order
8: Request payment
9: DO payment
10: validate payment
11: Successful
12: Order placed



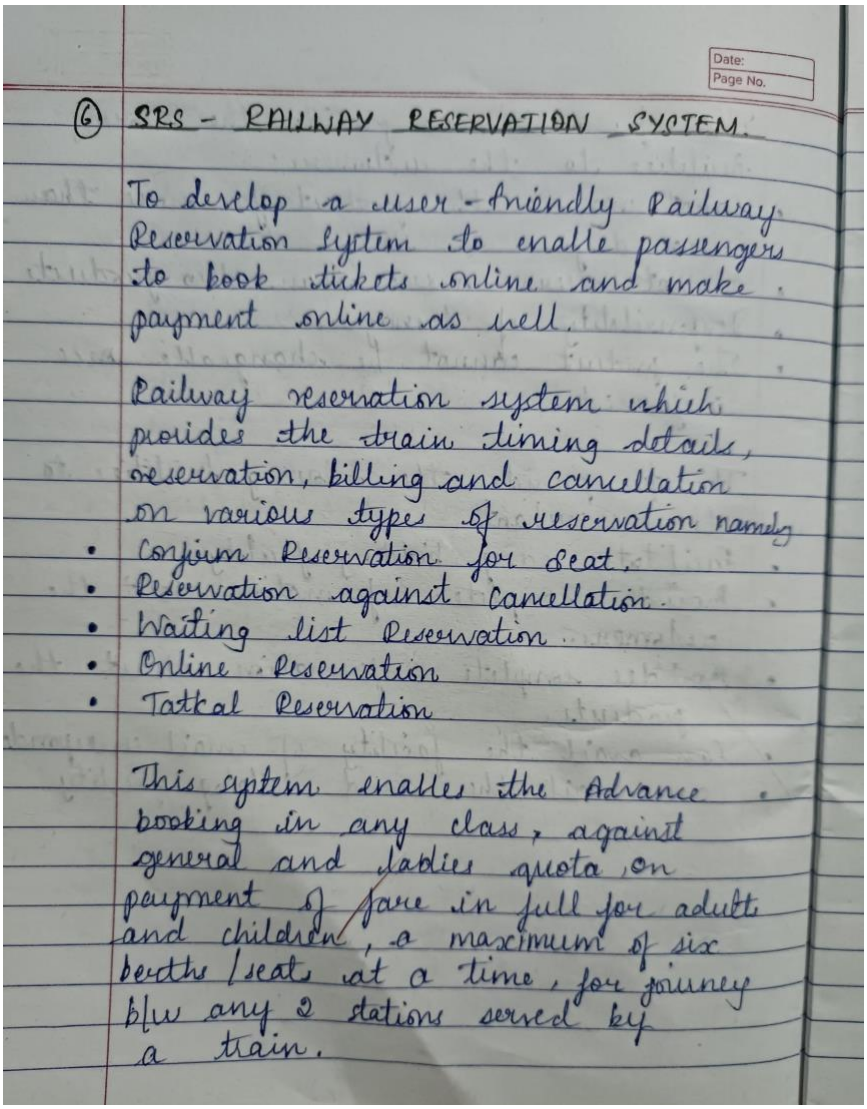
f) Activity Diagram:


Advance_Activity_diagram
customer
onlineshop
product cart
Check out
Confirm order
Send


customer
online shop
add product to cart
[no]
[finished?]
[yes]
checkout
login/register
[anonymous user]
[user known]
confirm order
query order/delivery information
process payment
process delivery
send confirmation

6. Railway reservation system-

a) SRS:



Date:
Page No.

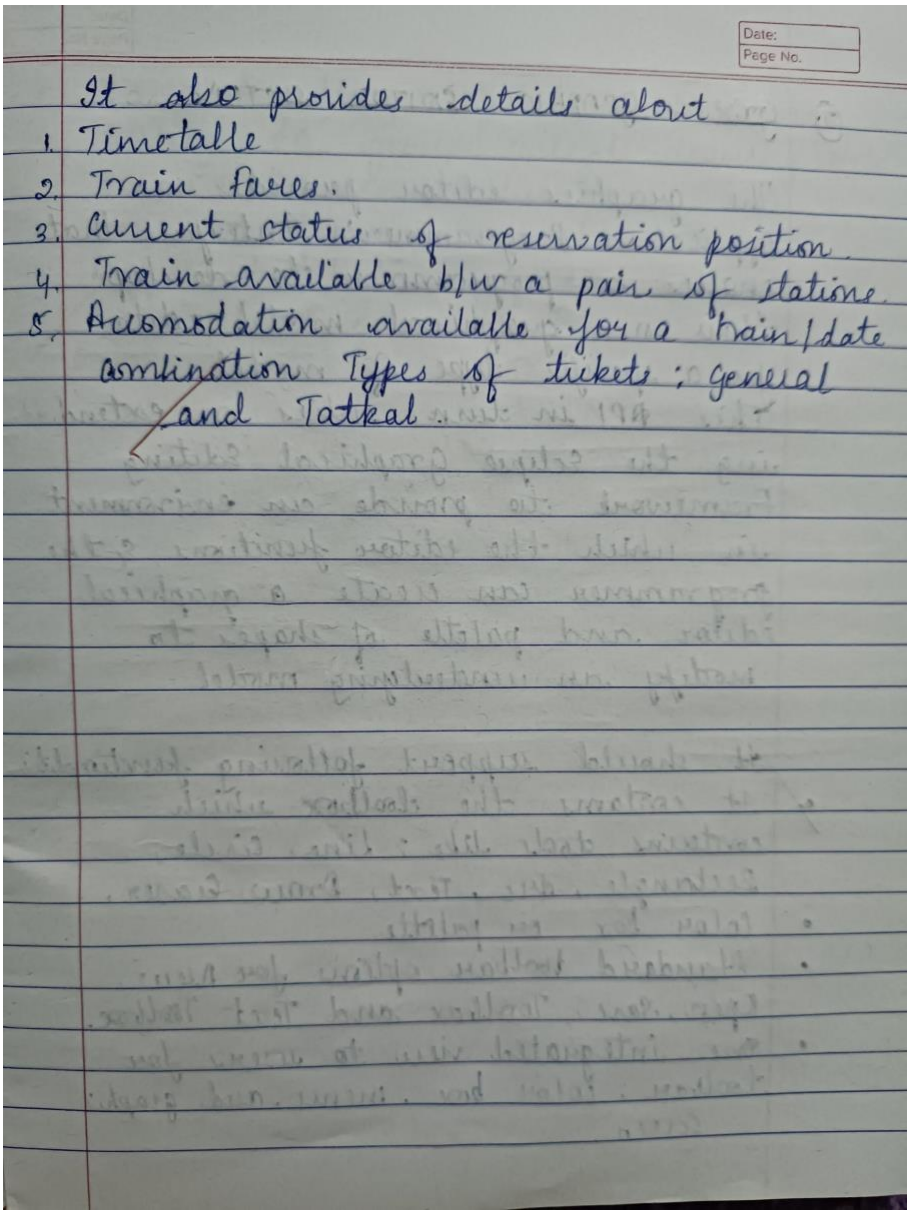
⑥ SRS - RAILWAY RESERVATION SYSTEM.

To develop a user-friendly Railway Reservation System to enable passengers to book tickets online and make payment online as well.

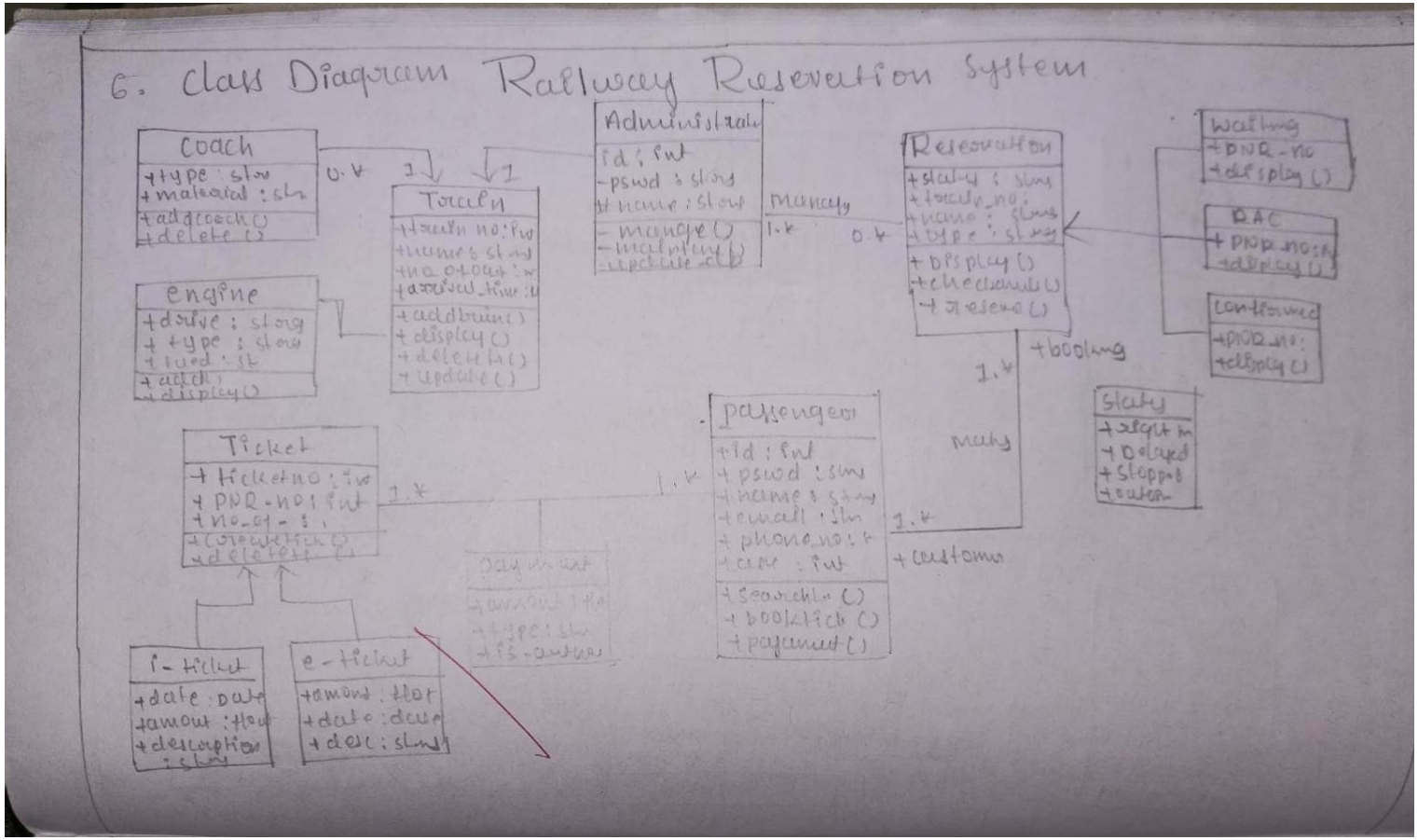
Railway reservation system which provides the train timing details, reservation, billing and cancellation on various types of reservation namely

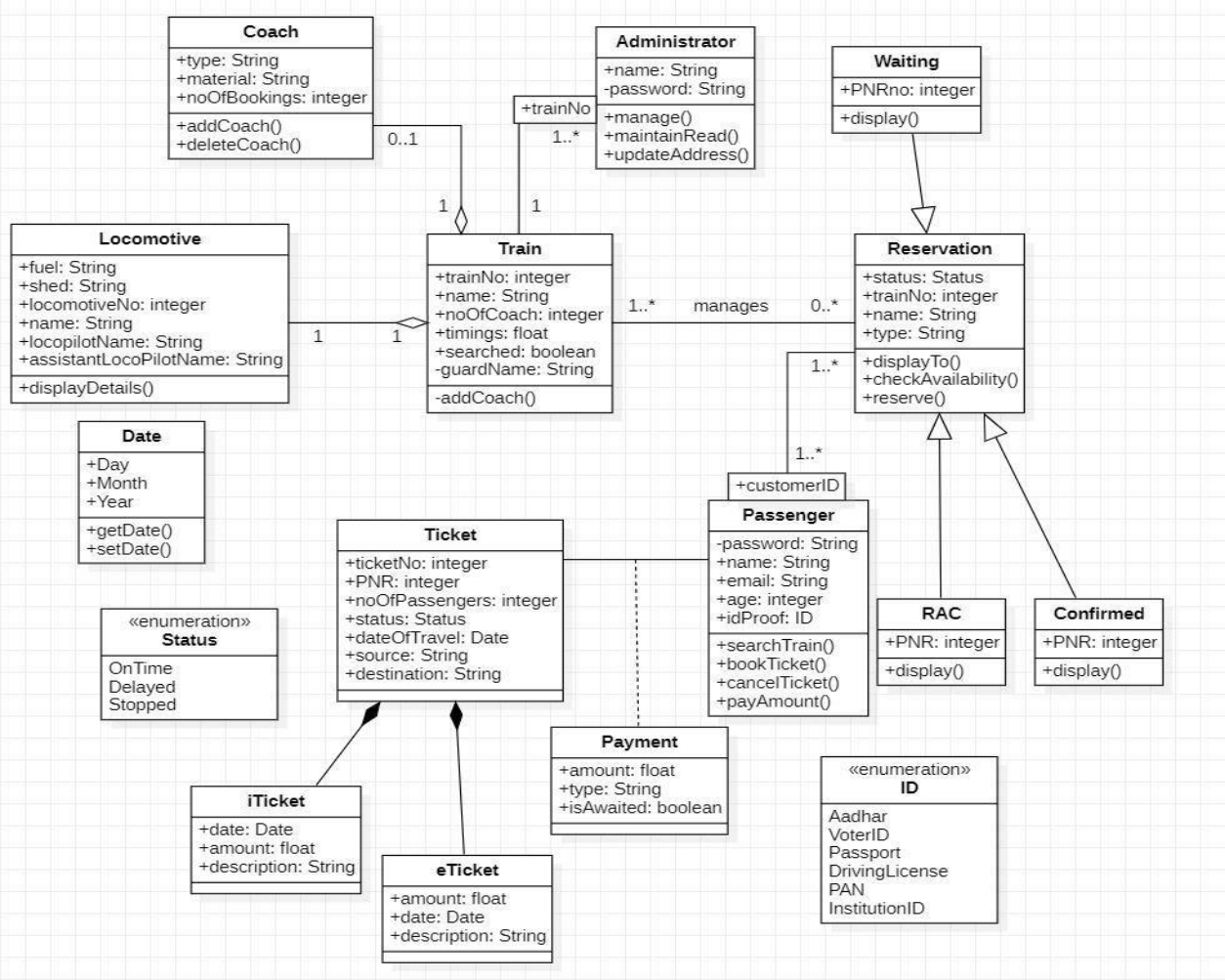
- Confirm Reservation for seat.
- Reservation against cancellation.
- Waiting list Reservation.
- Online Reservation
- Tatkal Reservation

This system enables the Advance booking in any class, against general and ladies quota on payment of fare in full for adults and children, a maximum of six berths/seats at a time, for journey b/w any 2 stations served by a train.



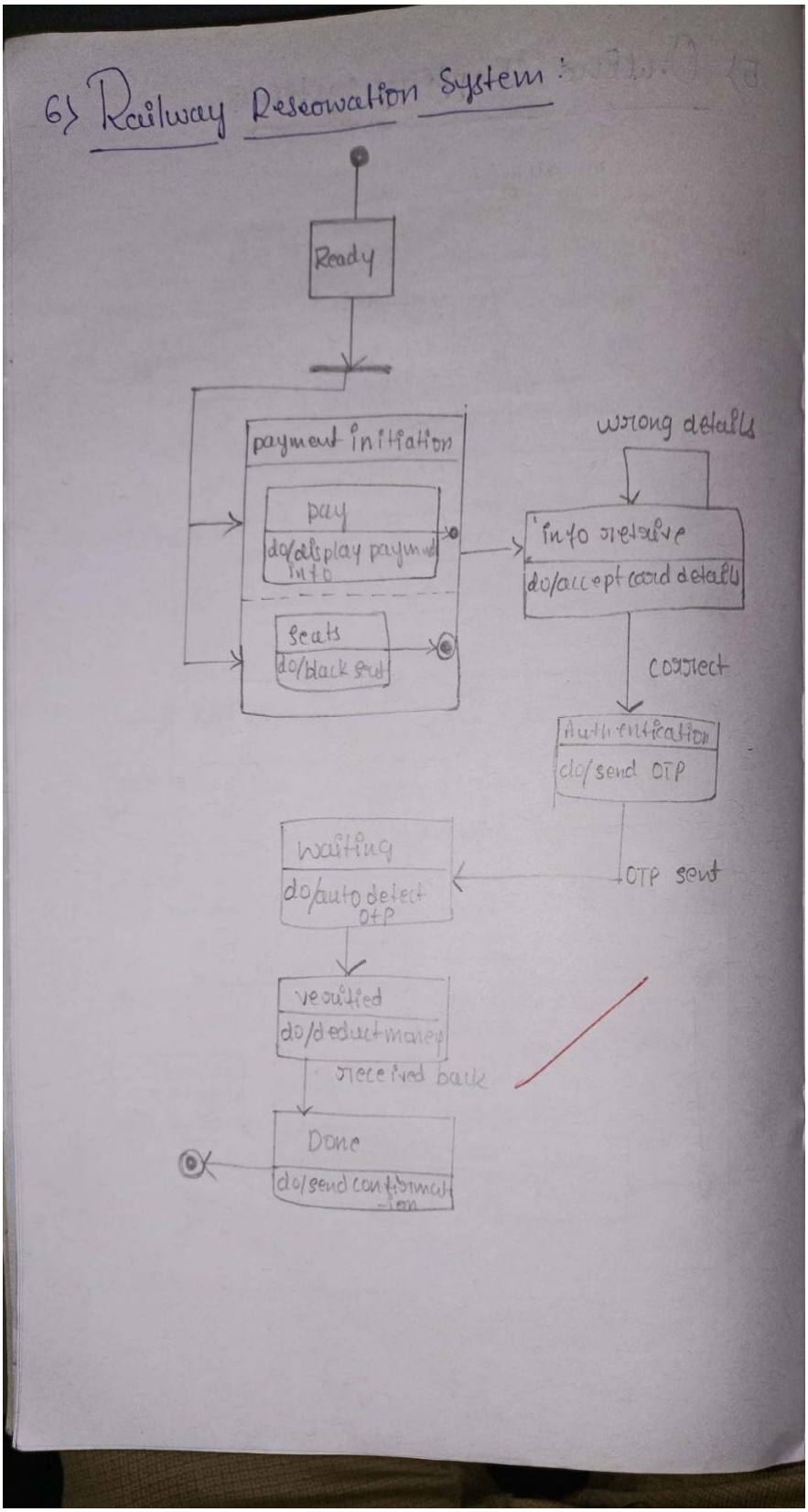
b) Advance Class Diagram:

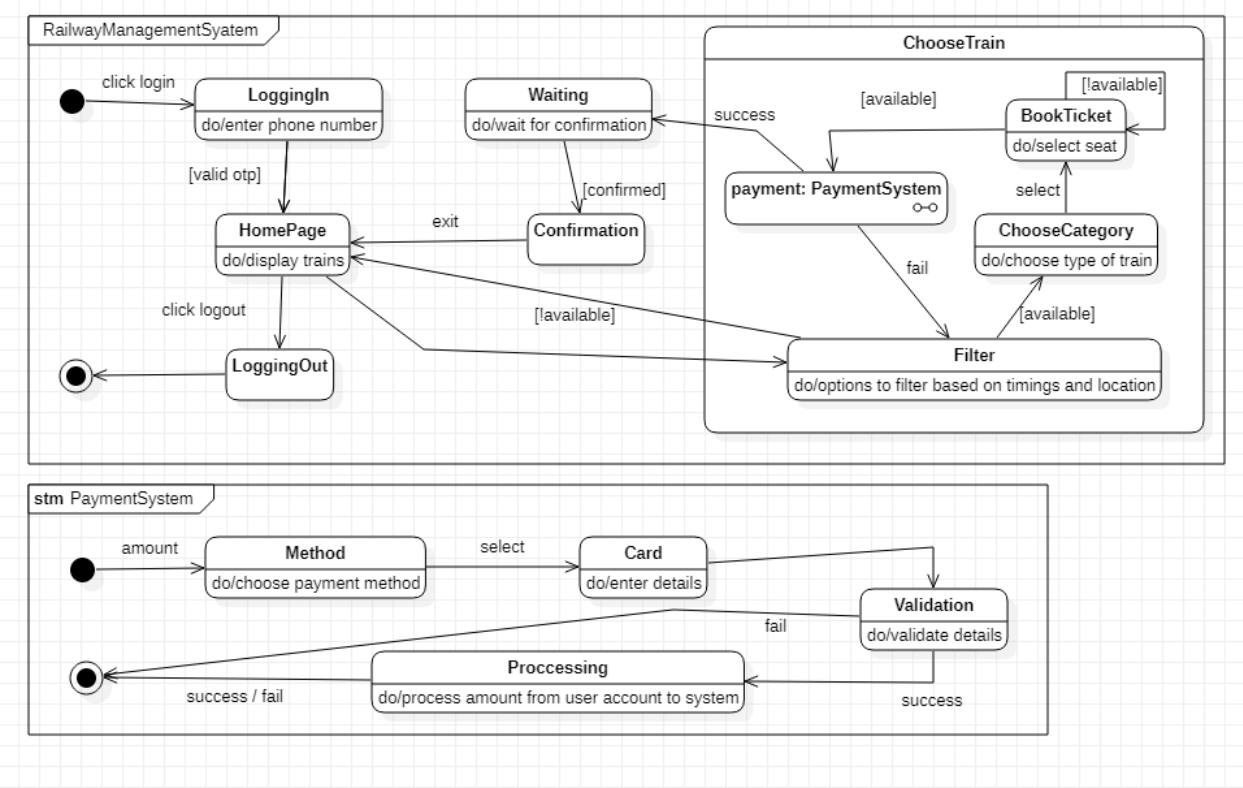

6. Class Diagram Railway Reservation System
Coach
+type : str
+material : str
+addcoach()
+delete()
0..*
1
1
Train
+train no : int
+name : str
+no of car : int
+arrival_time : t
+addtrain()
+display()
+deletetr()
+update()
Administrator
id : int
-pswd : str
+name : str
- manage()
- maintain()
- inspect_car()
manage
1..*
0..*
Reservation
+status : str
+train_no :
+name : str
+type : str
+ display()
+checkavail()
+reserve()
+booking
1..*
many
+customer
Waiting
+PNR_no
+display()
RAC
+PNR_no :
+display()
confirmed
+PNR_no :
+display()
Engine
+drive : str
+type : str
+fuel : str
+attach()
+display()
Ticket
+ticketNO : int
+PNR_no : int
+no_of_ :
+cancelticket()
+deleteti()
1..*
1..*
Payment
+amount : flo
+type : str
+is_amount
Passenger
+id : int
+pswd : str
+name : str
+email : str
+phone_no : i
+age : int
+searchtr()
+bookticket()
+payment()
Status
+right t
+Delayed
+Stopped
+cancel
i-ticket
+date : dat
+amount : flo
+description
: str
e-ticket
+amount : flo
+date : dou
+desc : str



c) Advance State Diagram:

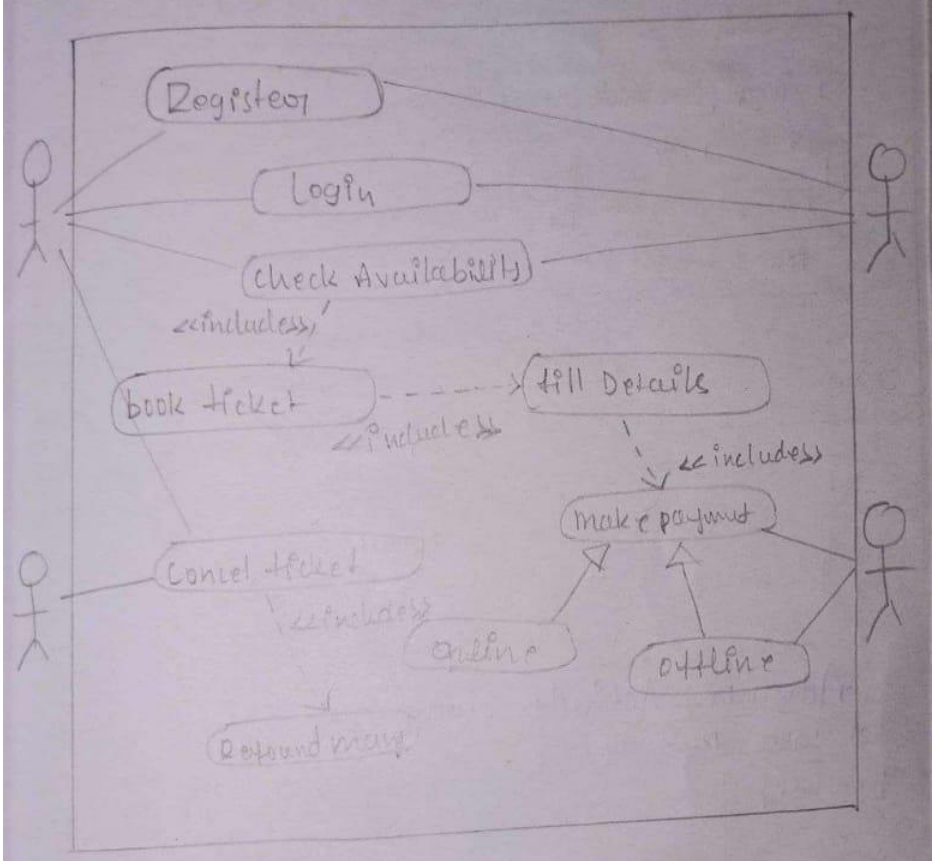
6) Railway Reservation System :

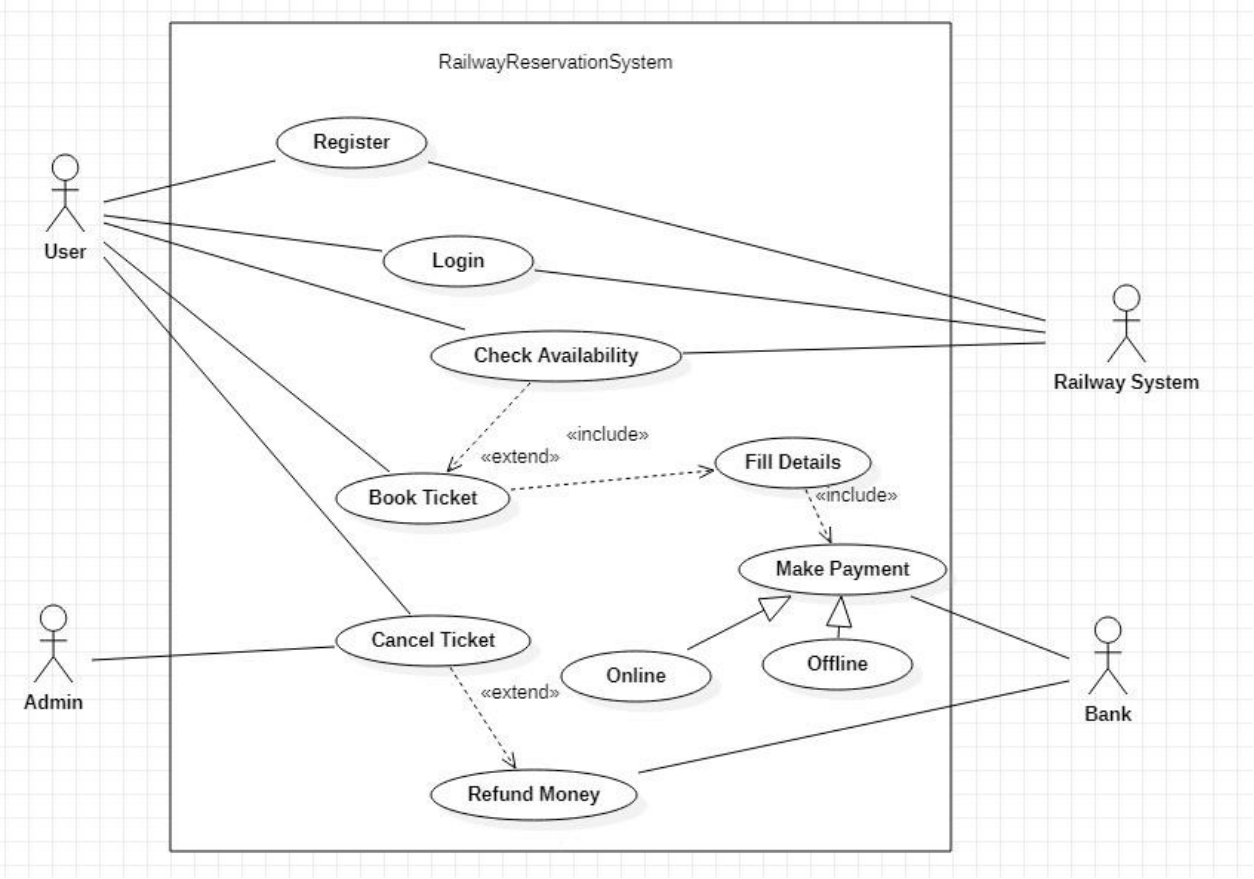




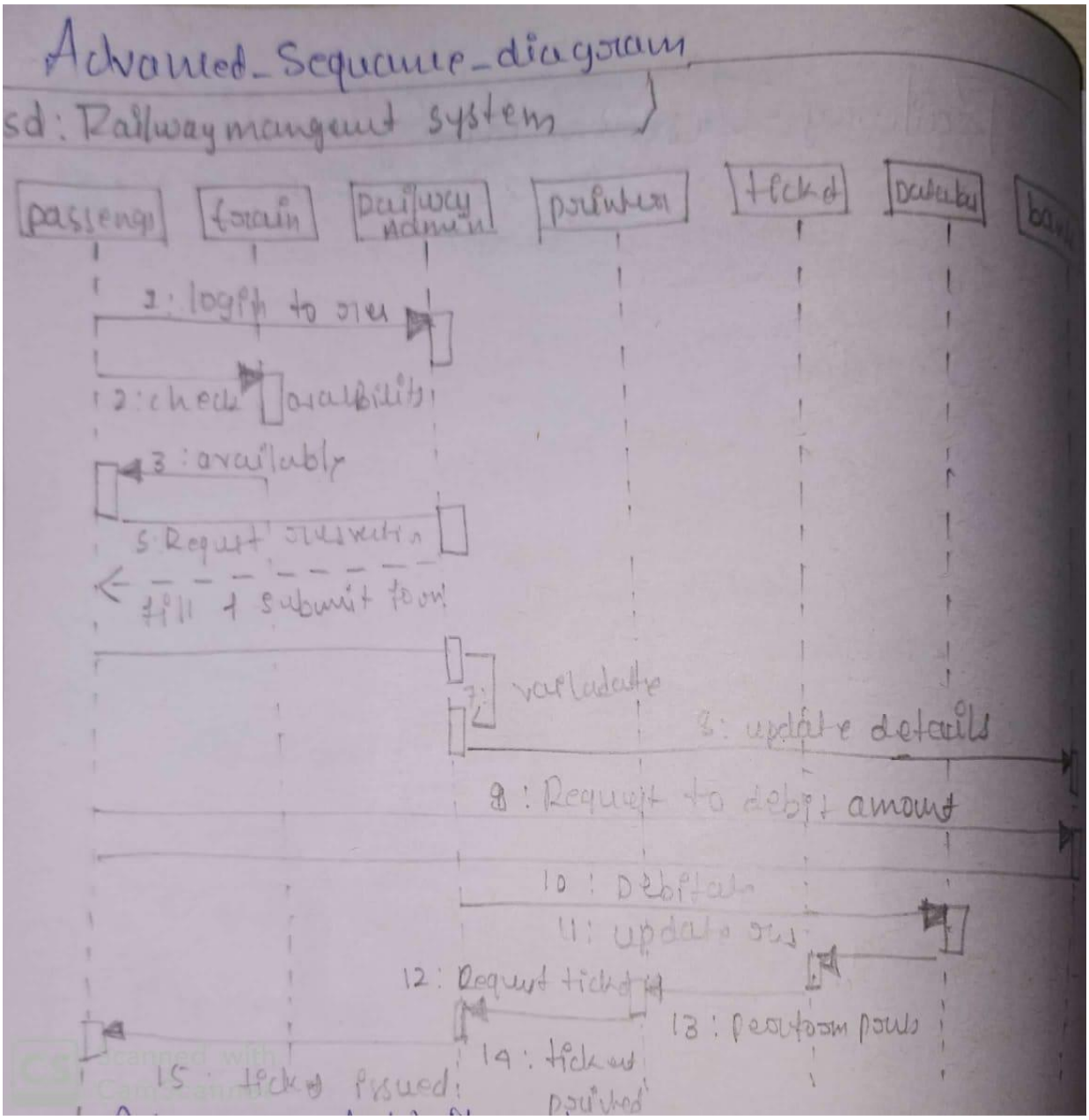
d) Advance Use Case Diagram:

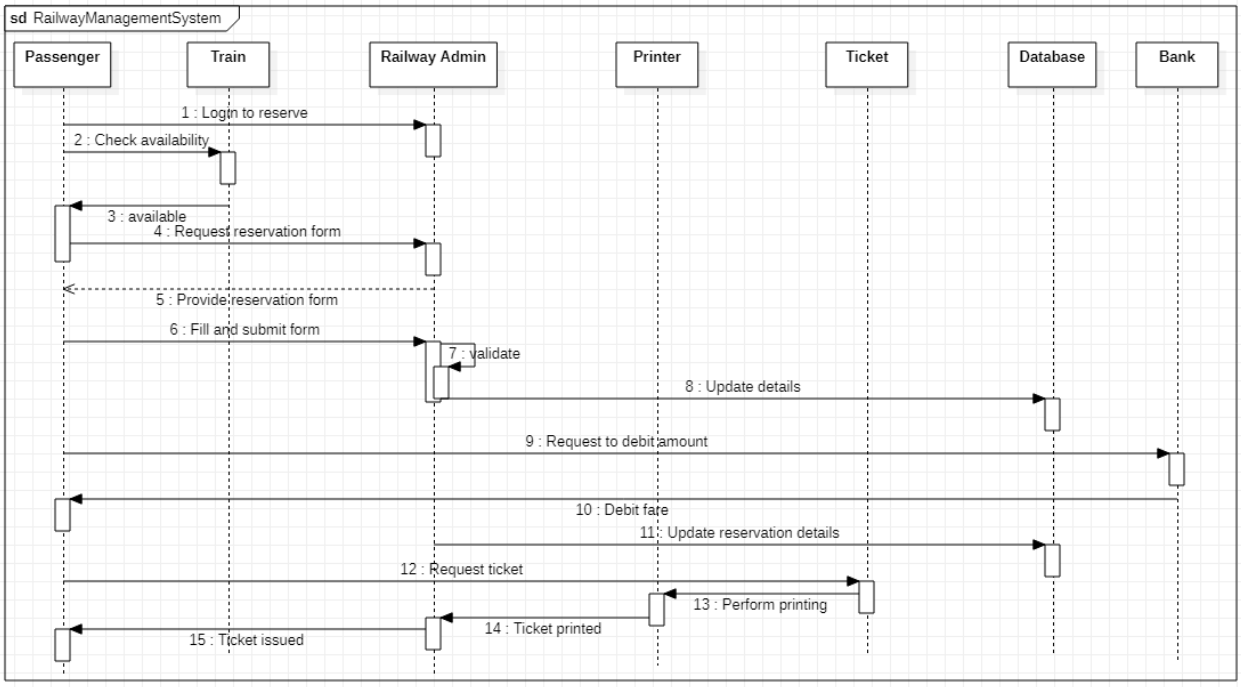
06: Railway Reservation System Usecase



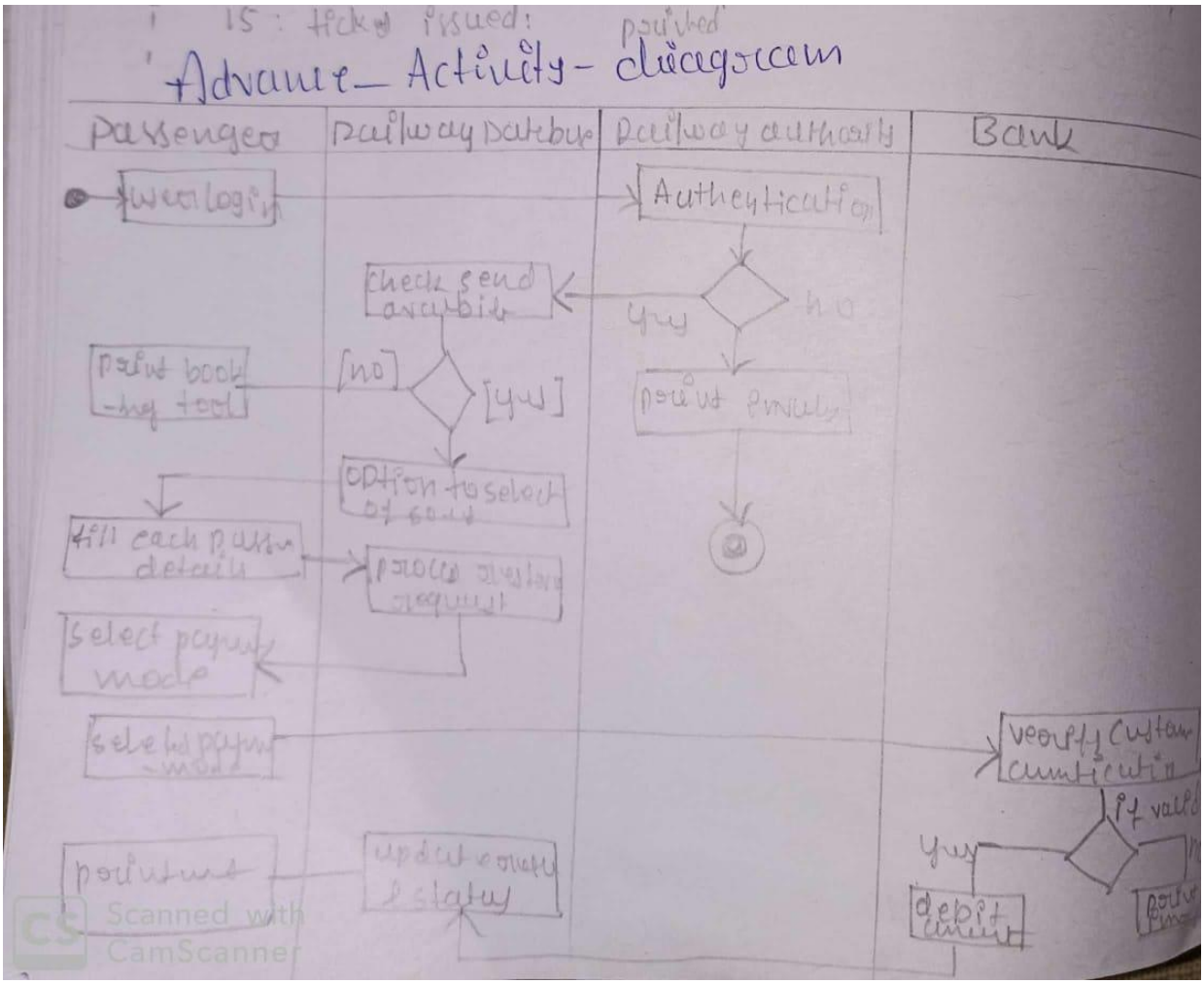


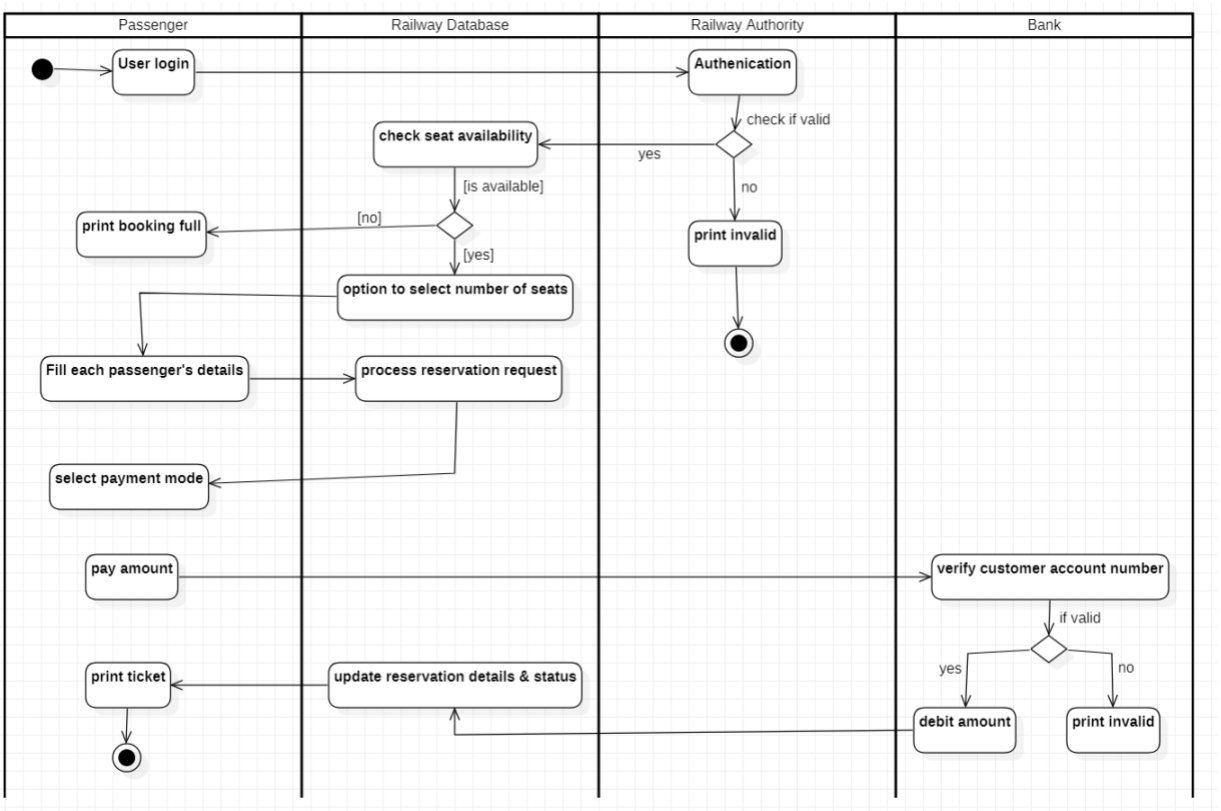
e) Sequence Diagram:


Advanced_Sequence_diagram
sd: Railway mangent system
passengr
train
Railway Admin
printer
tickt
database
1: login to sys
2: check availbility
3: availably
5: Requst
fill & submit form
validate
8: update details
9: Request to debit amount
10: Debit a/c
11: update res
12: Request ticket
13: Reserve room
14: ticket printed
15: ticket issued



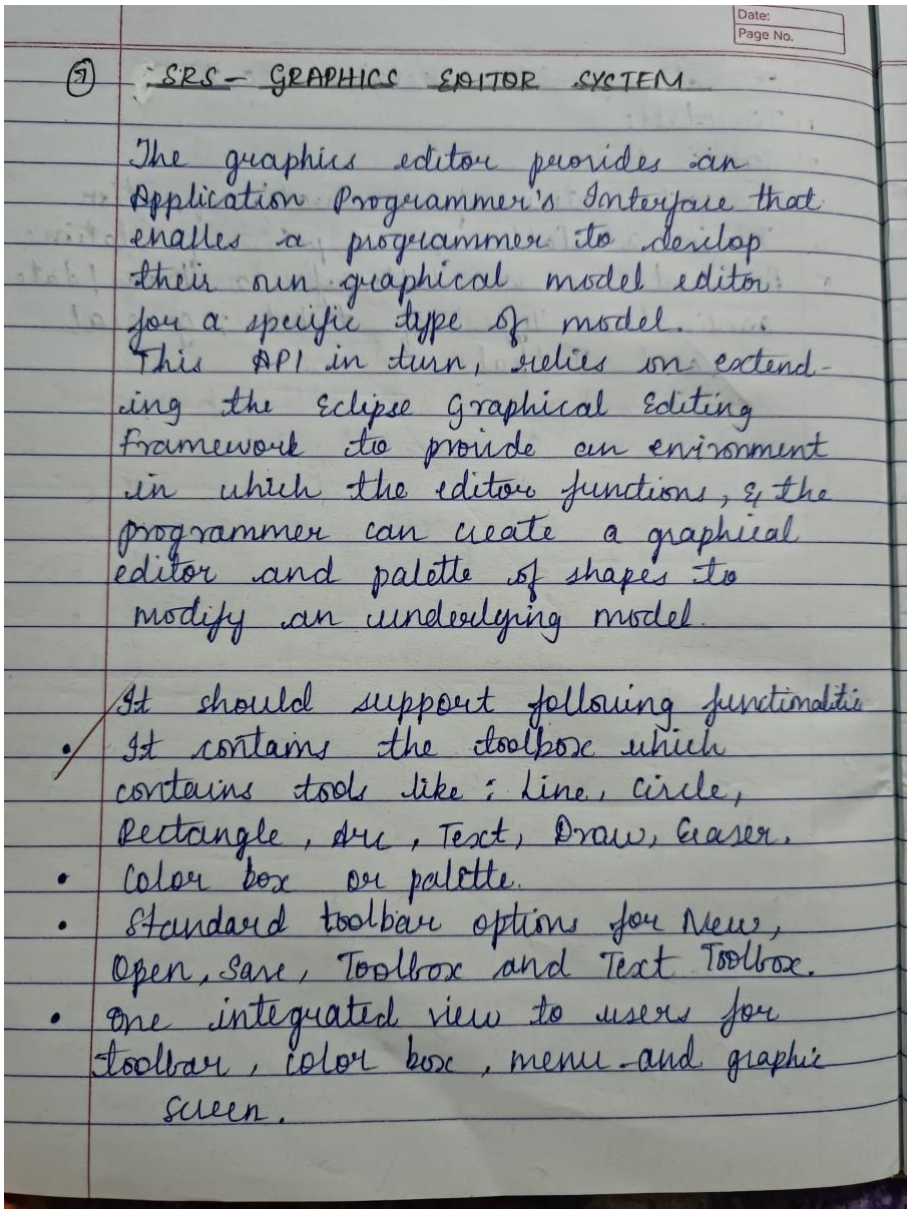
f) Activity Diagram:


Advance Activity diagram
Passenger
Railway Database
Railway authority
Bank
web login
Authentication
check seat availability
yes
no
print book-ing ticket
[no]
[yes]
print ticket
option to select of seat
fill each passe details
select payment mode
verify Customer confirmation
if valid
yes
update seat status
Scanned with CamScanner


Passenger
Railway Database
Railway Authority
Bank
User login
Authenication
check if valid
check seat availability
yes
no
[is available]
print booking full
[no]
print invalid
[yes]
option to select number of seats
Fill each passenger's details
process reservation request
select payment mode
pay amount
verify customer account number
if valid
yes
no
print ticket
update reservation details & status
debit amount
print invalid

7. Graphics Editor-

a) SRS:



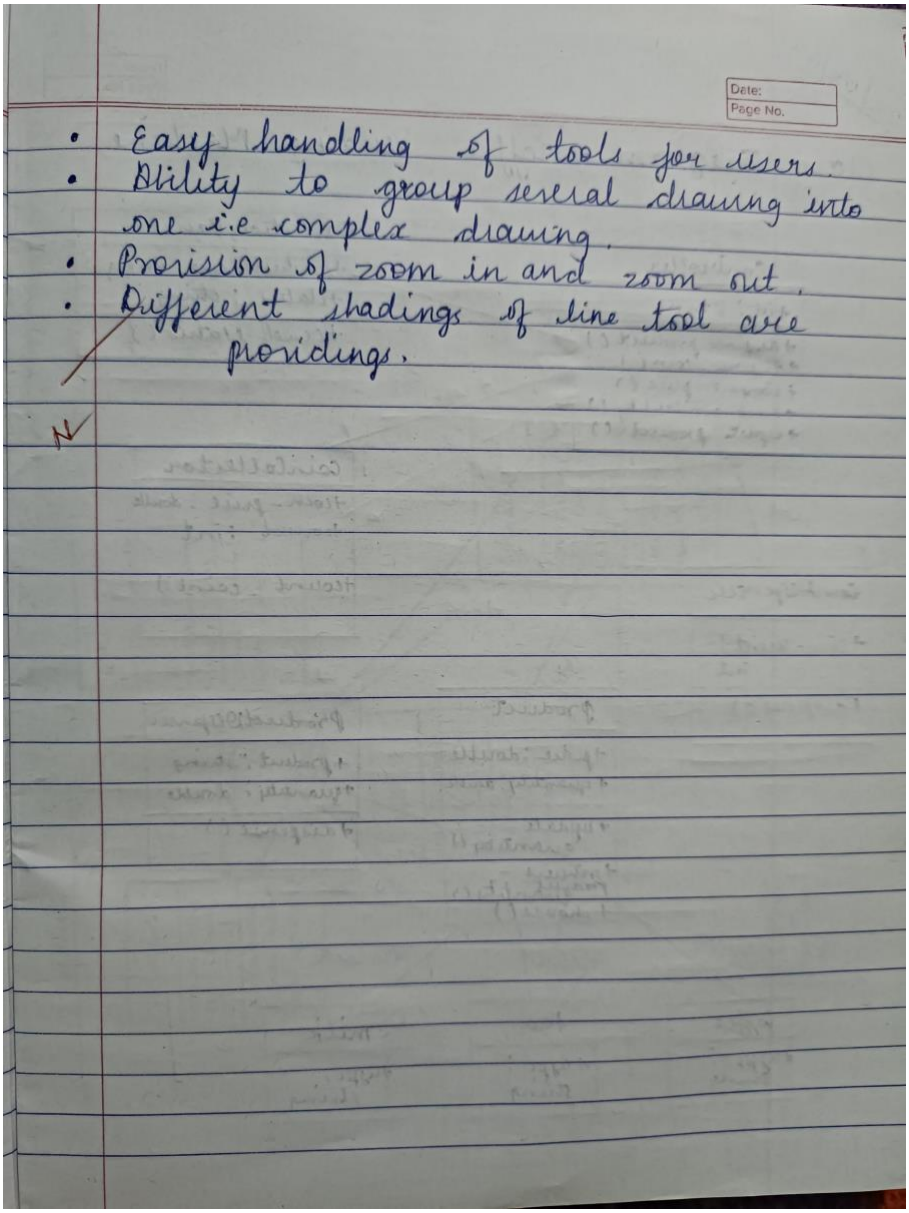
Date:
Page No.

(7) SRS – GRAPHICS EDITOR SYSTEM

The graphics editor provides an Application Programmer's Interface that enables a programmer to develop their own graphical model editor for a specific type of model. This API in turn, relies on extending the Eclipse Graphical Editing Framework to provide an environment in which the editor functions, & the programmer can create a graphical editor and palette of shapes to modify an underlying model.

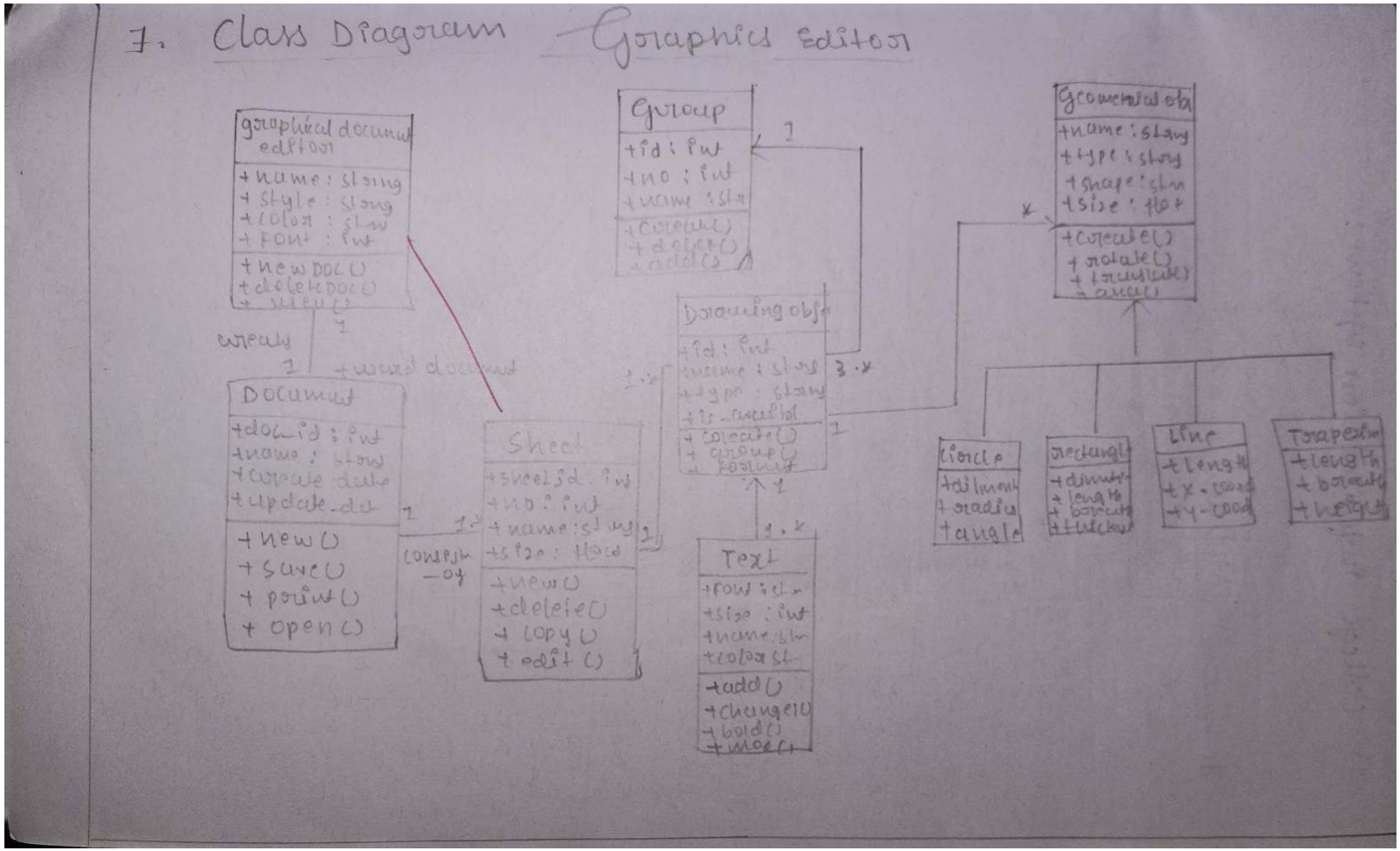
It should support following functionalities

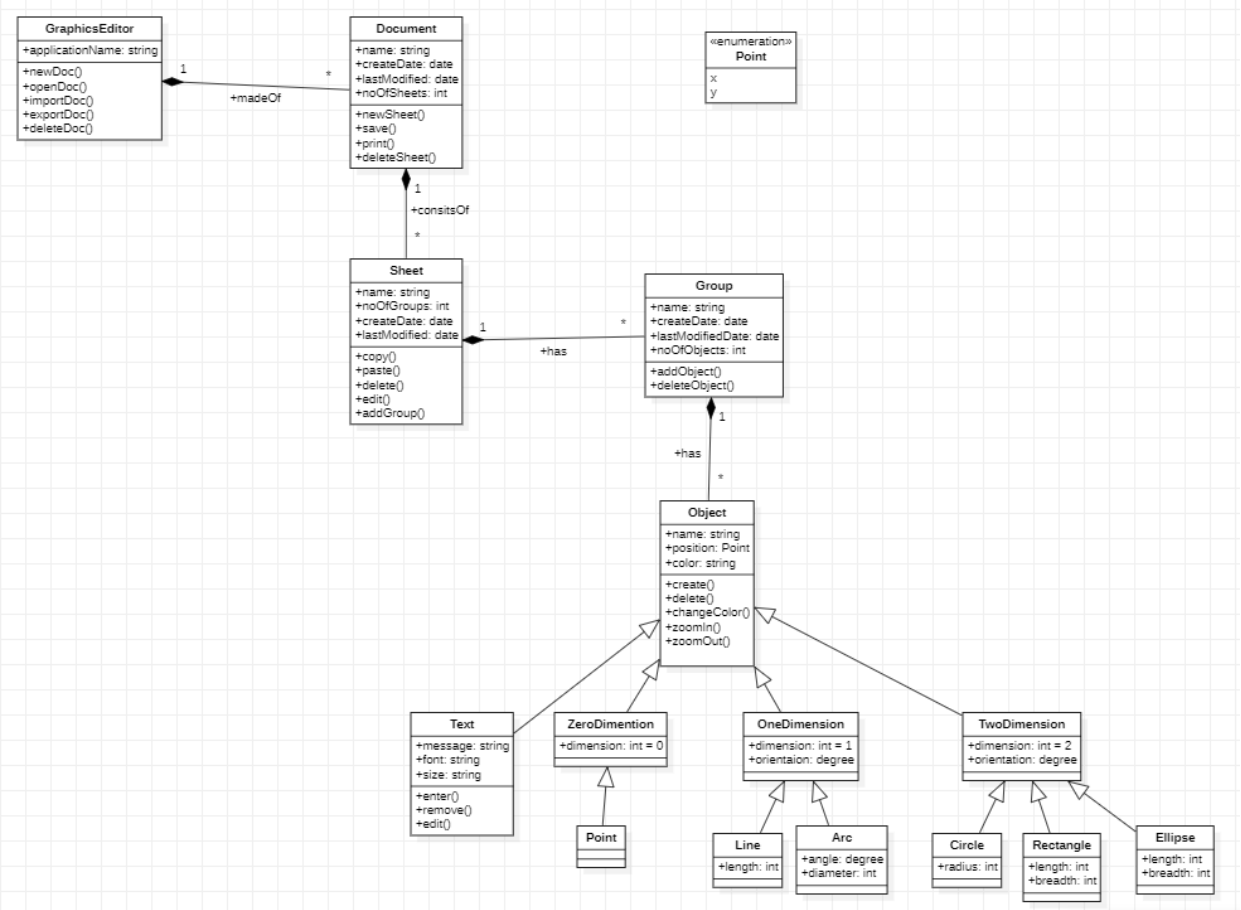
- It contains the toolbox which contains tools like : Line, Circle, Rectangle, Arc, Text, Draw, Eraser.
- Color box or palette.
- Standard toolbar options for New, Open, Save, Toolbox and Text Toolbox.
- One integrated view to users for toolbar, color box, menu and graphic screen.



- Easy handling of tools for users.
- Ability to group several drawing into one i.e complex drawing.
- Provision of zoom in and zoom out.
- Different shadings of line tool are providings.

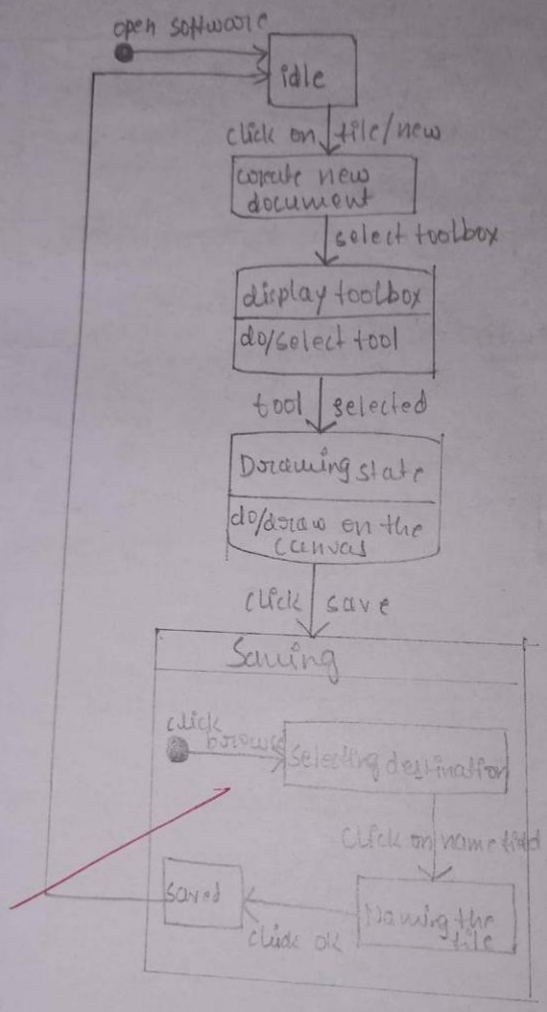
b) Advance Class Diagram:


7. Class Diagram Graphics Editor
graphical document editor
+name : string
+style : string
+color : string
+FONT : int
+newDOC()
+deleteDOC()
Document
+doc_id : int
+name : string
+create_date
+update_da
+new()
+save()
+print()
+open()
Sheet
+sheetId : int
+NO : int
+name : string
+size : float
+new()
+delete()
+copy()
+edit()
Group
+id : int
+no : int
+name : str
Drawing obj
+id : int
+name : string
+type : string
+is_selected
+calculate()
+group()
Text
+row : str
+size : int
+name : str
+color : str
+add()
+change()
+bold()
Geometrical obj
+name : string
+type : string
+shape : str
+size : float
+create()
+rotate()
+translate()
+area()
Circle
+diameter
+radius
+angle
Rectangle
+diameter
+length
+breadth
+thickness
Line
+length
+x-coord
+y-coord
Trapezium
+length
+breadth
+height

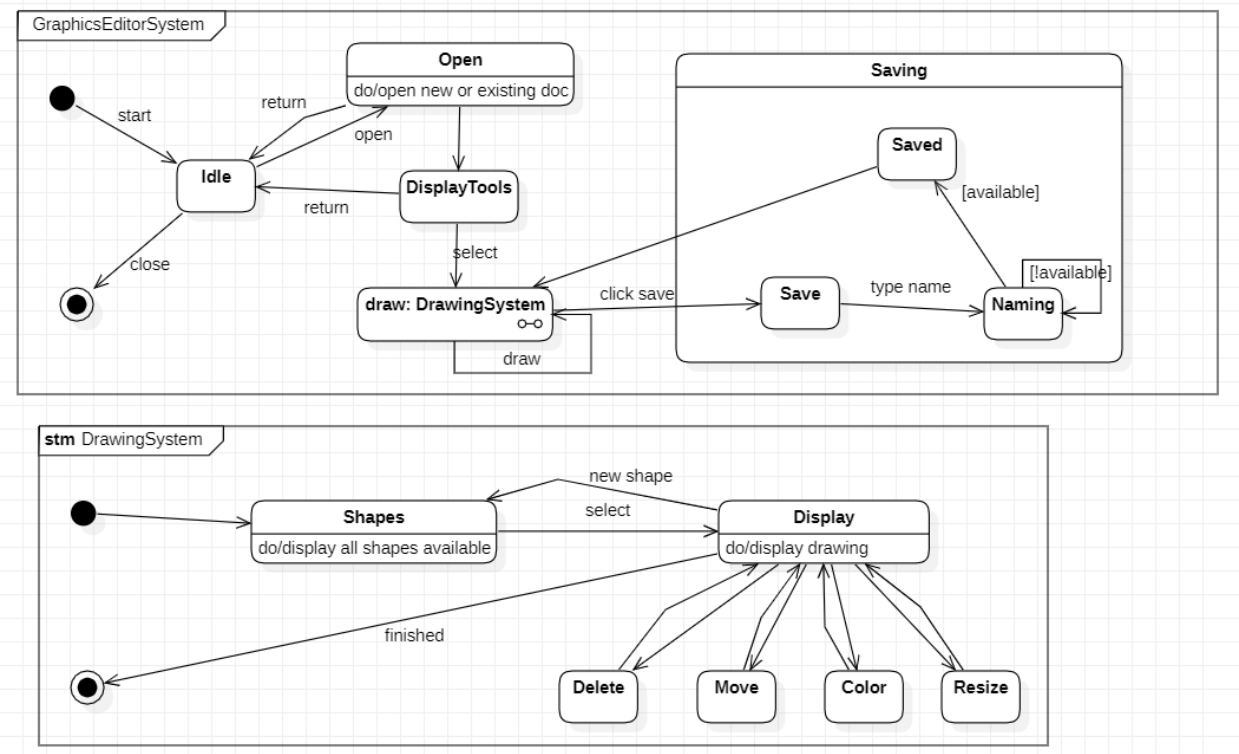


c) Advance State Diagram:

7) Graphics editor system:

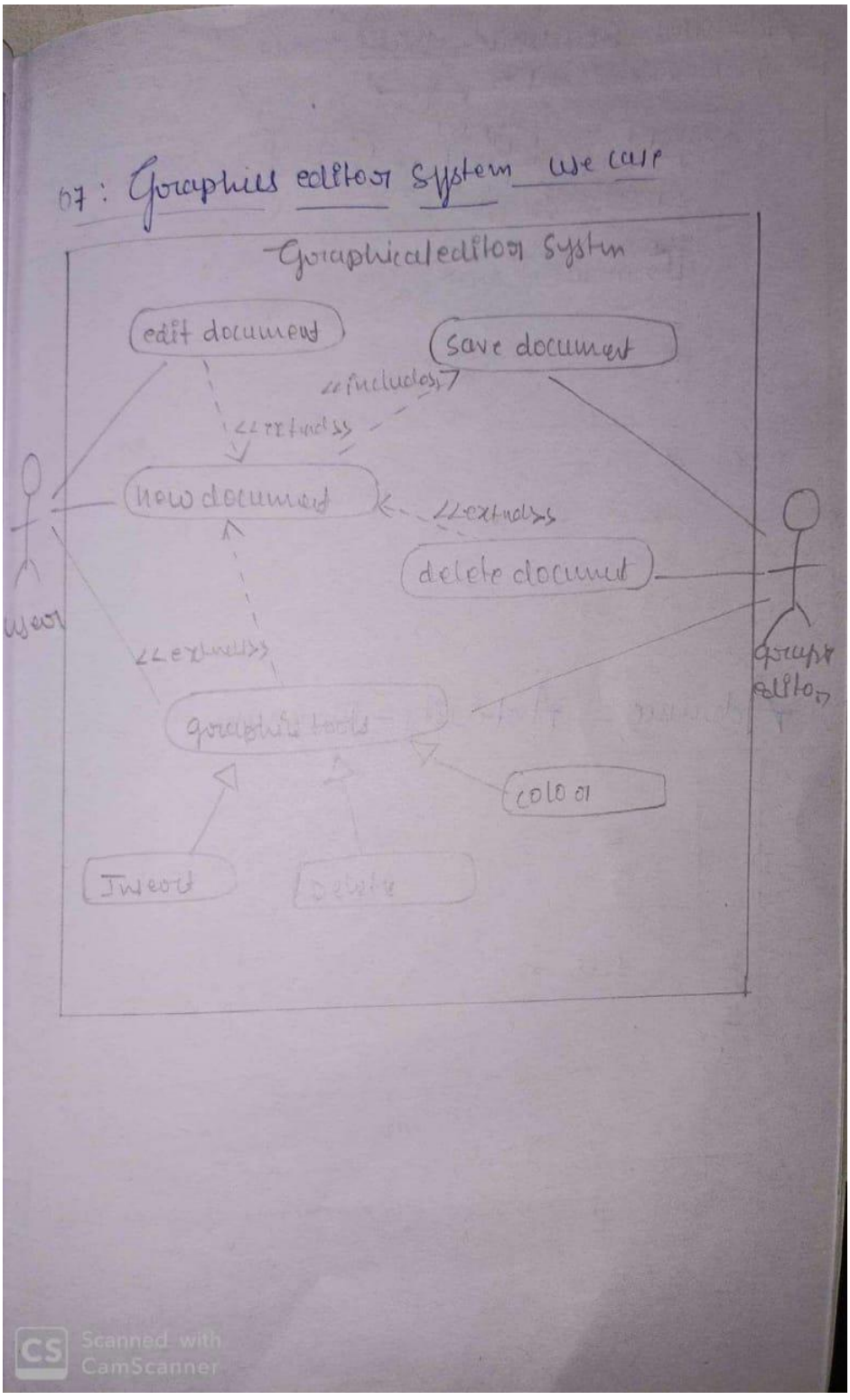


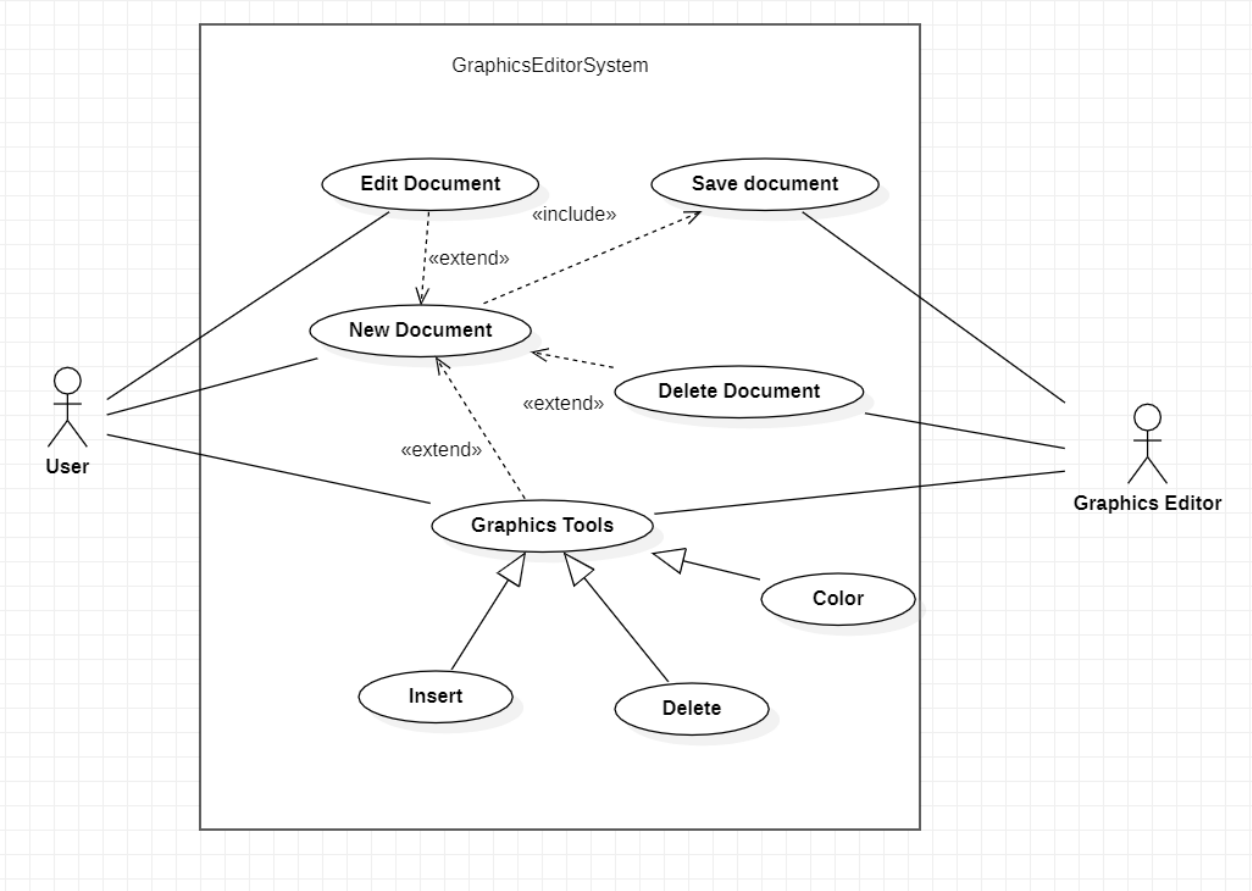
Mon 17/6/22



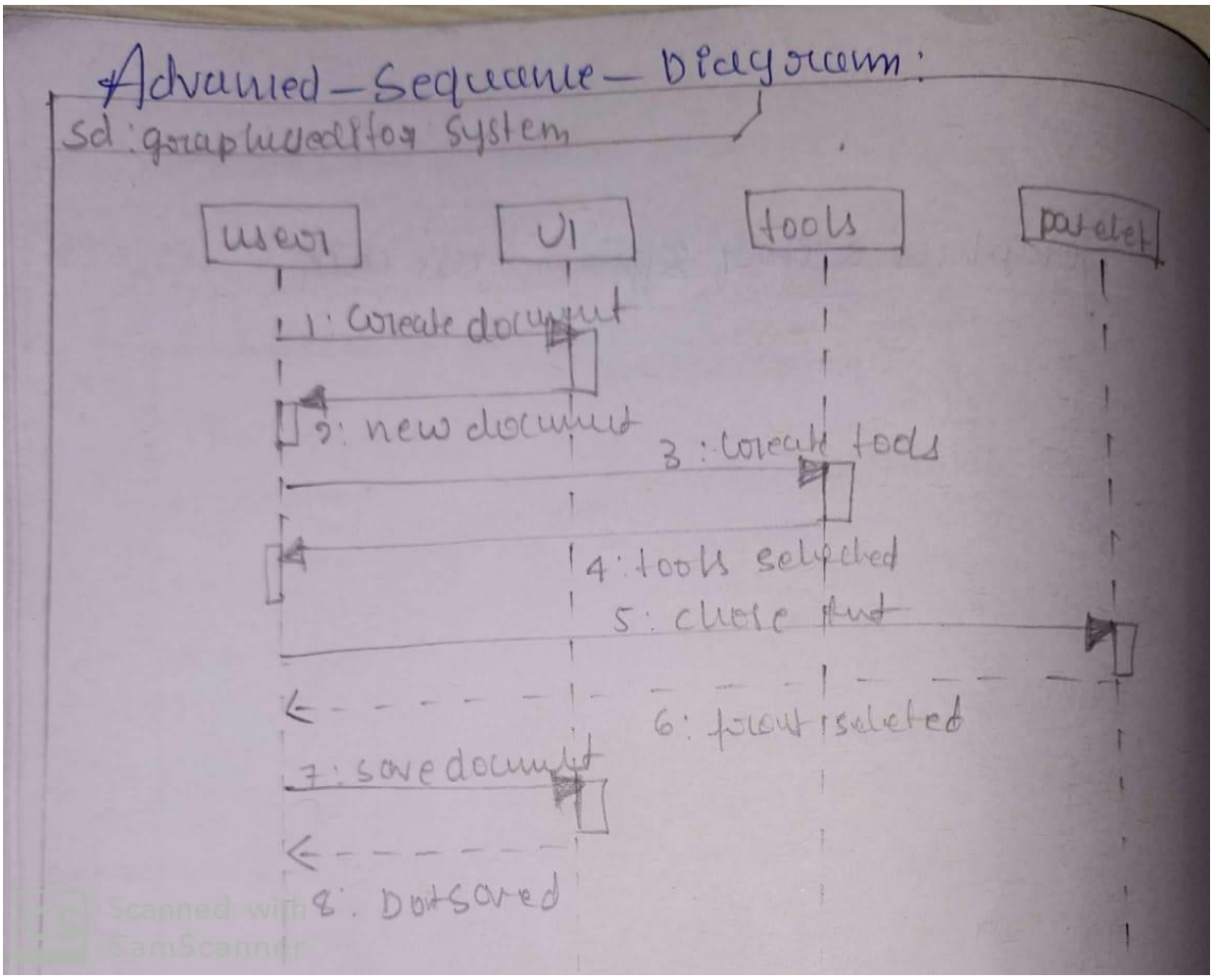
d) Advance Use Case Diagram:

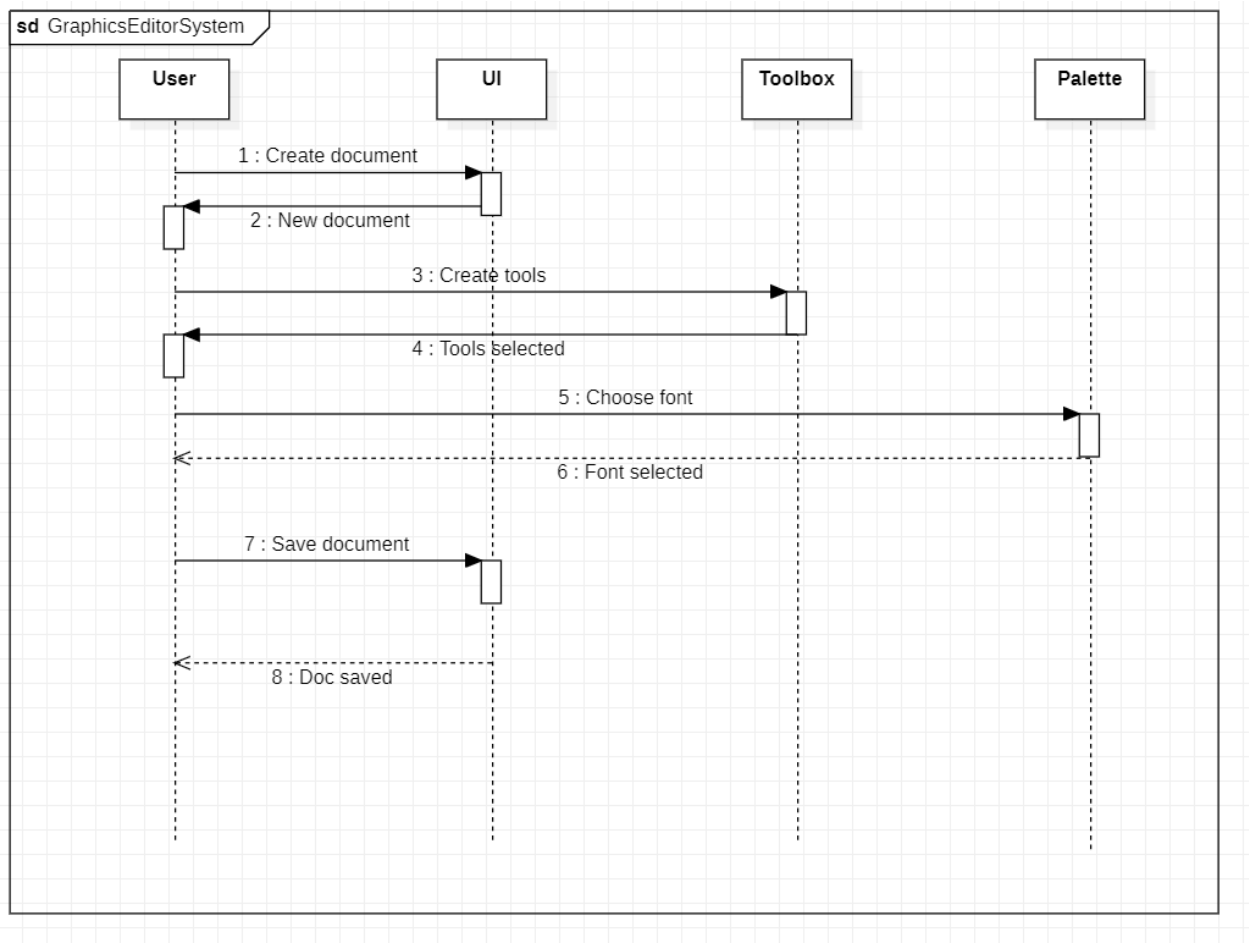
07: Graphics editor System use case





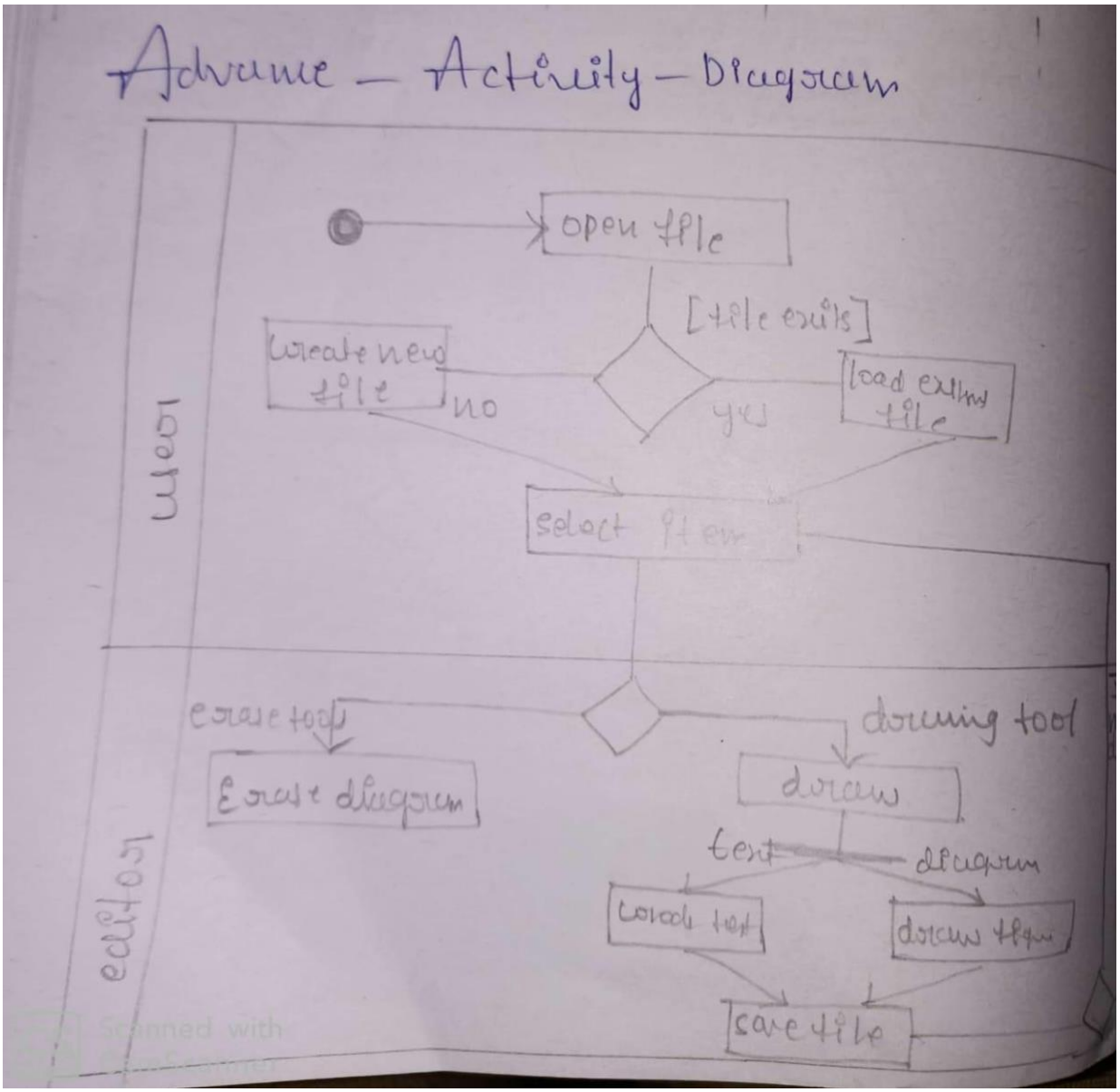
e) Sequence Diagram:

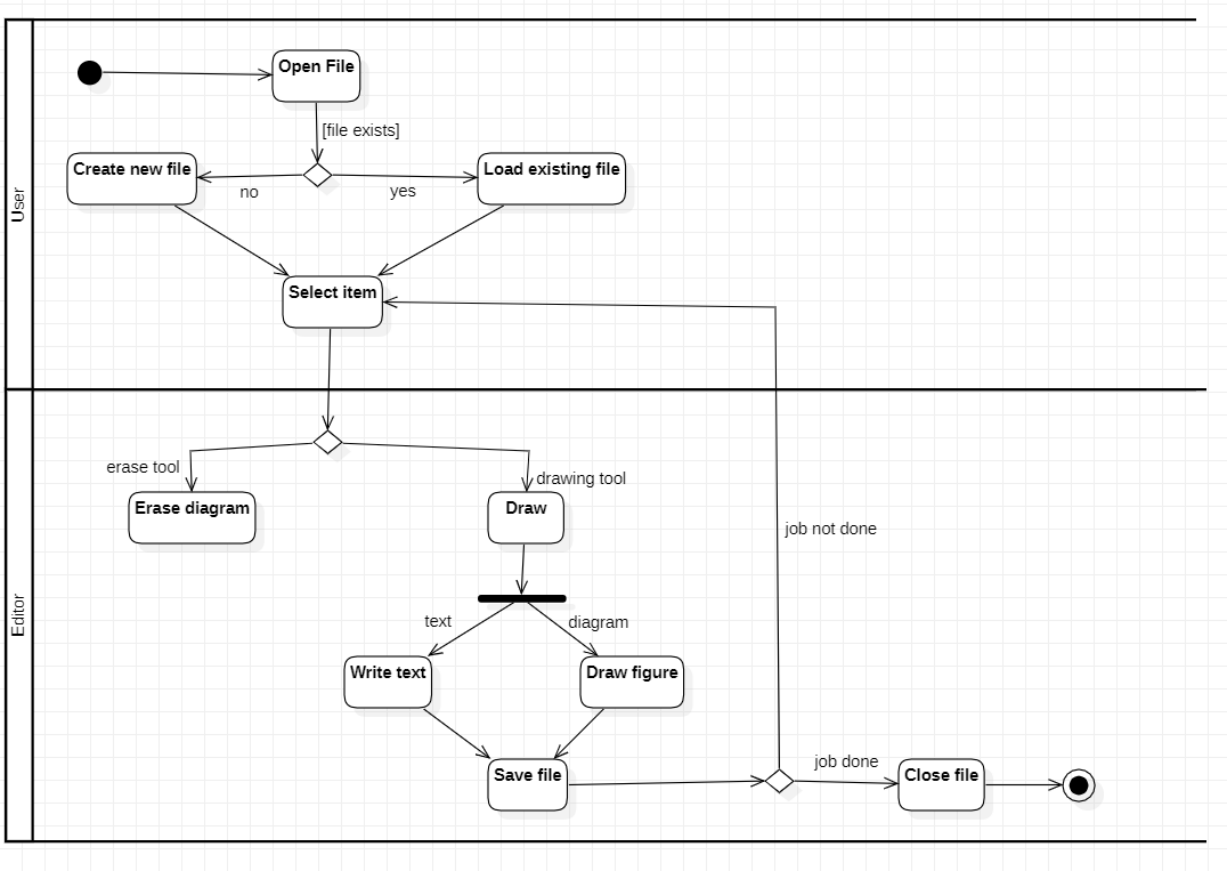

Advanced – Sequence – Diagram:
sd: graphicedit for system
user
UI
tools
parcelet
1: Create document
2: new document
3: create tools
4: tools selected
5: create tool
6: tool selected
7: save document
8: Document saved



f) Activity Diagram:

Advance – Activity – Diagram




User
Editor
Open File
[file exists]
Create new file
no
yes
Load existing file
Select item
erase tool
drawing tool
Erase diagram
Draw
job not done
text
diagram
Write text
Draw figure
Save file
job done
Close file