VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

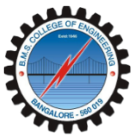
LAB REPORT
on

OBJECT ORIENTED MODELING

Submitted by

Rohan Siwach (1BM19CS132)

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 "LAB COURSE **Object Oriented Modeling**" carried out by **Rohan Siwach (1BM19CS132),** 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 a **Object oriented modeling - ()** work prescribed for the said degree.

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

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

`

Index Sheet

Course Outcome

C04	Ability to conduct practical experiments to solve a given problem using Unified Modeling Language

Lab 1 college information system

Problem Statement: 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.

Software Requirements:

● Educational institutions should be able to add, edit and view student personal details, like name, age, gender, email, phone number, address and so on.

● Educational institutions should be able to add, edit and view student academic details, like USN, department, semester and registered courses.

● Faculty should be able to view all student personal 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 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.

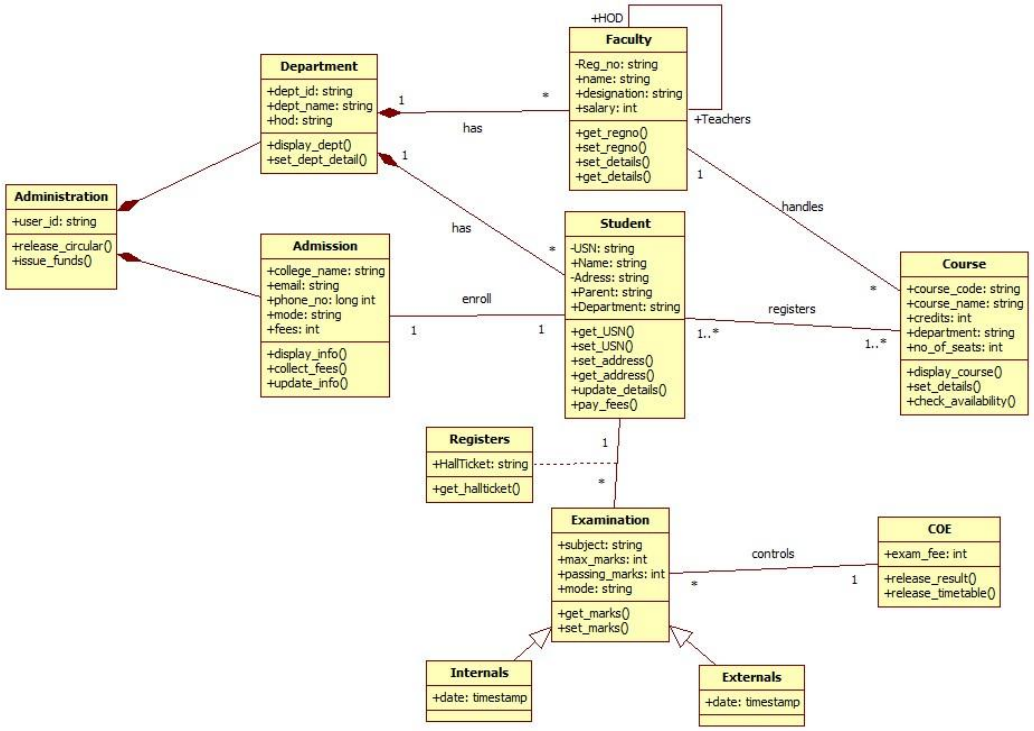
● Management section should be able to view, add and edit teaching and non-teaching staff details.

● Students should not be allowed to edit their personal or academic details.

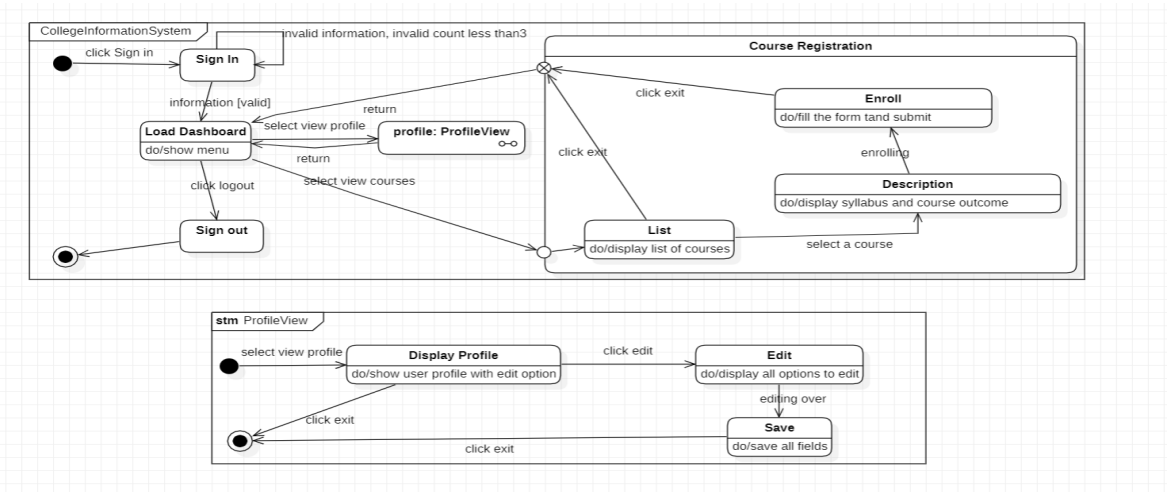
● The system should be convenient and easy to use by students, management and faculty.

● The system should be a reliable source of information viewing (most importantly, academic grades) for students, COE and faculty

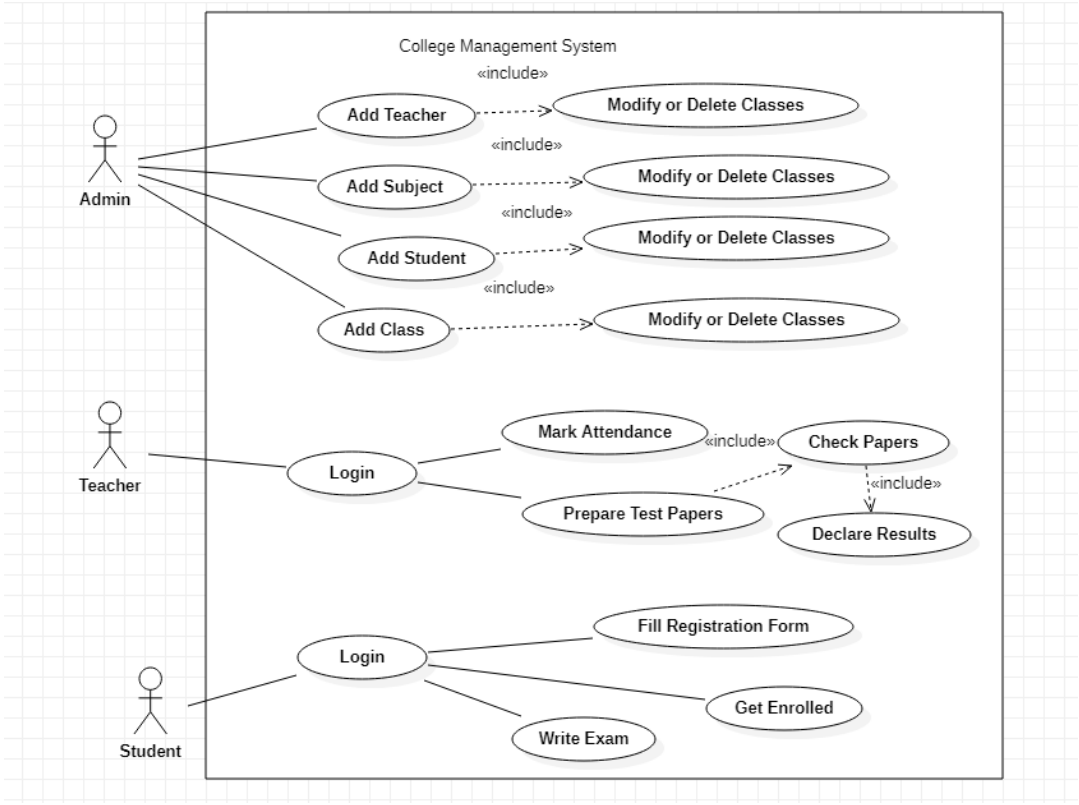
Class diag:



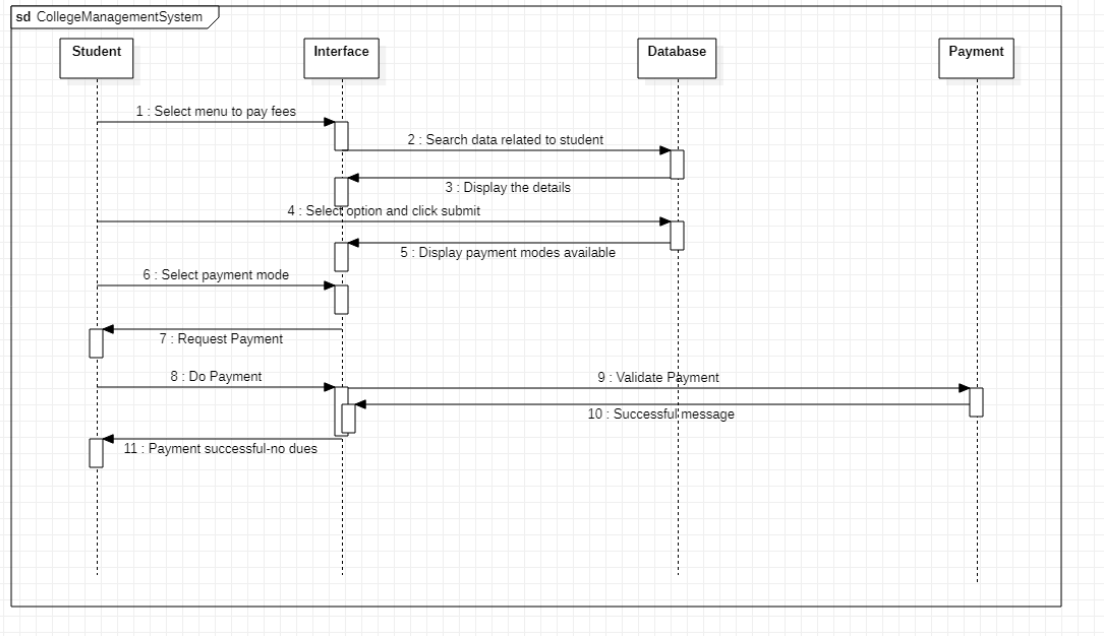
State Diag:



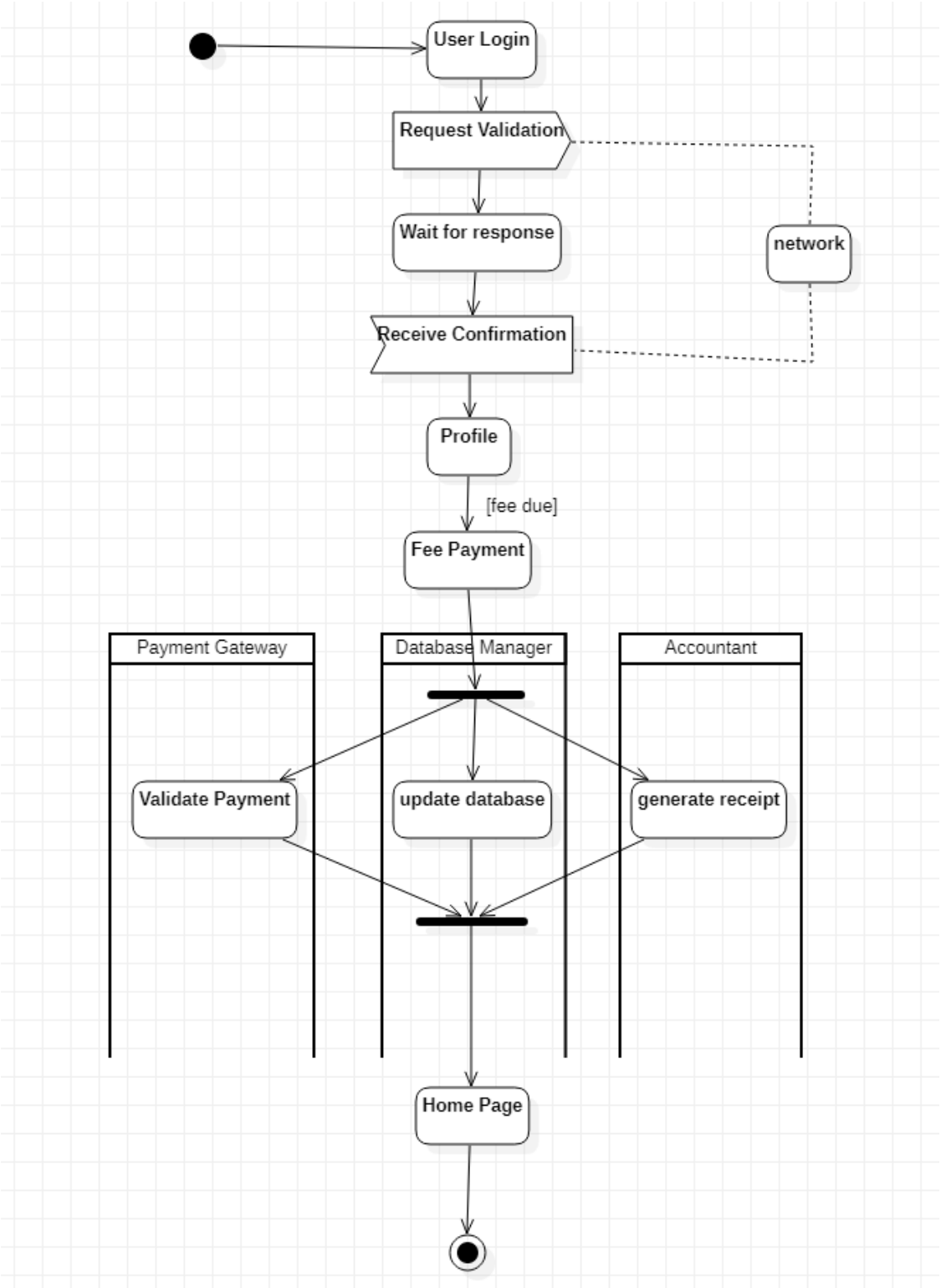
Use case Diag:



Sequence diag:



Activity diag:



Lab 2 Hostel management System

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

Software Requirements:

● 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.

● Admin can appoint staff.

● Students can provide message feedback.

● Mess managers can review the mess feedback.

● Mess manager can update the menu list

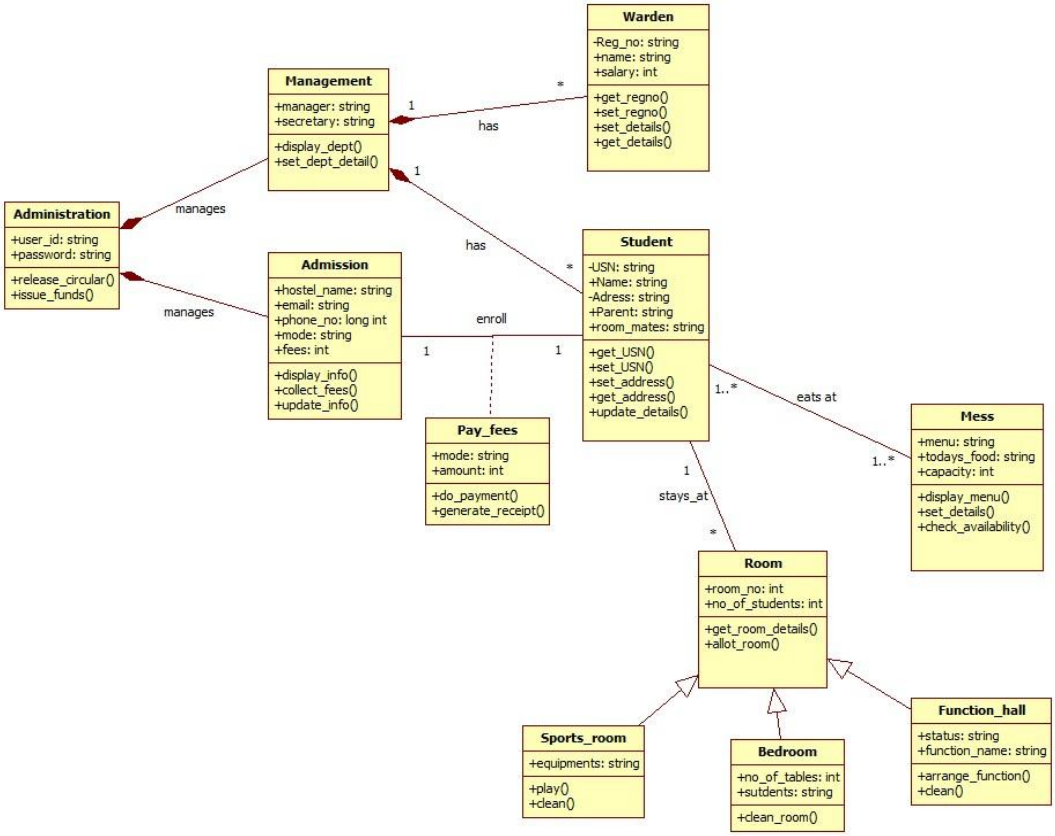
. ● Admin can assign work to staff members.

● The system should be easy to handle.

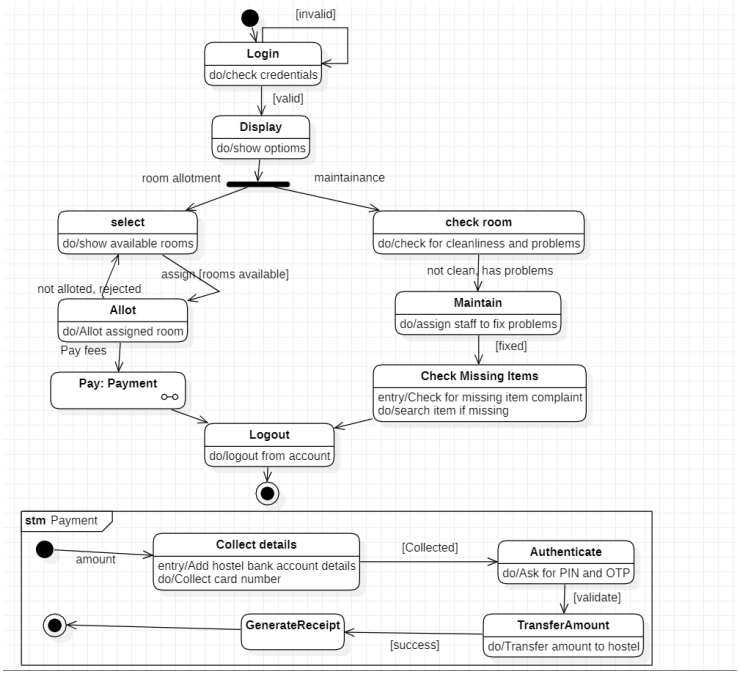
● System should give expected performance results.

● The response time should be small

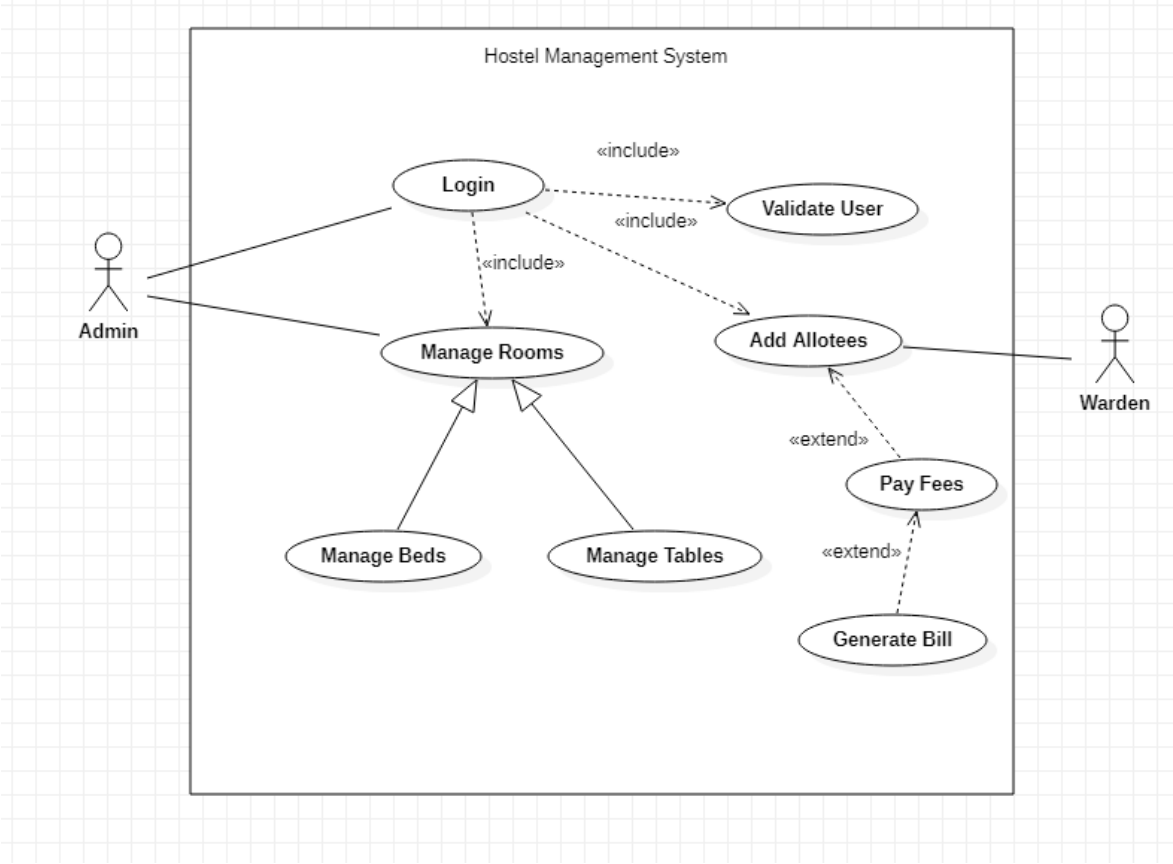
Class diag:


Warden
-Reg_no: string
+name: string
+salary: int
+get_regno()
+set_regno()
+set_details()
+get_details()
Management
+manager: string
+secretary: string
+display_dept()
+set_dept_detail()
1
has
*
1
has
*
Administration
+user_id: string
+password: string
+release_circular()
+issue_funds()
manages
manages
Admission
+hostel_name: string
+email: string
+phone_no: long int
+mode: string
+fees: int
+display_info()
+collect_fees()
+update_info()
enroll
1
1
Pay_fees
+mode: string
+amount: int
+do_payment()
+generate_receipt()
Student
-USN: string
+Name: string
-Adress: string
+Parent: string
+room_mates: string
+get_USN()
+set_USN()
+set_address()
+get_address()
+update_details()
1..*
eats at
1..*
Mess
+menu: string
+todays_food: string
+capacity: int
+display_menu()
+set_details()
+check_availability()
1
stays_at
*
Room
+room_no: int
+no_of_students: int
+get_room_details()
+allot_room()
Sports_room
+equipments: string
+play()
+clean()
Bedroom
+no_of_tables: int
+sutdents: string
+clean_room()
Function_hall
+status: string
+function_name: string
+arrange_function()
+clean()

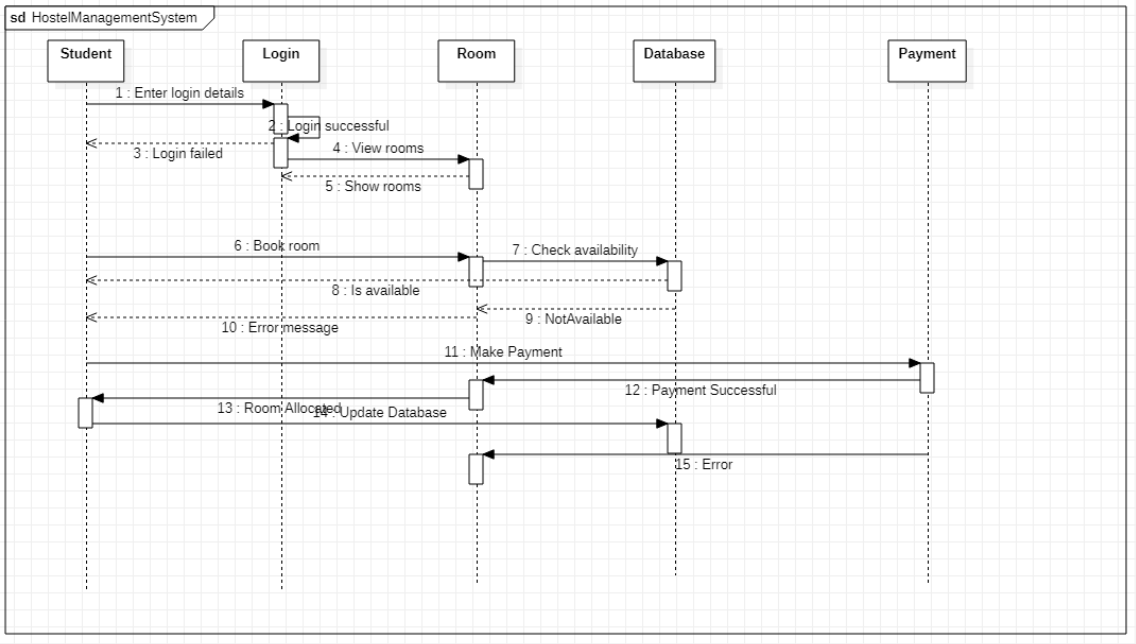
State diag:


[invalid]
Login
do/check credentials
[valid]
Display
do/show optioms
room allotment
maintainance
select
do/show available rooms
check room
do/check for cleanliness and problems
assign [rooms available]
not alloted, rejected
not clean, has problems
Allot
do/Allot assigned room
Maintain
do/assign staff to fix problems
Pay fees
[fixed]
Pay: Payment
Check Missing Items
entry/Check for missing item complaint
do/search item if missing
Logout
do/logout from account
stm Payment
amount
Collect details
entry/Add hostel bank account details
do/Collect card number
[Collected]
Authenticate
do/Ask for PIN and OTP
[validate]
GenerateReceipt
[success]
TransferAmount
do/Transfer amount to hostel

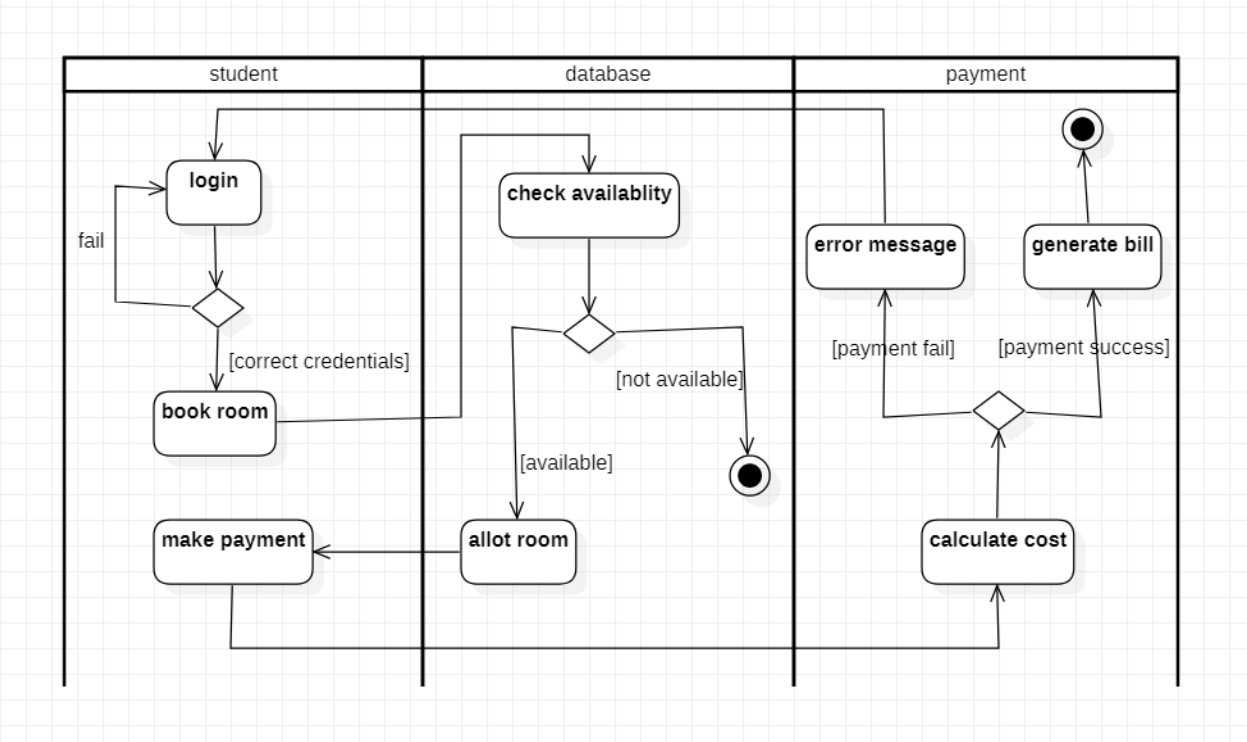
Usecase:



sequence diag:



activity diag :



Lab 3 Stock Maintence

Problem Statement: 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 and the price of the items. The customer can just view them but cannot make any changes.

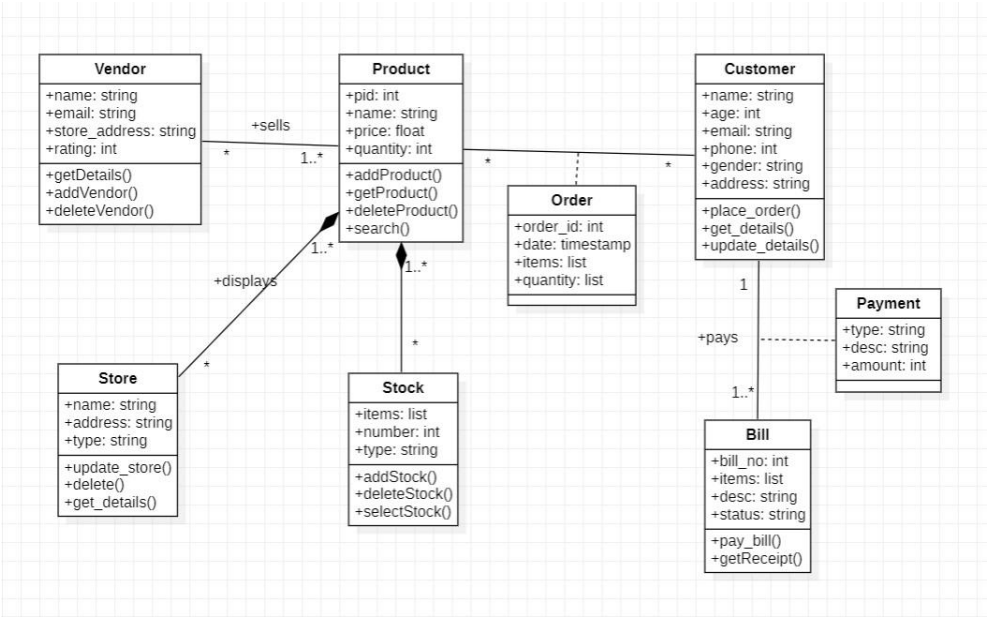
Software Requirements: The process of the stock maintenance system is that the customer logs in to the particular site to place the order for the customer product.

The stock maintenance system is described sequentially through steps

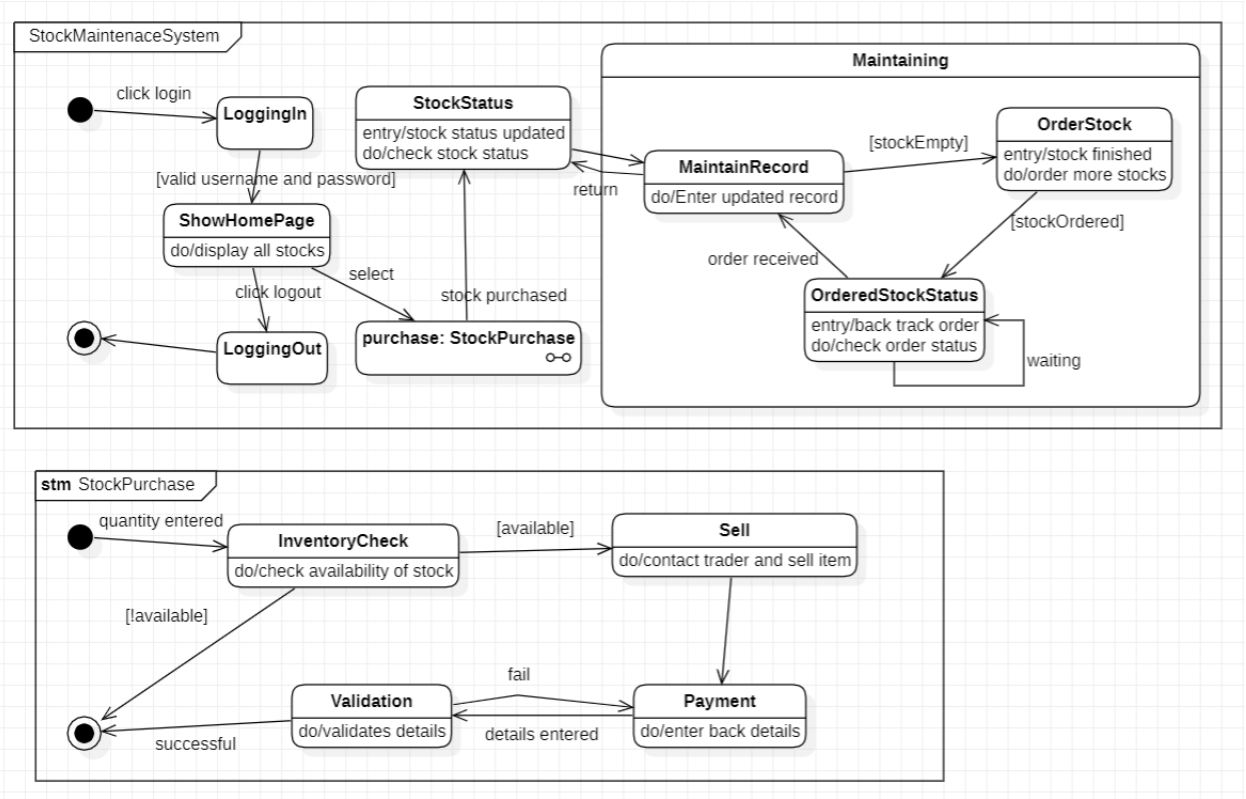
- The customer logs in to the particular site.
- They fill the customer details.
- They place the orders for their product

. • The vendor logs in and views the customer details and order

Class Diag:


Vendor
+name: string
+email: string
+store_address: string
+rating: int
+getDetails()
+addVendor()
+deleteVendor()
+sells
*
1..*
Product
+pid: int
+name: string
+price: float
+quantity: int
+addProduct()
+getProduct()
+deleteProduct()
+search()
Order
+order_id: int
+date: timestamp
+items: list
+quantity: list
Customer
+name: string
+age: int
+email: string
+phone: int
+gender: string
+address: string
+place_order()
+get_details()
+update_details()
+displays
Store
+name: string
+address: string
+type: string
+update_store()
+delete()
+get_details()
Stock
+items: list
+number: int
+type: string
+addStock()
+deleteStock()
+selectStock()
1
+pays
Payment
+type: string
+desc: string
+amount: int
Bill
+bill_no: int
+items: list
+desc: string
+status: string
+pay_bill()
+getReceipt()

State diag:


StockMaintenaceSystem
click login
LoggingIn
[valid username and password]
ShowHomePage
do/display all stocks
click logout
LoggingOut
select
stock purchased
purchase: StockPurchase
StockStatus
entry/stock status updated
do/check stock status
return
Maintaining
MaintainRecord
do/Enter updated record
[stockEmpty]
OrderStock
entry/stock finished
do/order more stocks
[stockOrdered]
order received
OrderedStockStatus
entry/back track order
do/check order status
waiting
stm StockPurchase
quantity entered
InventoryCheck
do/check availability of stock
[available]
Sell
do/contact trader and sell item
[!available]
fail
Validation
do/validates details
Payment
do/enter back details
details entered
successful

Lab4 : Coffee vending machine

Problem Statement: The Objective of the system is to prepare a coffee vending machine for commercial purposes. The system will be able to prepare coffee by processing all its required ingredients. Users will be provided with sophisticated and easy to use user interfaces.

Software Requirements: There are many different types of coffee makers using a number of different brewing principles, in the most common devices, coffee grounds are placed in a paper or metal filter inside a funnel, which is set over a glass or ceramic coffee pot, a cooking pot in the kettle family. Cold water is poured into a separate chamber, which is then heated up to the boiling point, and directed into the funnel.

• Cash Box:Knows amount of money put in; Give change; Knows price of coffee; Turns front panel on and off.

• Front panel:Captures selection; Knows what to mix in each; Instructs mixer when to mix. • Mixer:Knows how to talk to the dispensers.

• Dispenser [cup-, coffee powder-, sugar-, creamer-, water-]:Knows how to dispense a fixed amount, knows when it is empty. Features :

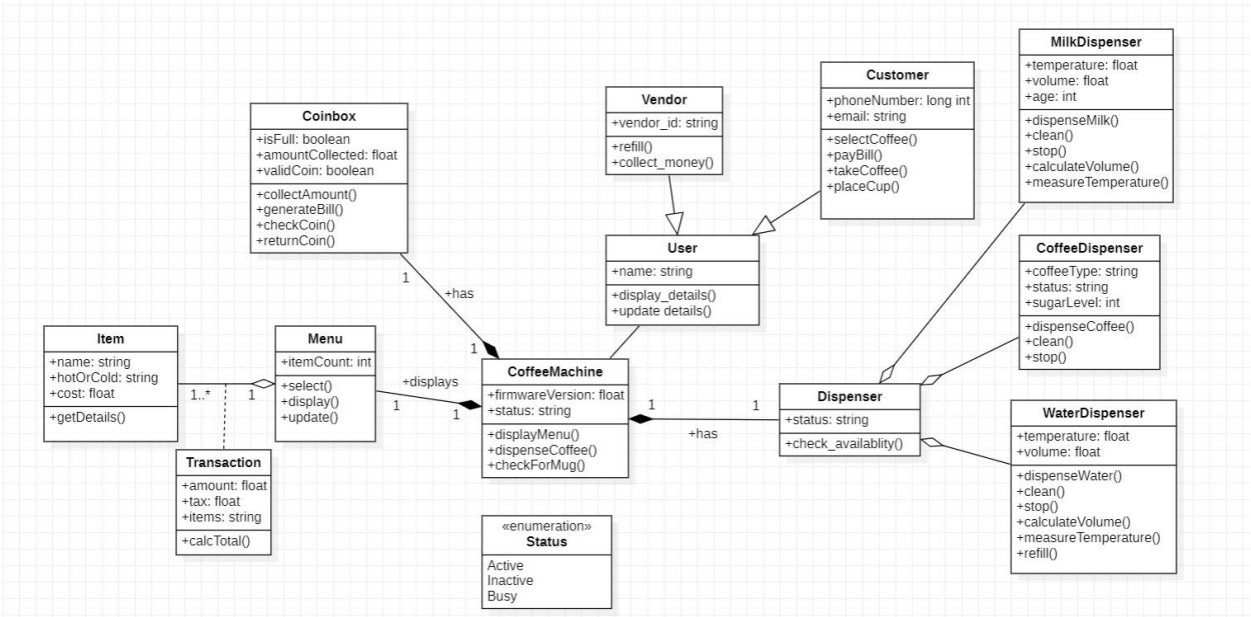
• Small carbon footprint

• Energy saving advanced power management system

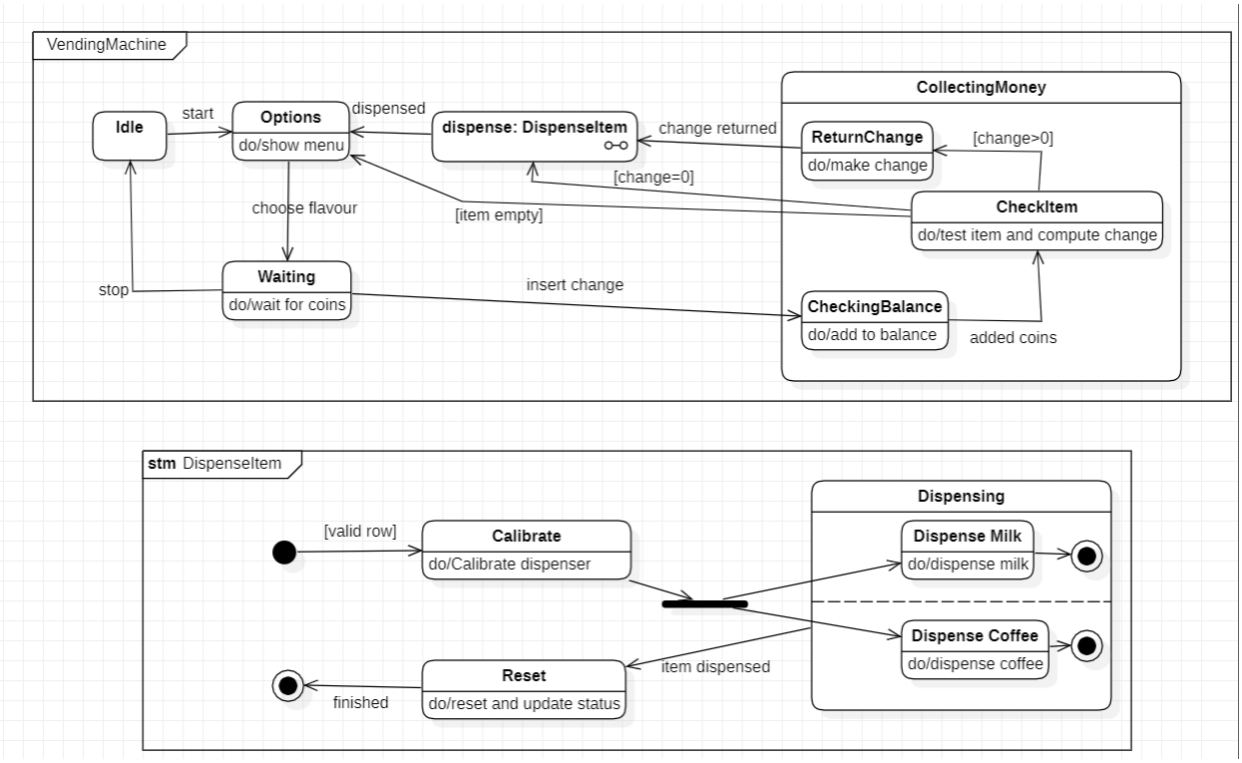
• Comprehensive drink range

• Simple user interface

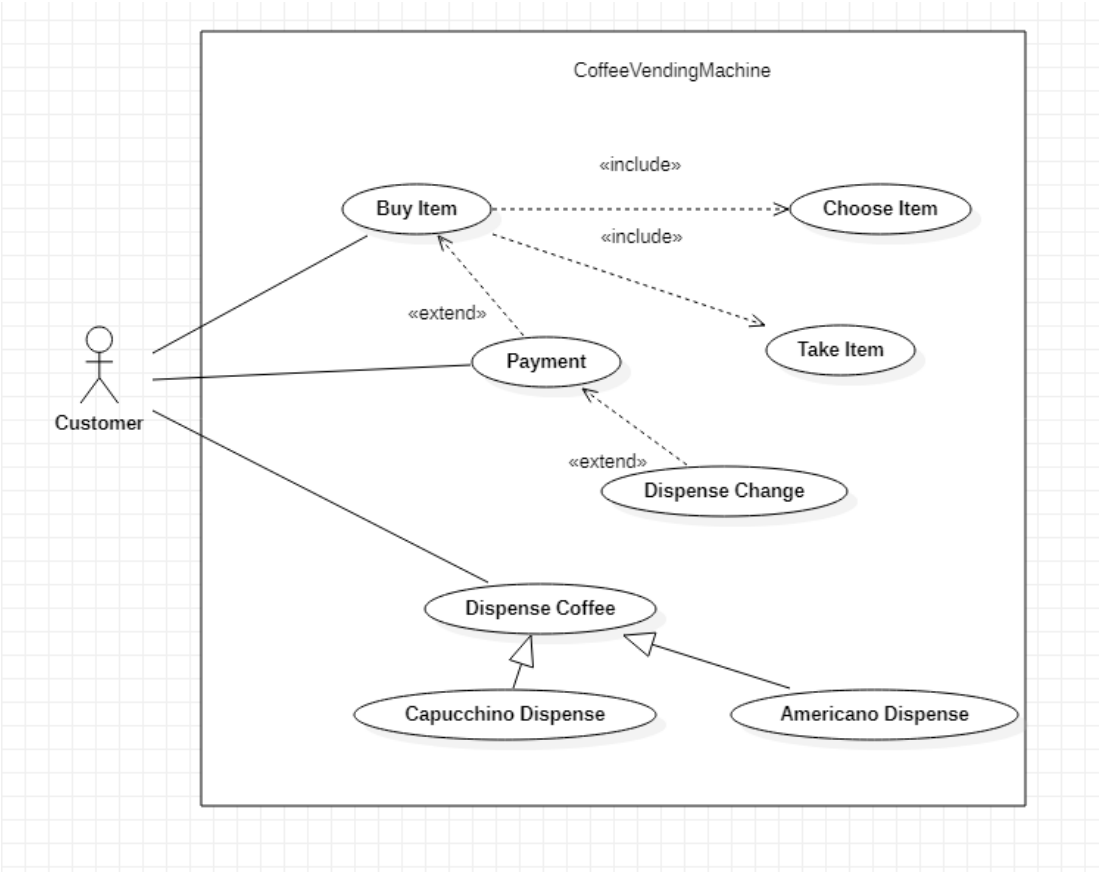
Class diag:



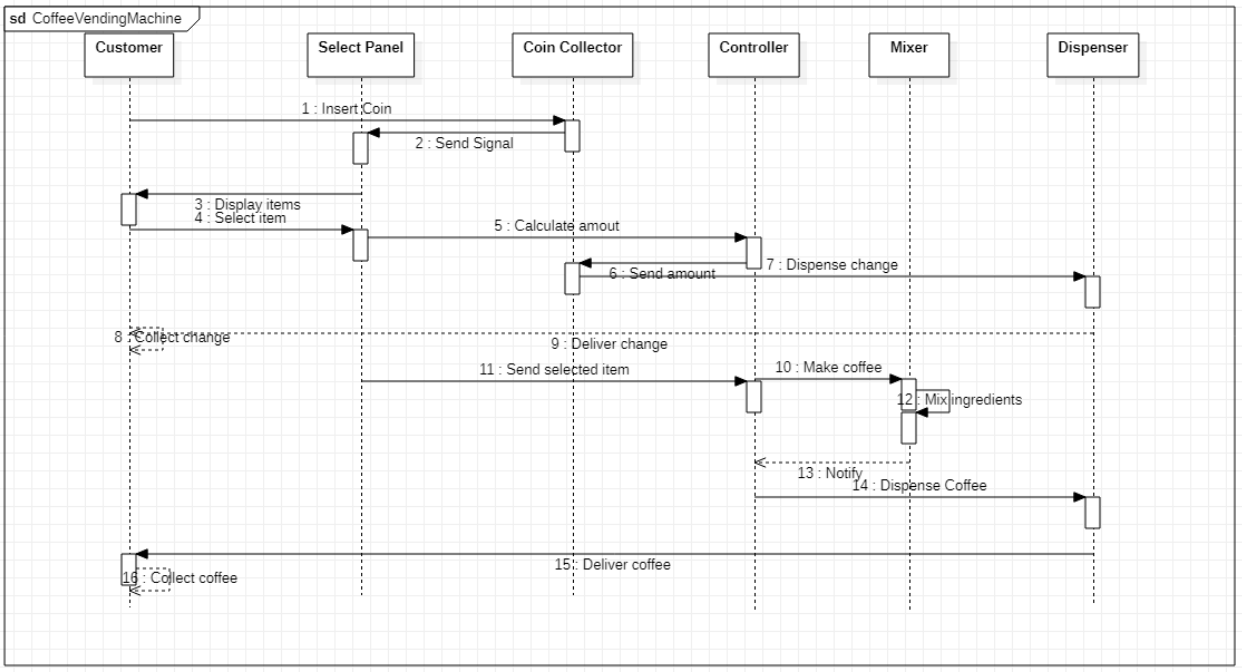
State diag:



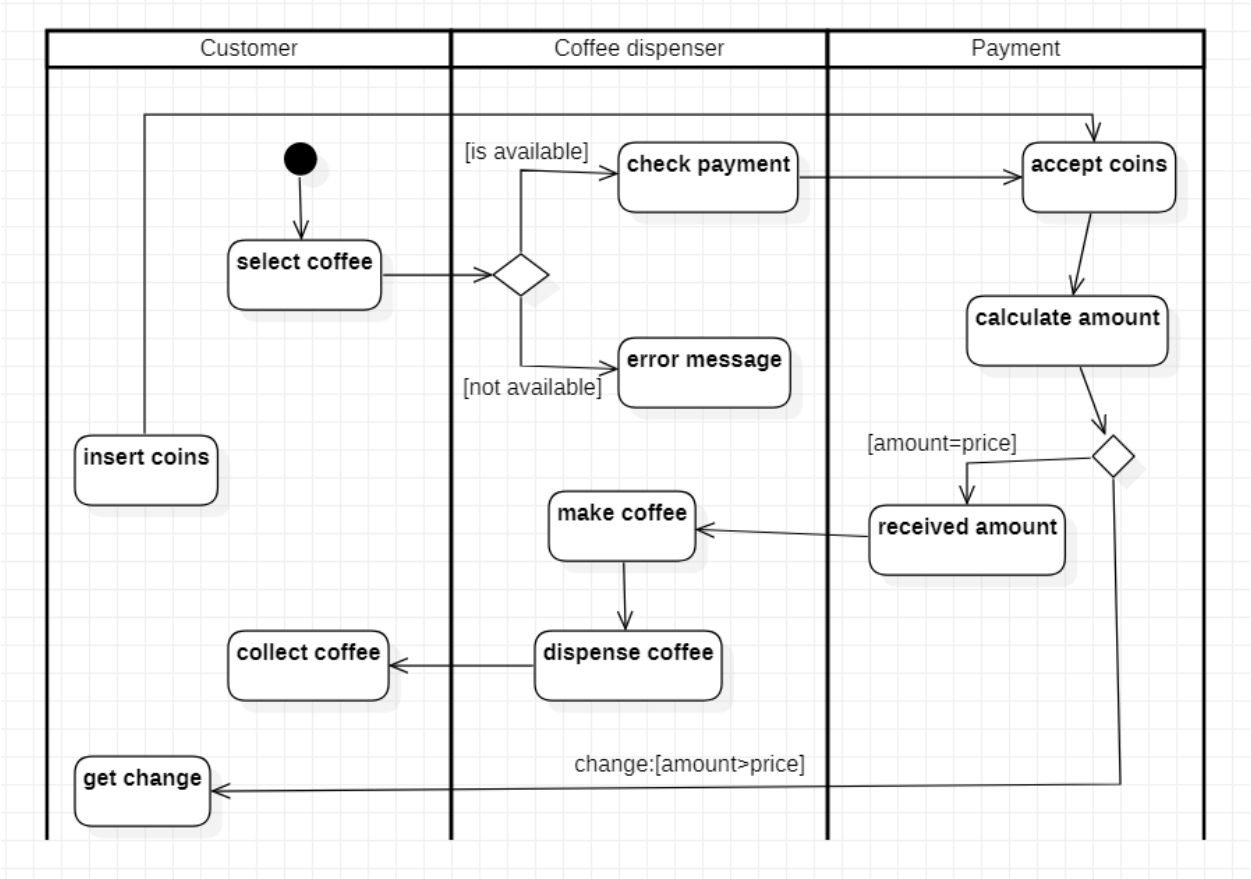
usecasediag:



Sequence Diag:



activity diag:



Lab5 Online Shopping System

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

Software Requirements:

The software provides the following facilities to the customers:

• Facilitates easy shopping online anywhere with free shipping (conditions apply).

• Provides information about the products in categories

• Can avail the facility of purchasing second hand products

• Can reserve if the particular product is not available

• Customers are provided with up to date information on the products available

• Provides email facility for future correspondence

• Provides backup facility

• Can add nearly ten products to their shopping cart at a time

The software will not provide the following facilities to the customers:

• Cannot reserve the product for more than two days.

• Cannot reserve more than two products • Responsibility of damages

• The product cannot be changeable once confirmed

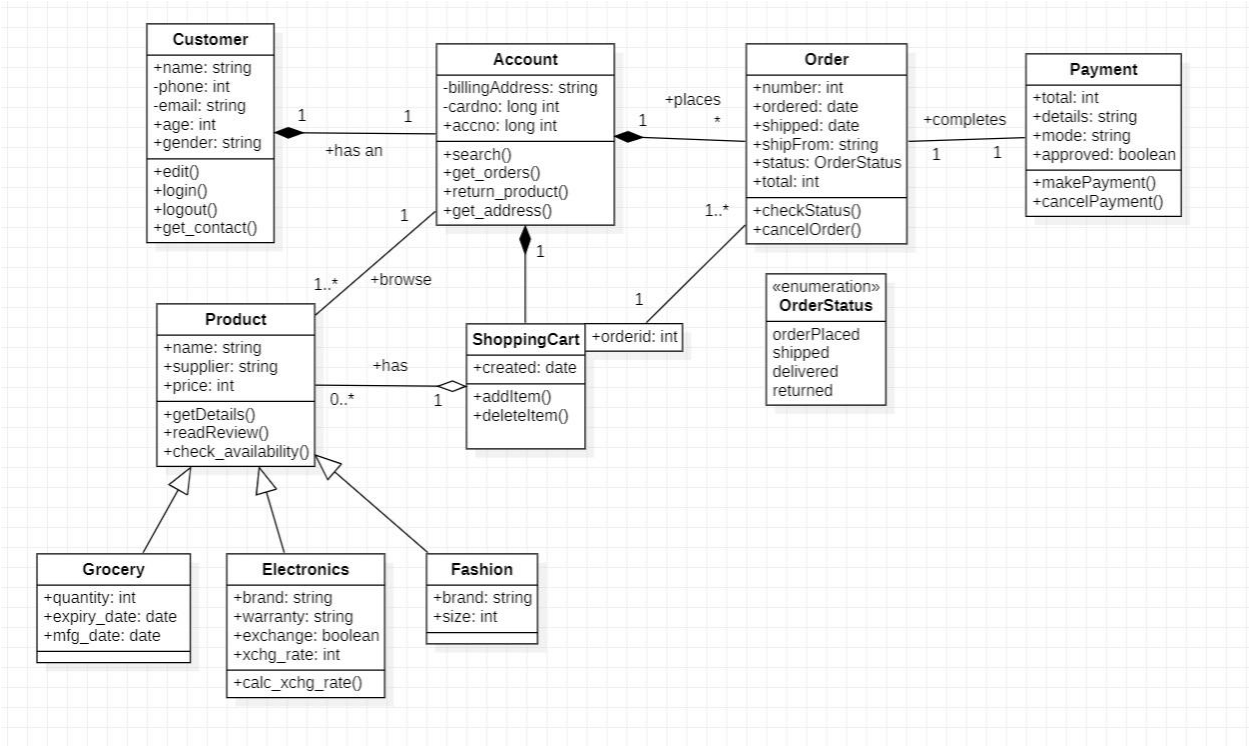
The software provides the following facilities to the merchants:

• Facilitates easy bidding facility

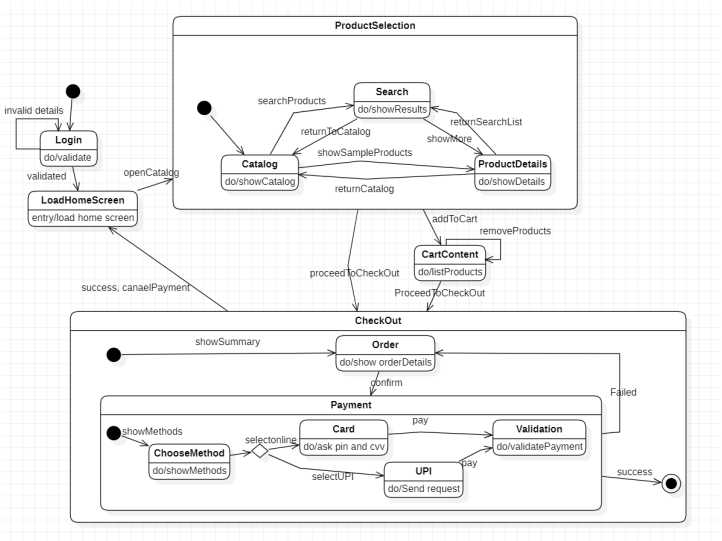
• Provides complete information about the customers

- Provides complete information about their products
- Can avail the facility of email correspondence
- Can avail the brand catalog facility

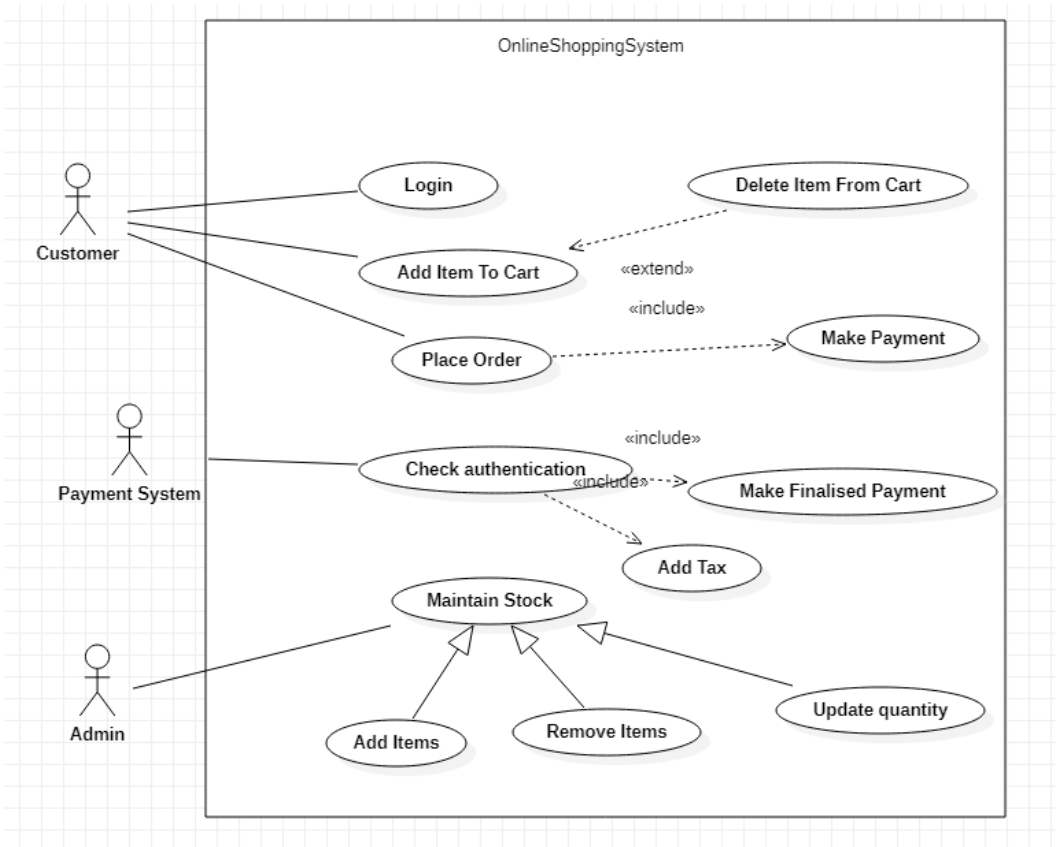
Class diag :



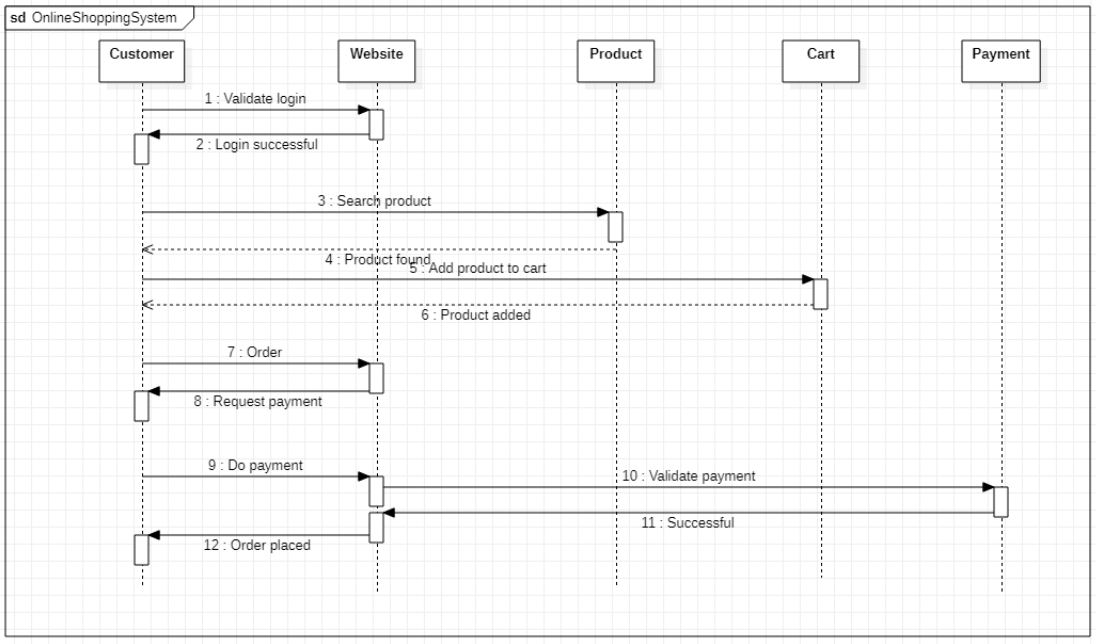
State Diag:



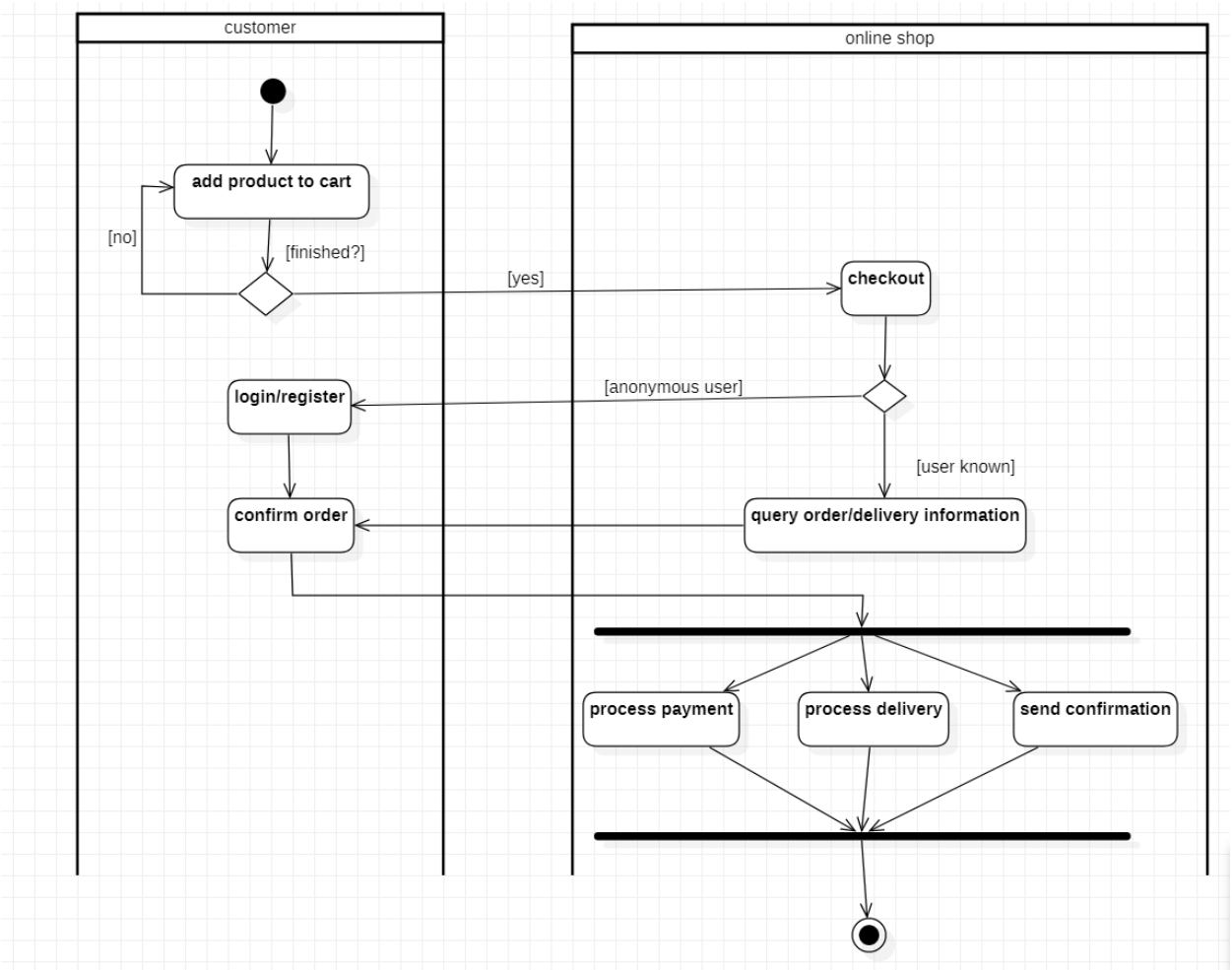
use case:



Sequence Diag:



Activity diag:



Lab6: Railway Reservation System

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

Software Requirements: Railway reservation system project 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 between any two stations served by a train. It also provides details about

1. Timetable

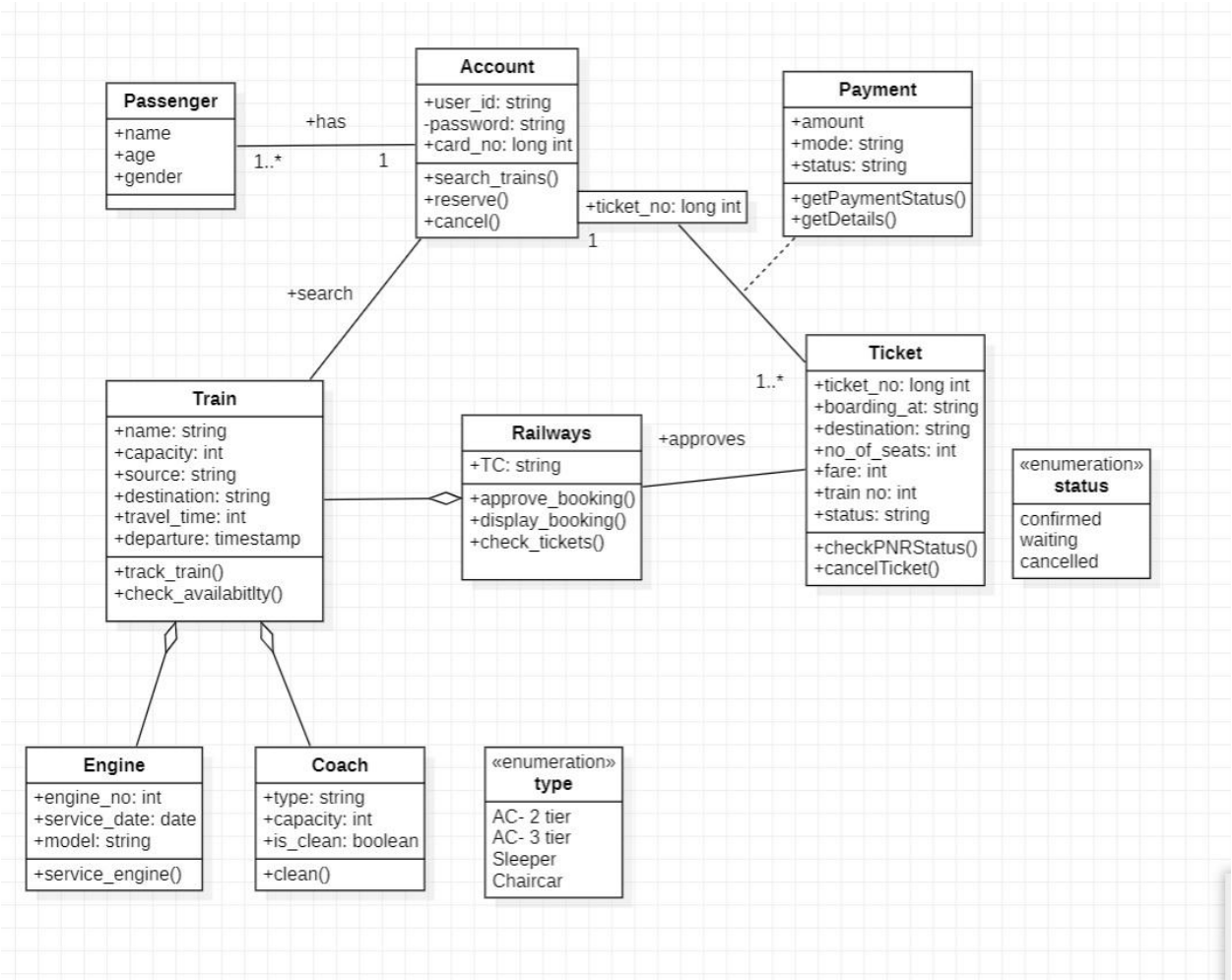
2. Train Fares

3. Current status of reservation position

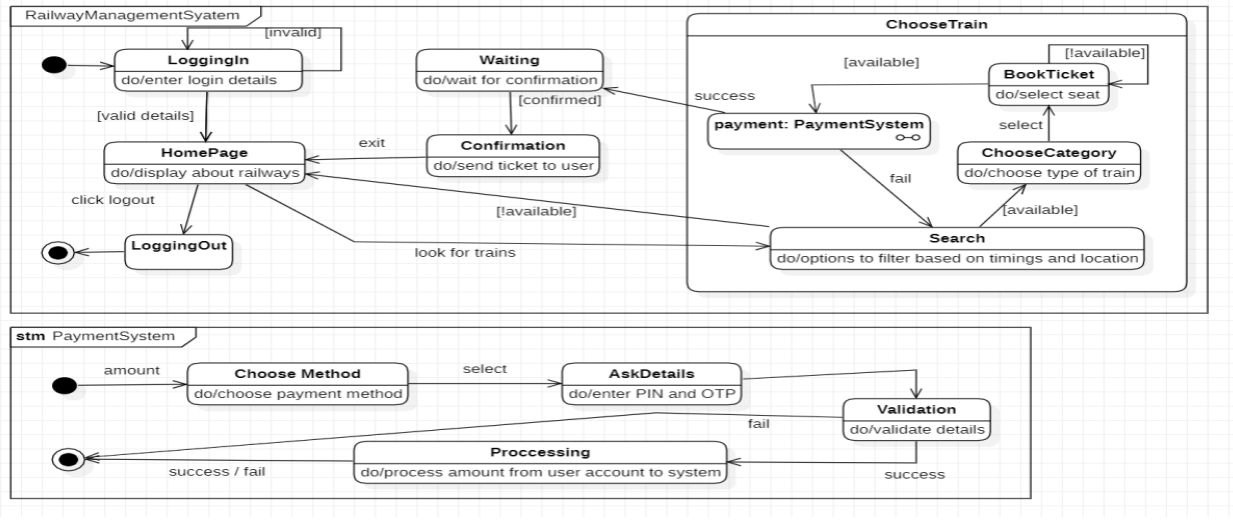
4. Train available between a pair of stations

5. Accommodation available for a train/date combination Types of tickets: General and Tatkaal

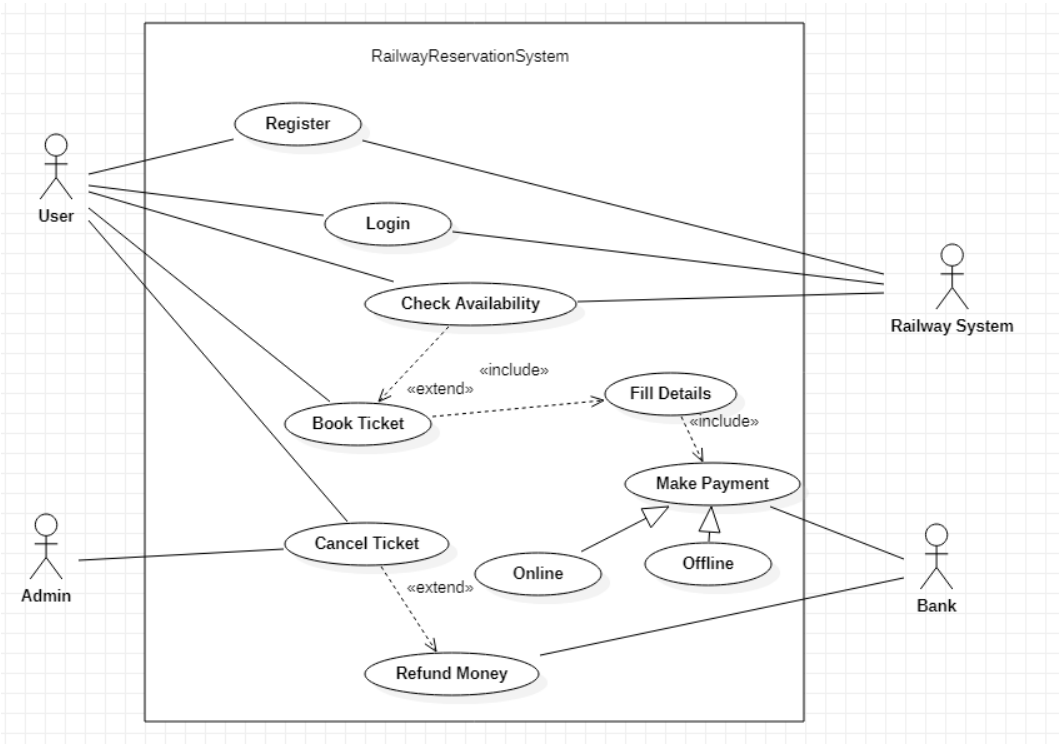
Class Diag:



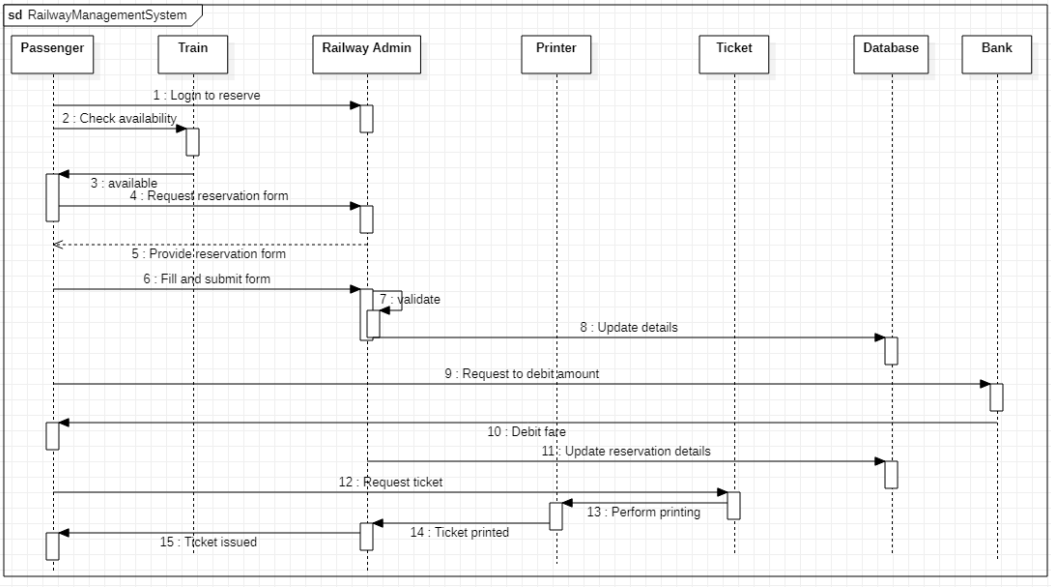
State Diag:



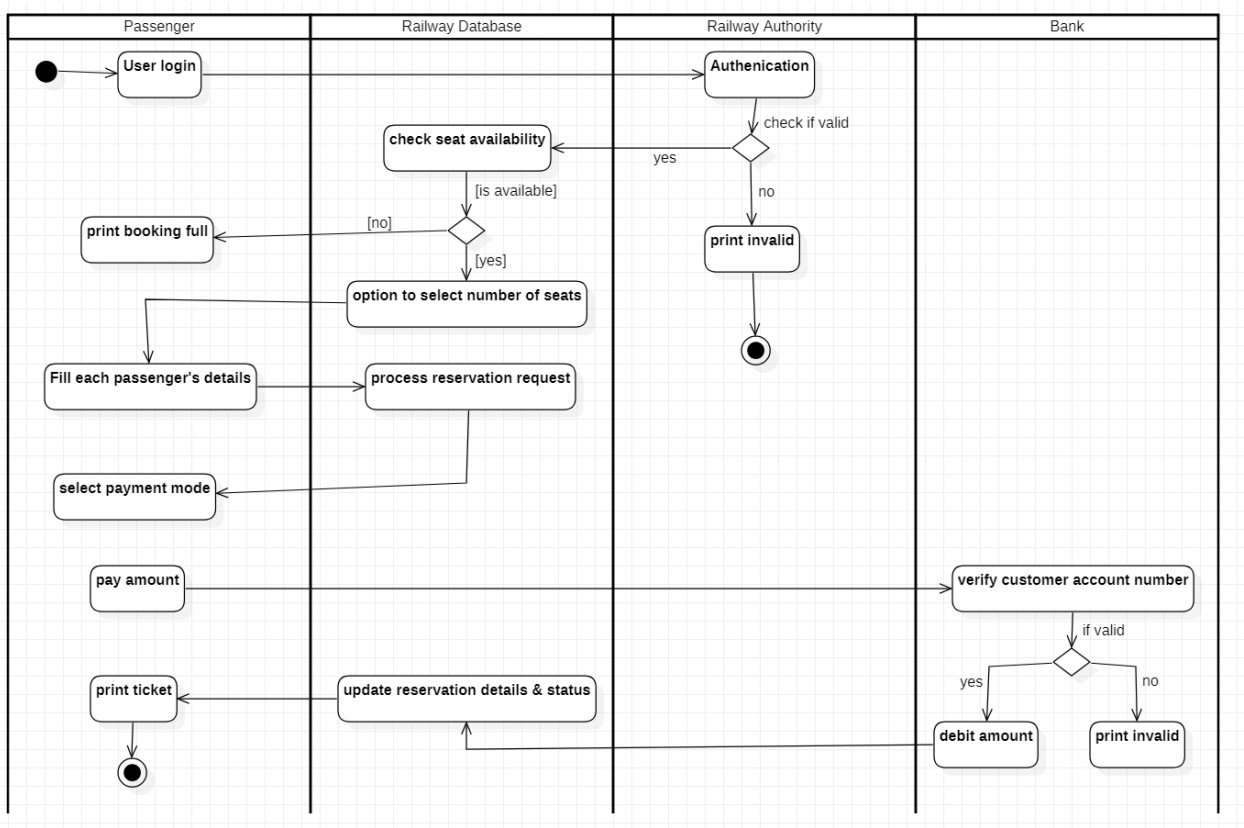
Use case :



Sequence Diag:



Activity diag:



Lab 7 Graphical Editor

Problem Statement: 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, and the programmer can create a graphical editor and palette of shapes in order to modify an underlying model. The graphical editor provides an interface with which the programmer implements said editor for a given underlying model. Such an instance of the graphical editor allows a user to drag objects from a specified model into a working graphical diagram

. Software Requirements:

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 with options for New, Open, Save, toolbox and Text Toolbox.

• One integrated view to users for toolbar, color box, menu, and graphic screen.

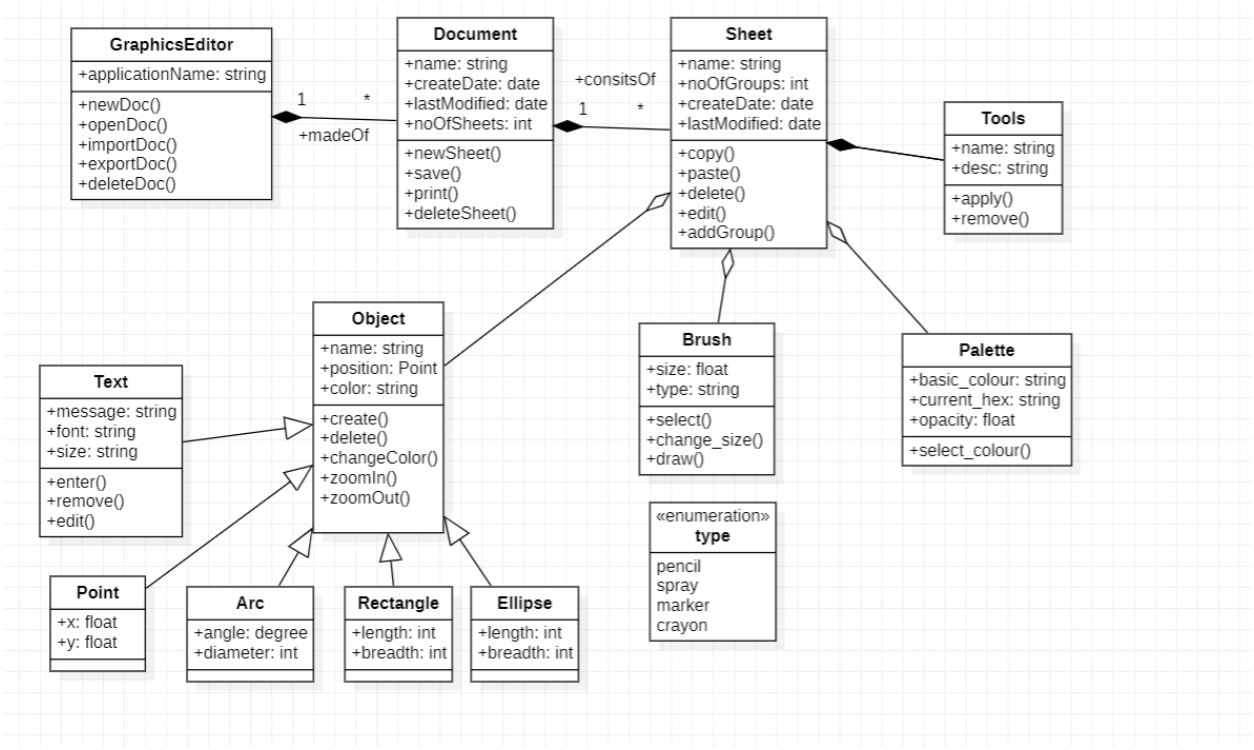
• Easy handling of tools for users.

• Ability to group several drawings into one i.e. complex drawing

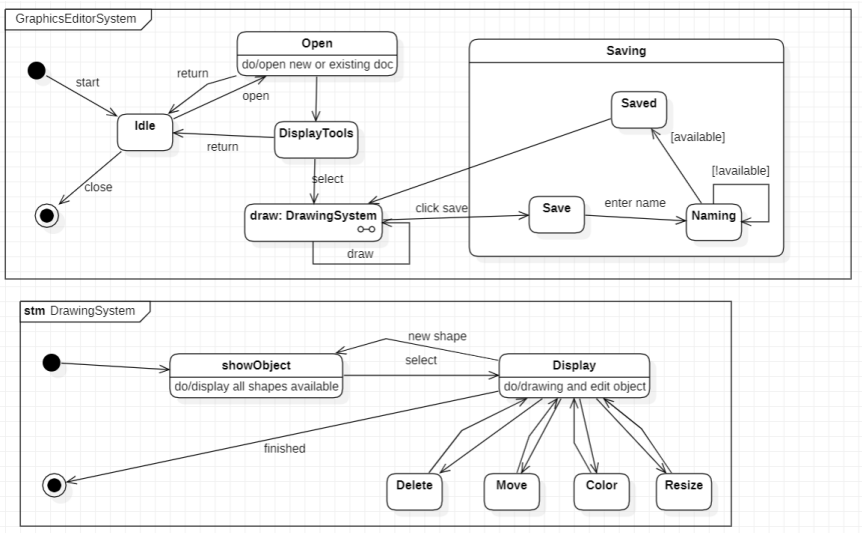
. • Provision of zoom in and zoom out.

• Different shadings of line tool are provide

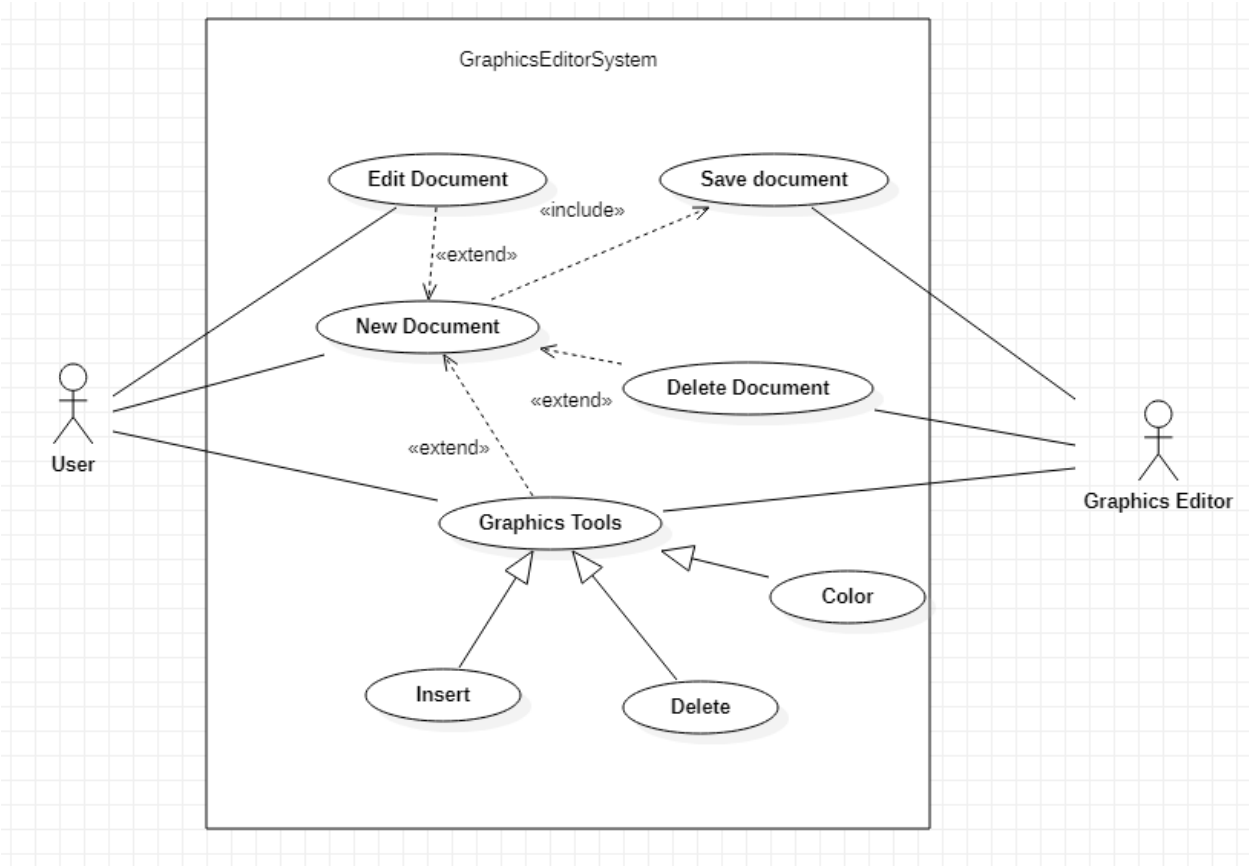
Class diag:



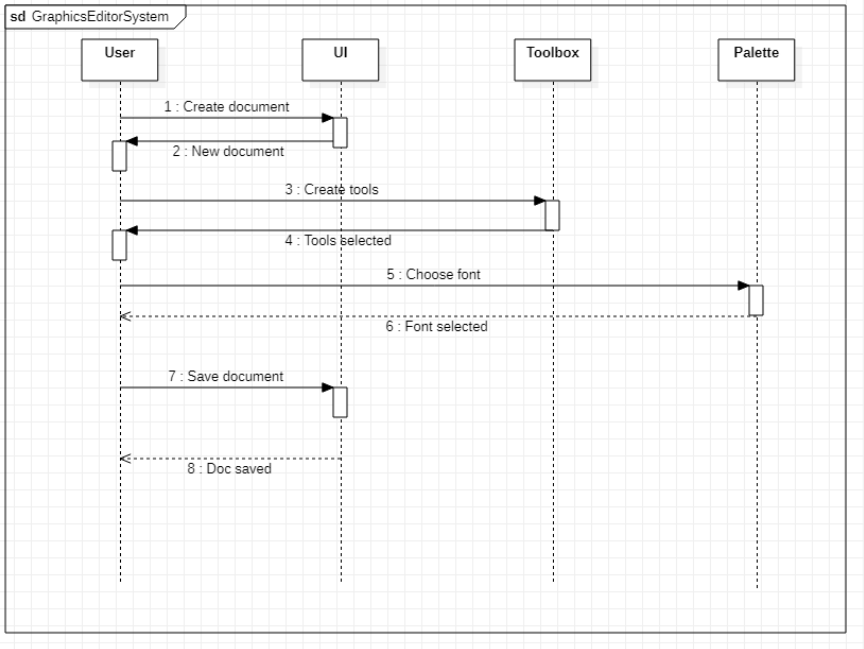
State diag



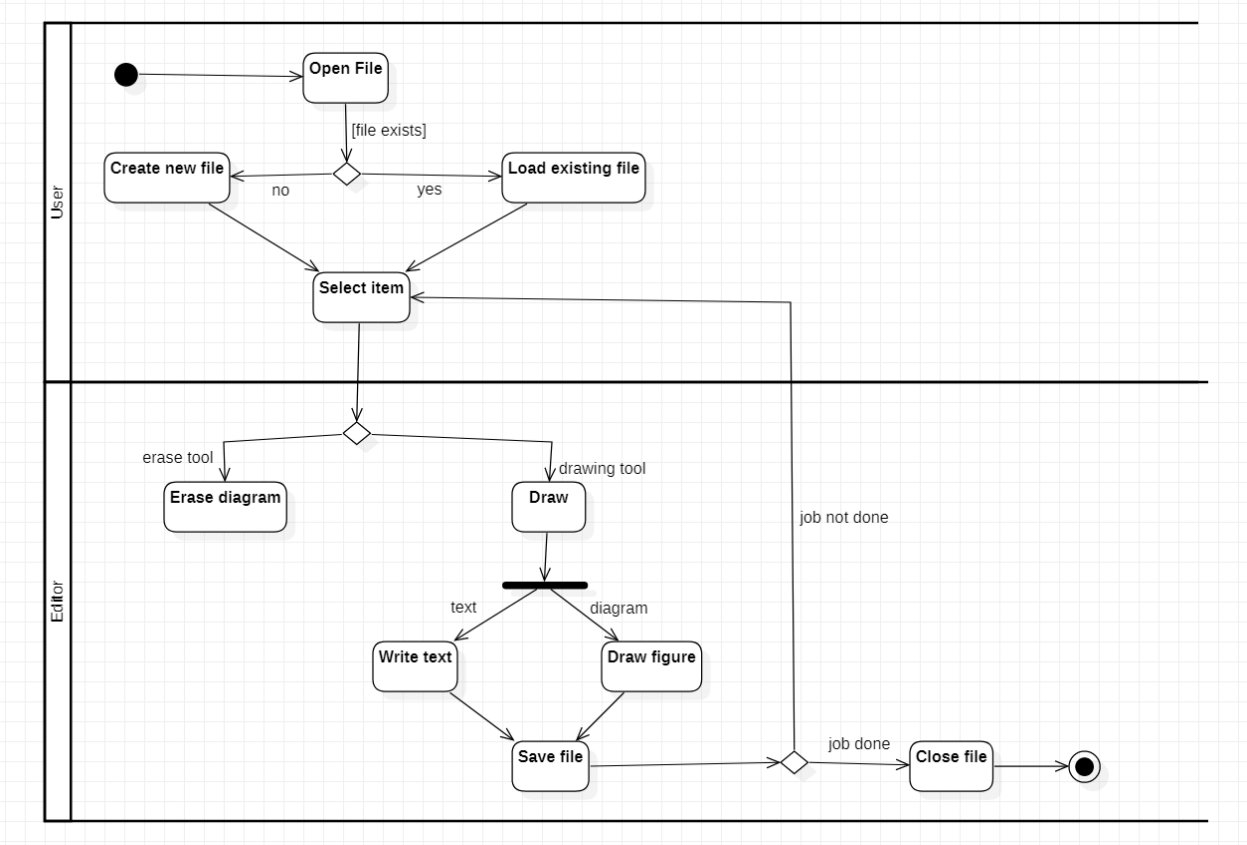
Use case:



Sequence:

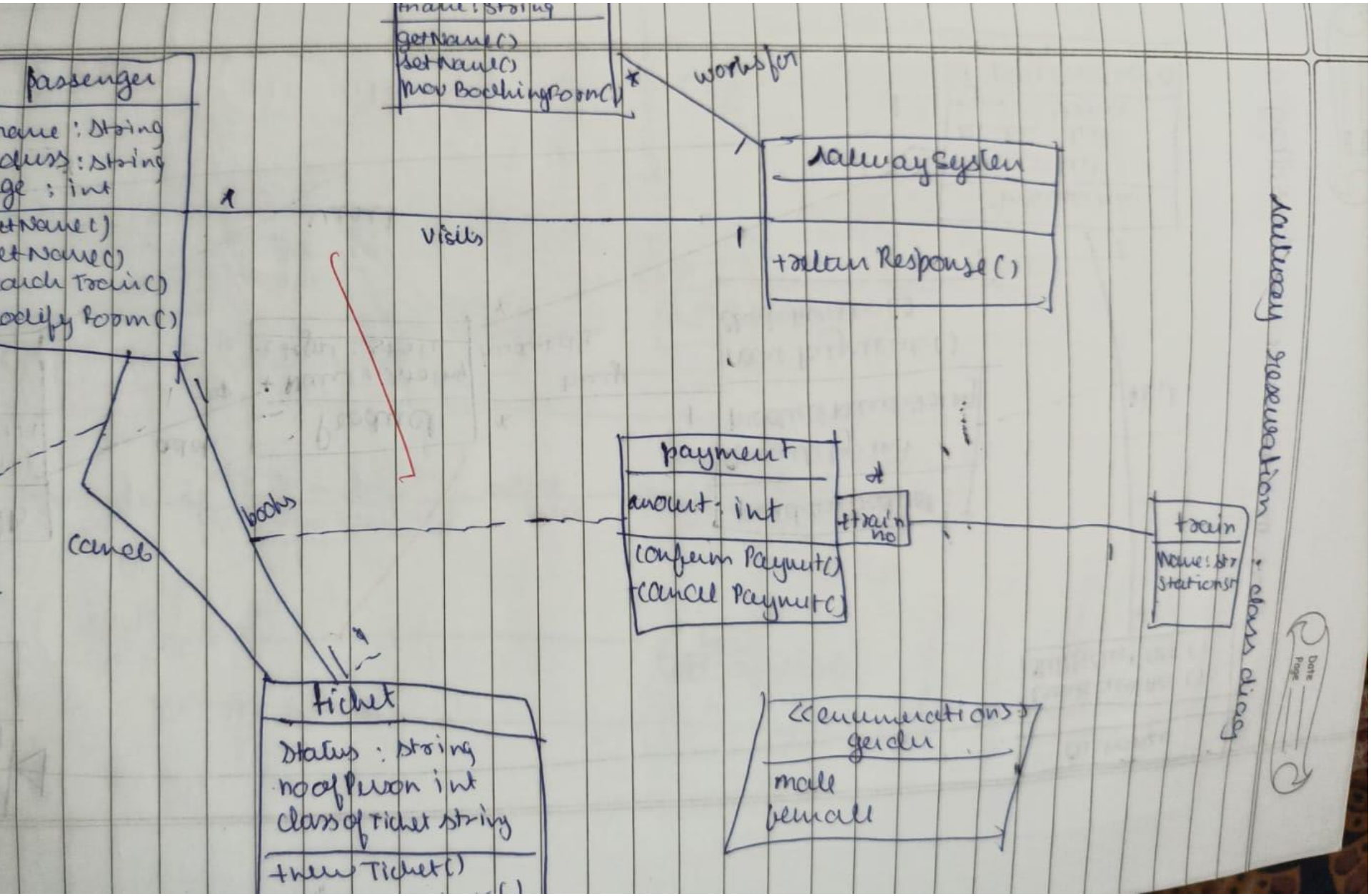


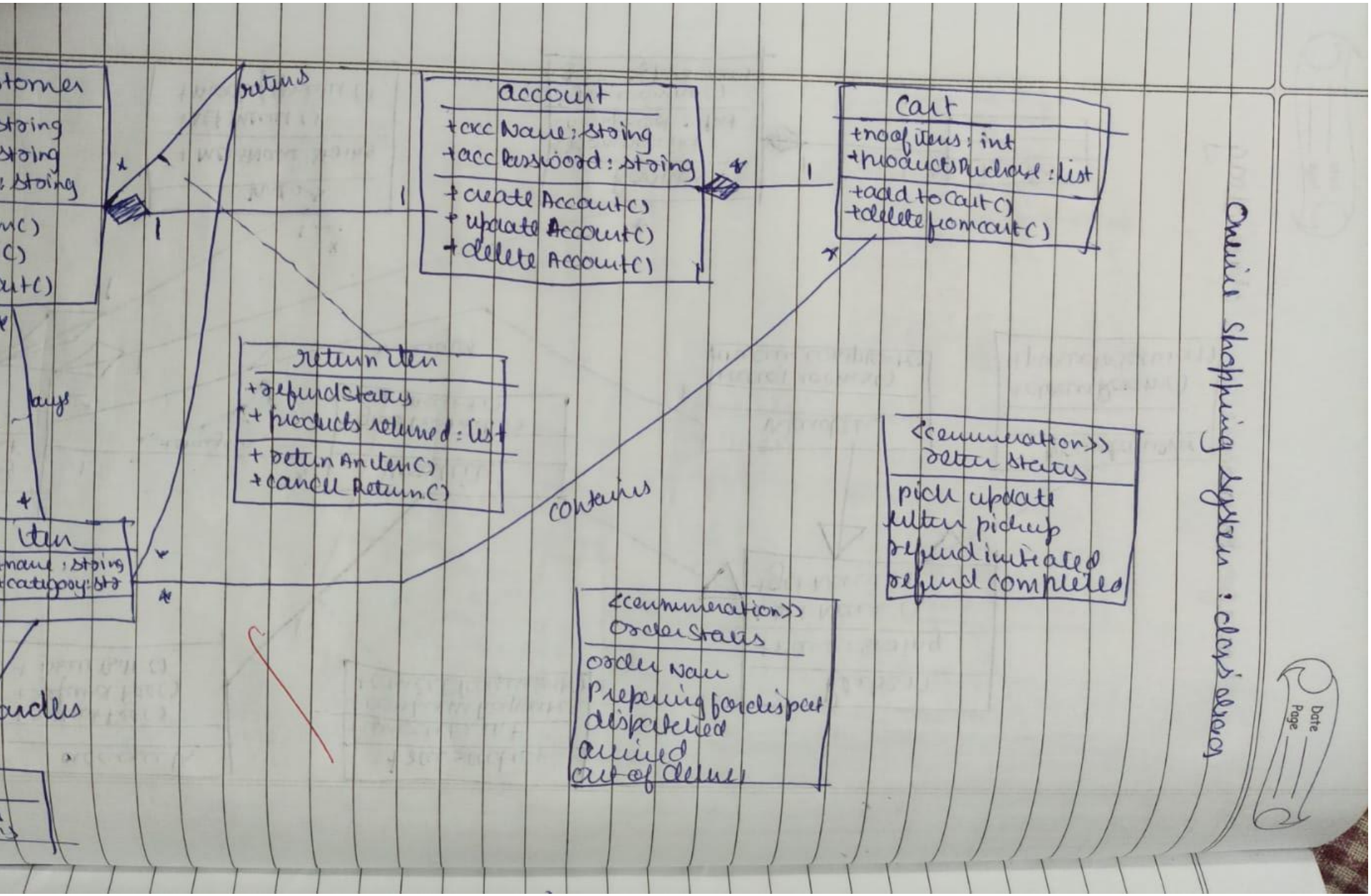
Activity:

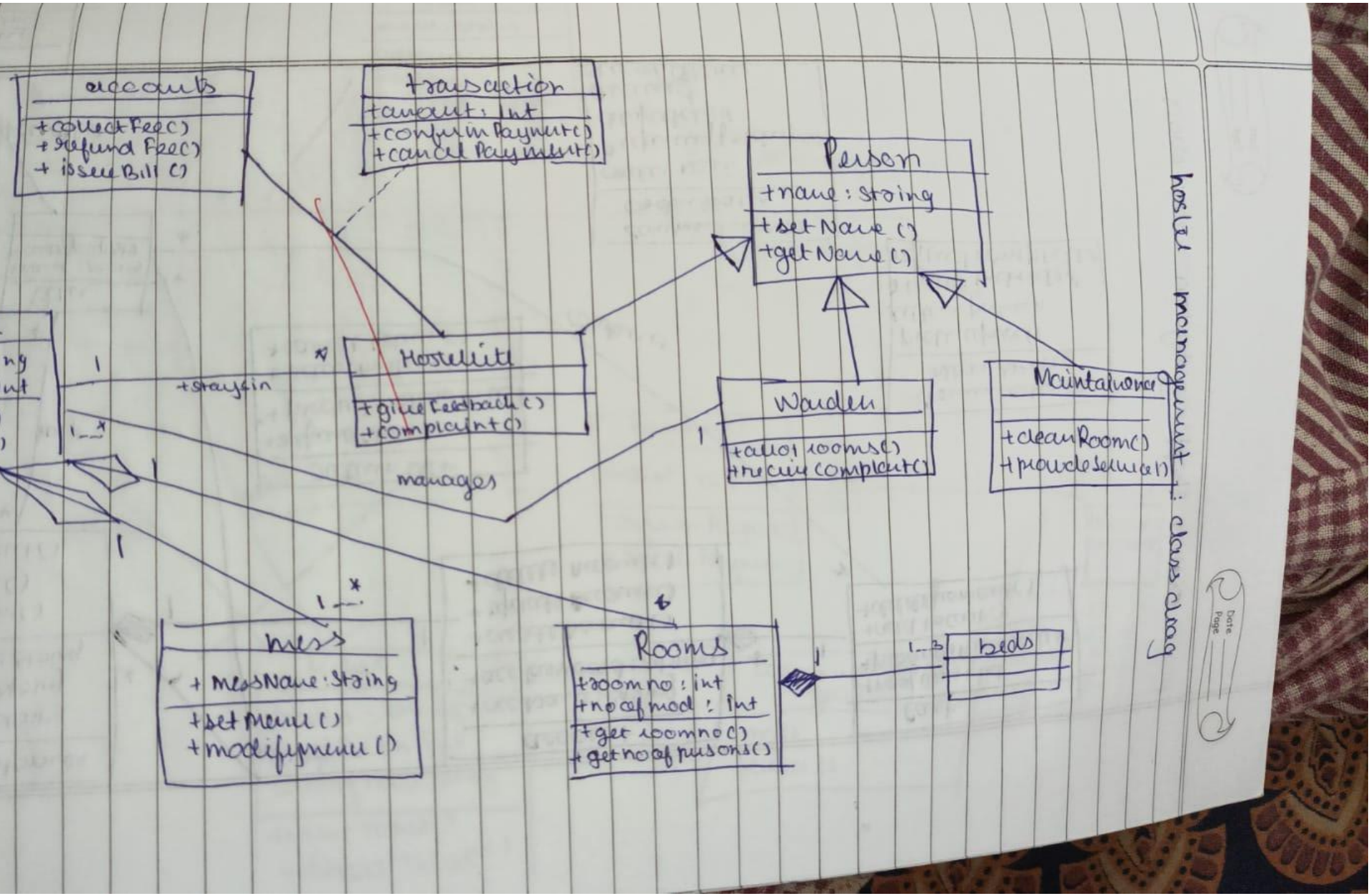

User
Editor
Open File
[file exists]
Create new file
no
yes
Load existing file
Select item
erase tool
drawing tool
Erase diagram
Draw
job not done
text
diagram
Write text
Draw figure
Save file
job done
Close file

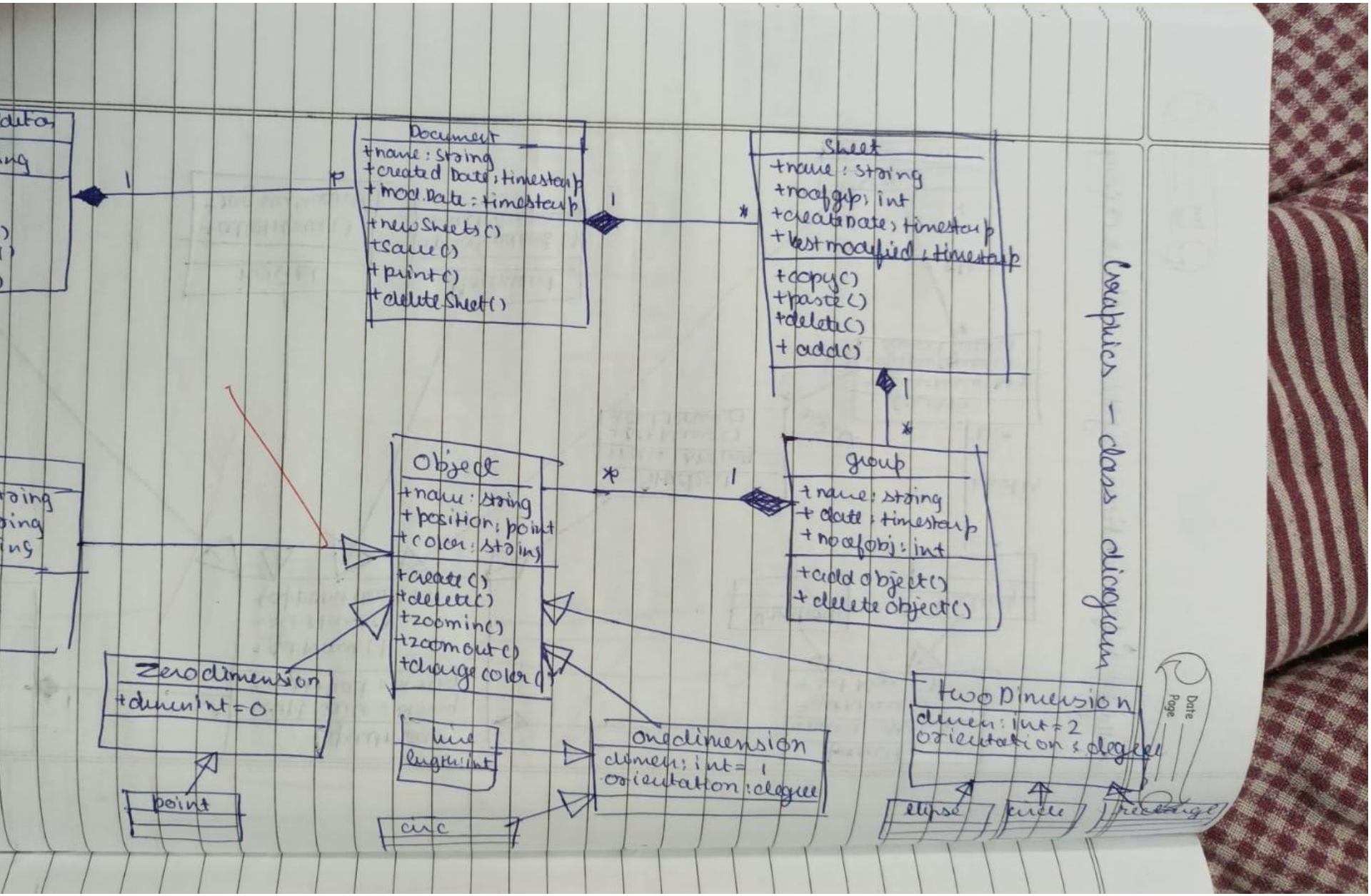
Written class diags

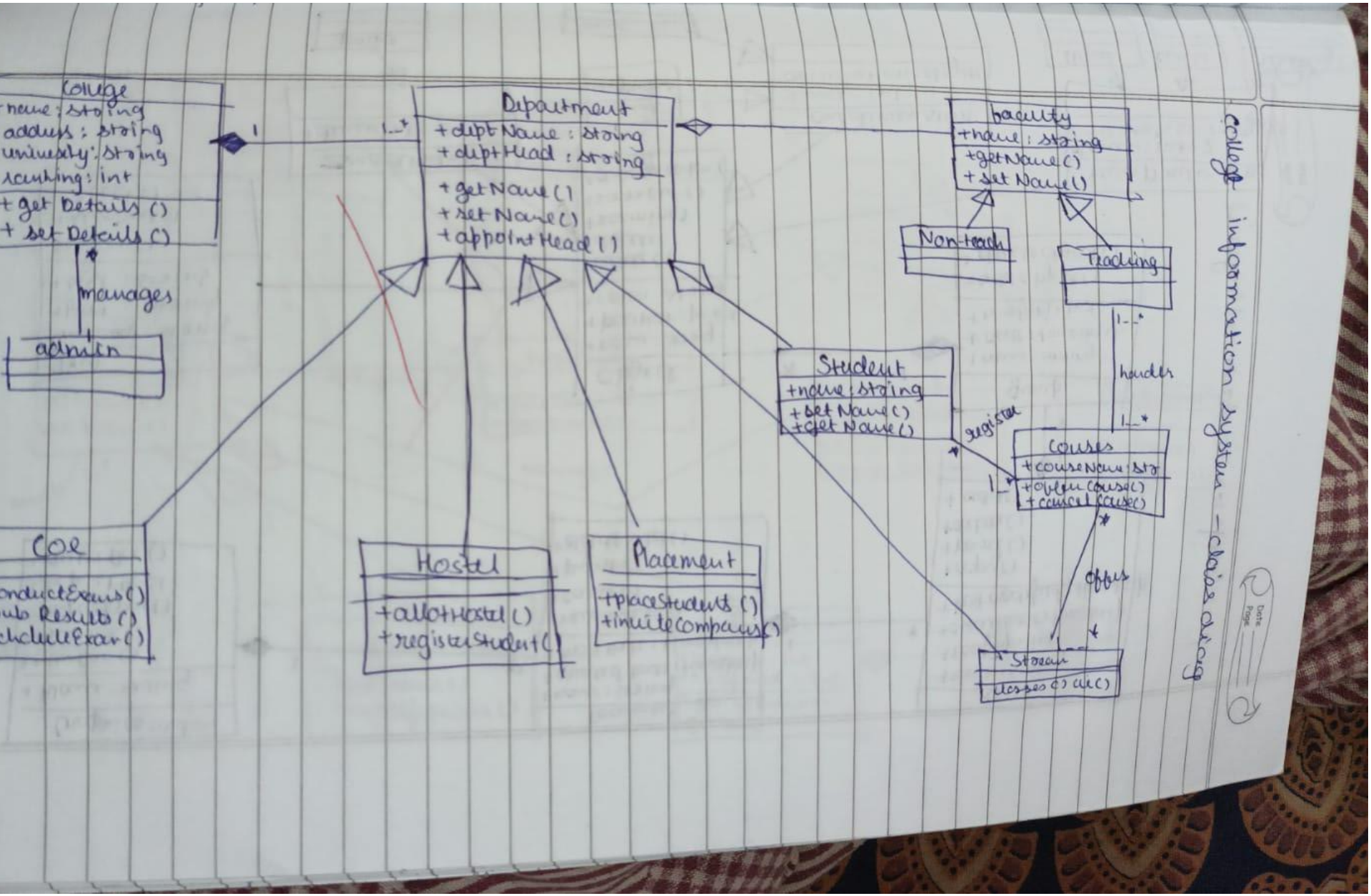
Stock maintainence - class diag


getName()
setName()
makeBookingRoom()
works for
Passenger
name : String
class : String
age : int
etName()
etName()
ancelTicket()
odifyRoom()
Visits
RailwaySystem
+returnResponse()
payment
amount : int
confirmPayment()
cancelPayment()
train no
train
name: Str
stations
books
cancel
Ticket
status : string
noofPerson int
classofTicket string
newTicket()
<<enumerations>>
gender
male
female
Railway reservation : class diag

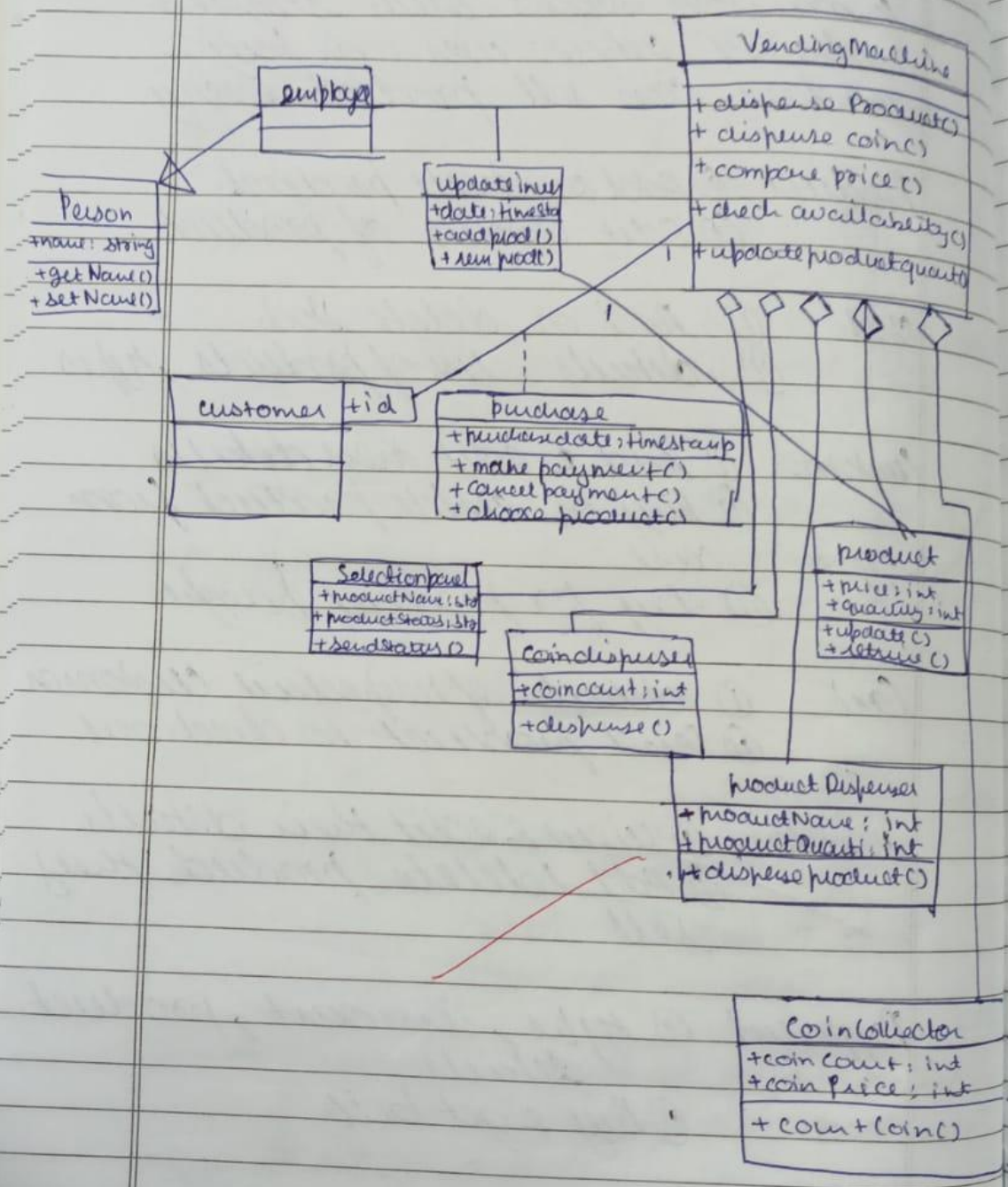

Online shopping system : class diagram
account
+acc Name : string
+acc password : string
+ create Account()
+ update Account()
+ delete Account()
Cart
+no of items : int
+products Purchased : list
+add to Cart()
+delete from cart()
return item
+refund status
+ products returned : list
+ return An item()
+ cancel Return()
<<enumeration>>
order status
order Nan
Preparing for dispatch
dispatched
arrived
out of deliv
<<enumeration>>
refund status
pick update
after pickup
refund initiated
refund completed
returns
contains
ays
Date
Page


accounts
+collectFee()
+refundFee()
+issueBill()
transaction
+amount: int
+confirmPayment()
+cancelPayment()
Person
+name: string
+setName()
+getName()
Hostelite
+giveFeedback()
+complaint()
Warden
+allotRooms()
+recieveComplaint()
Maintainance
+cleanRoom()
+provideService()
mess
+messName: string
+setMenu()
+modifyMenu()
Rooms
+roomno: int
+noofmod: int
+getroomno()
+getnoofpersons()
beds
+straysin
manages
1
*
hostel management : class diag
Date
Page


Graphics - class diagram
Document
+name : string
+created Date : timestamp
+modi. Date : timestamp
+newSheets()
+save()
+print()
+deleteSheet()
Sheet
+name : string
+noofgrp : int
+createDate : timestamp
+lastmodified : timestamp
+copy()
+paste()
+delete()
+add()
group
+name : string
+date : timestamp
+noofobj : int
+addobject()
+deleteobject()
Object
+name : string
+position : point
+color : string
+create()
+delete()
+zoomin()
+zoomout()
+changecolor()
zerodimension
+dimen int = 0
point
line
length : int
circ
onedimension
dimen : int = 1
orientation : degree
twoDimension
dimen : int = 2
orientation : degree
ellipse
circle
rectangle
Date
Page

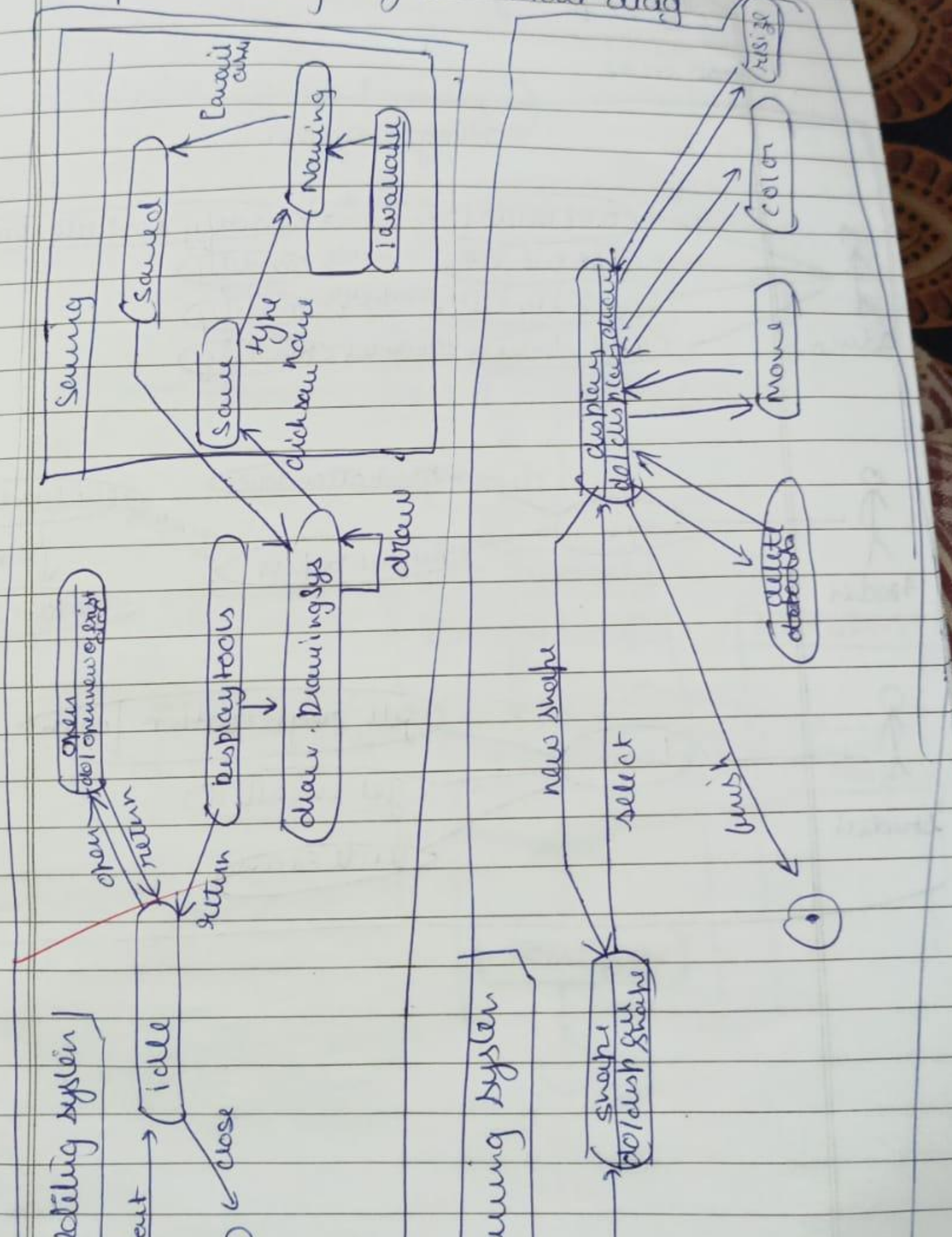

College
name : string
address : string
university : string
ranking : int
+ get Details ()
+ set Details ()
manages
admin
Department
+ dept Name : string
+ dept Head : string
+ get Name ()
+ set Name ()
+ appoint Head ()
faculty
+ name : string
+ get Name ()
+ set Name ()
Non-teach
Teaching
handles
Student
+ name : string
+ set Name ()
+ get Name ()
register
Courses
+ courseName : str
+ open Course ()
+ cancel Course ()
offers
Stream
COE
conductExams ()
Results ()
Hostel
+ allotHostel ()
+ registerStudent ()
Placement
+ placeStudents ()
+ inviteCompanies ()
college information system - class diag

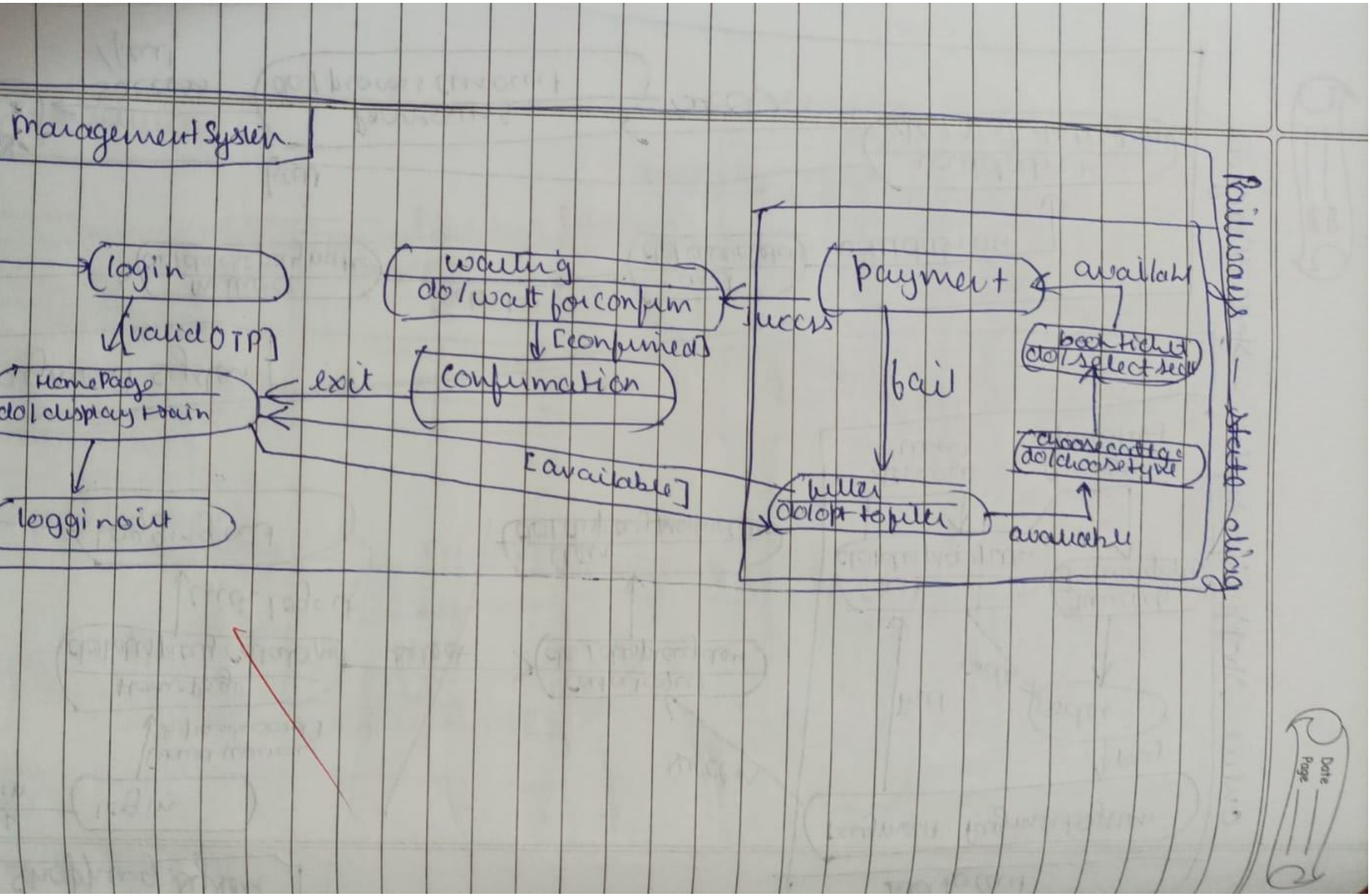
coffee vending machine - class diagram

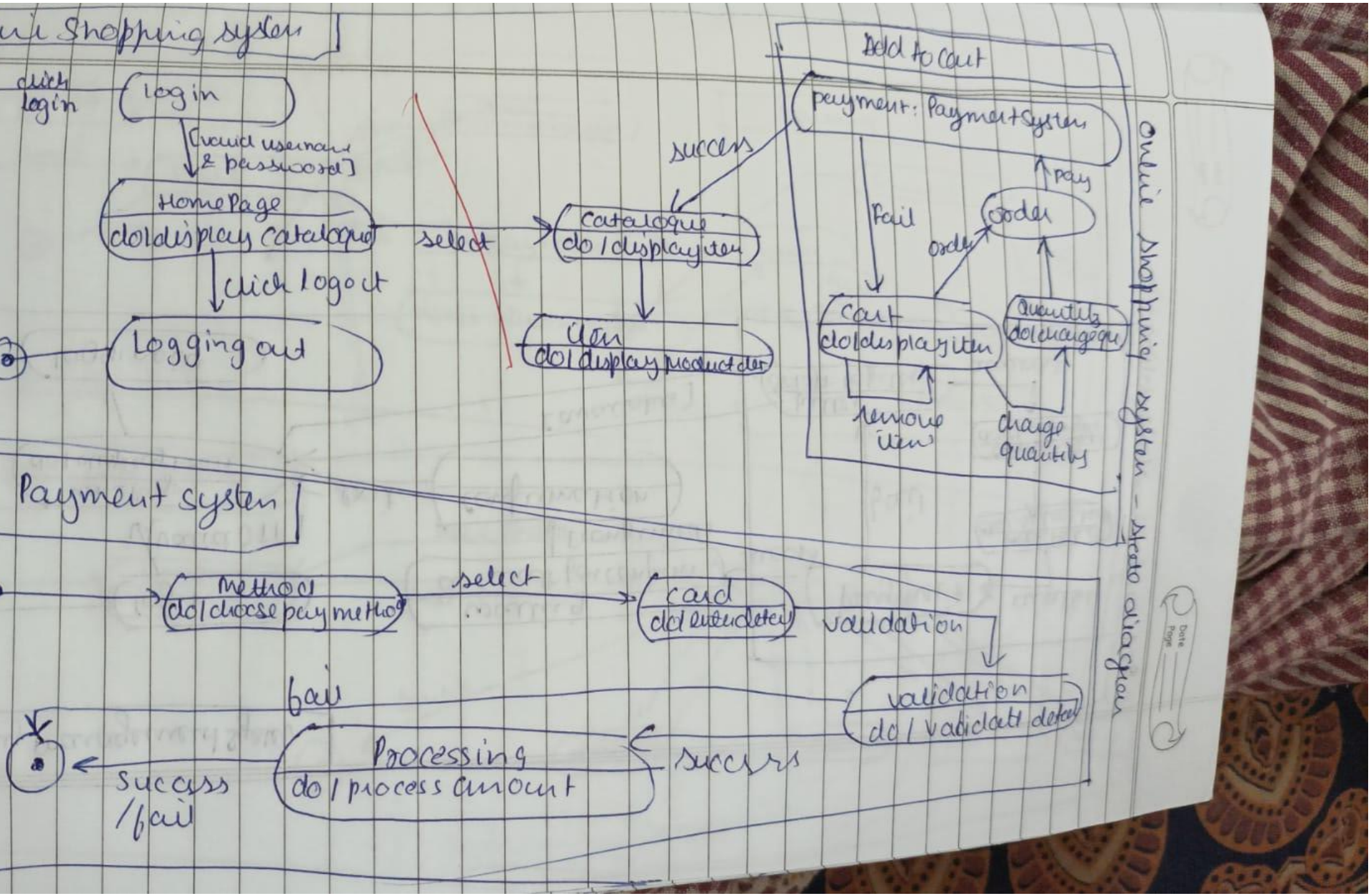


Written state diags

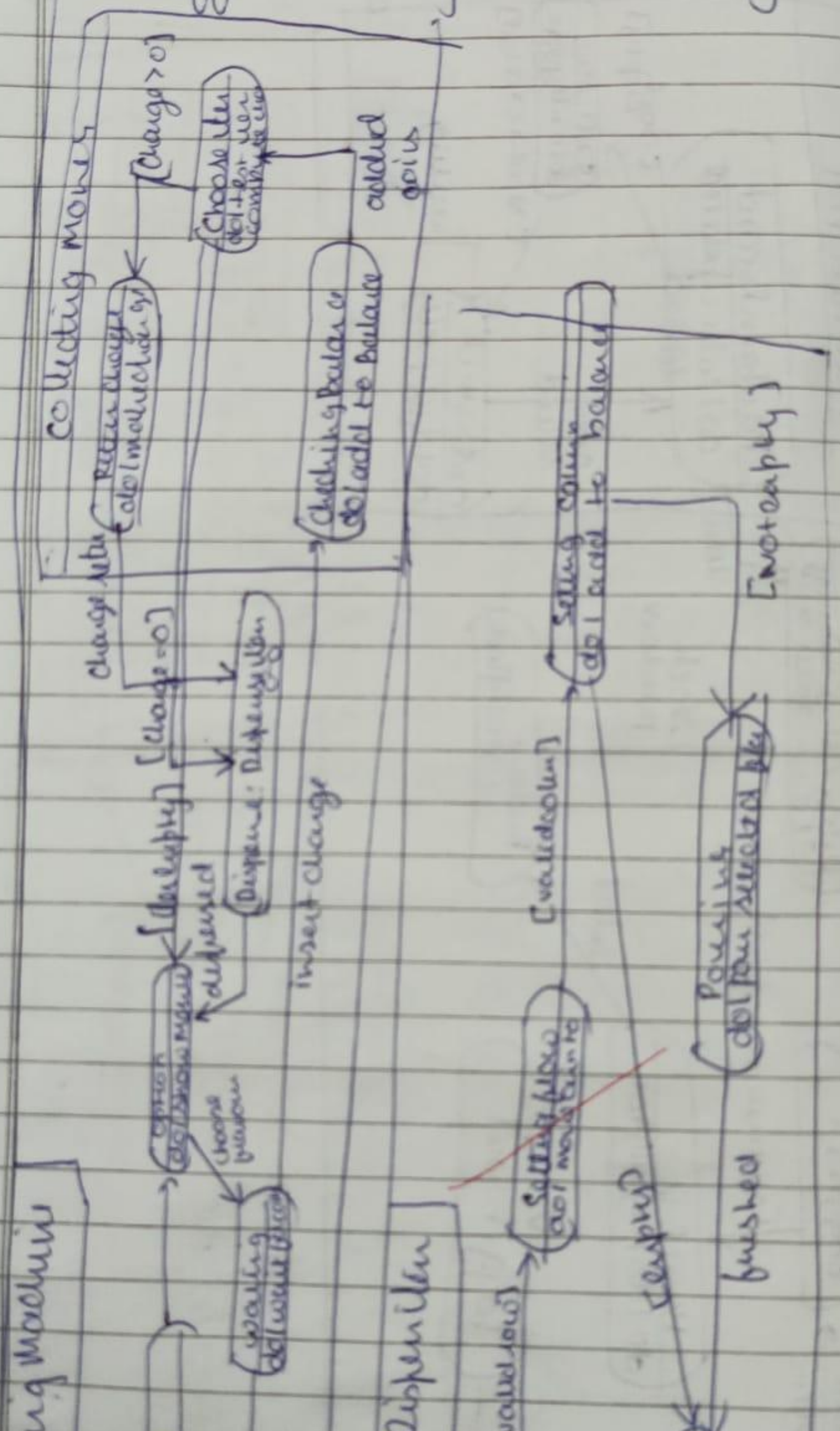
Graphics editing system - State diag

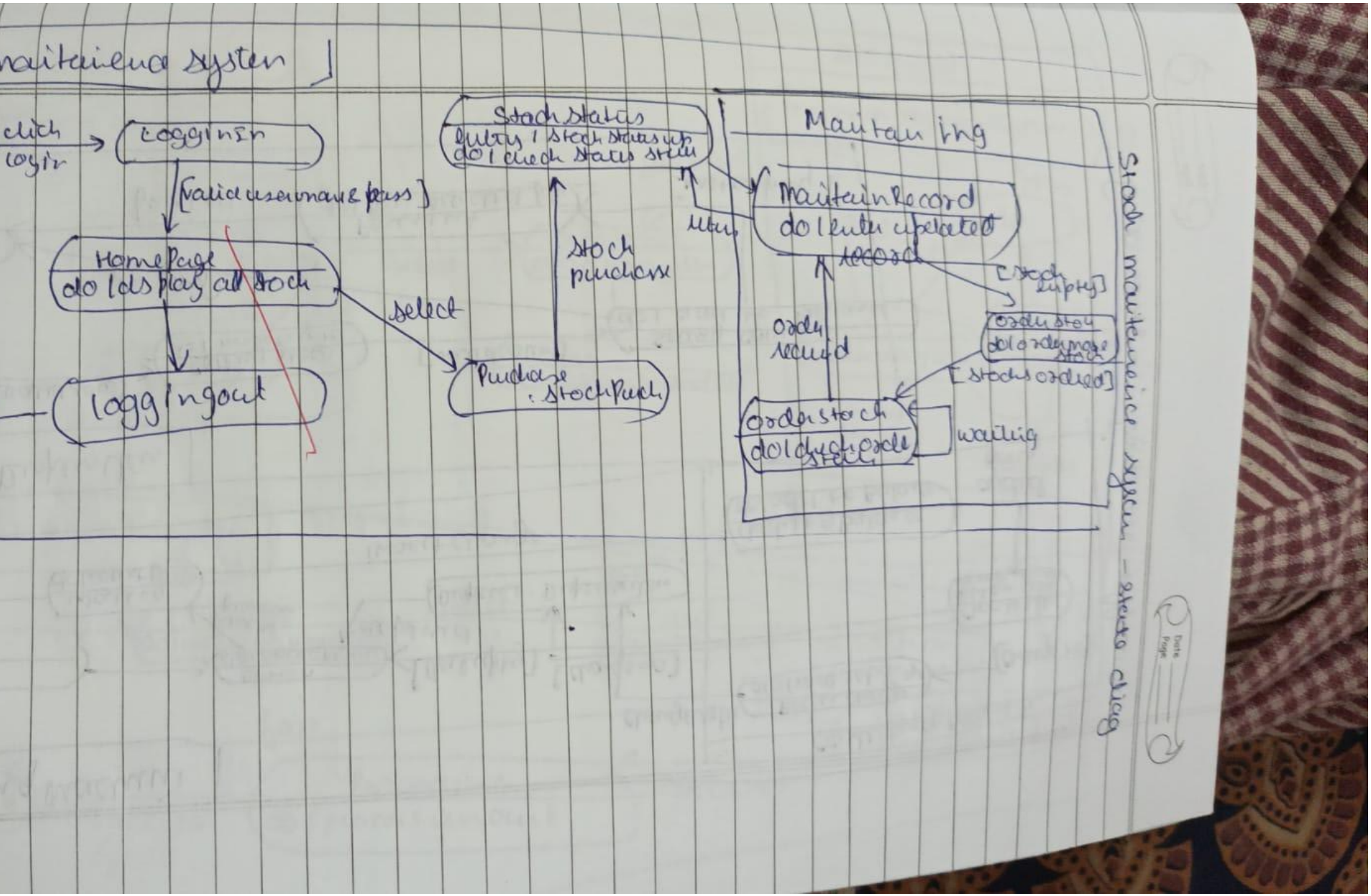


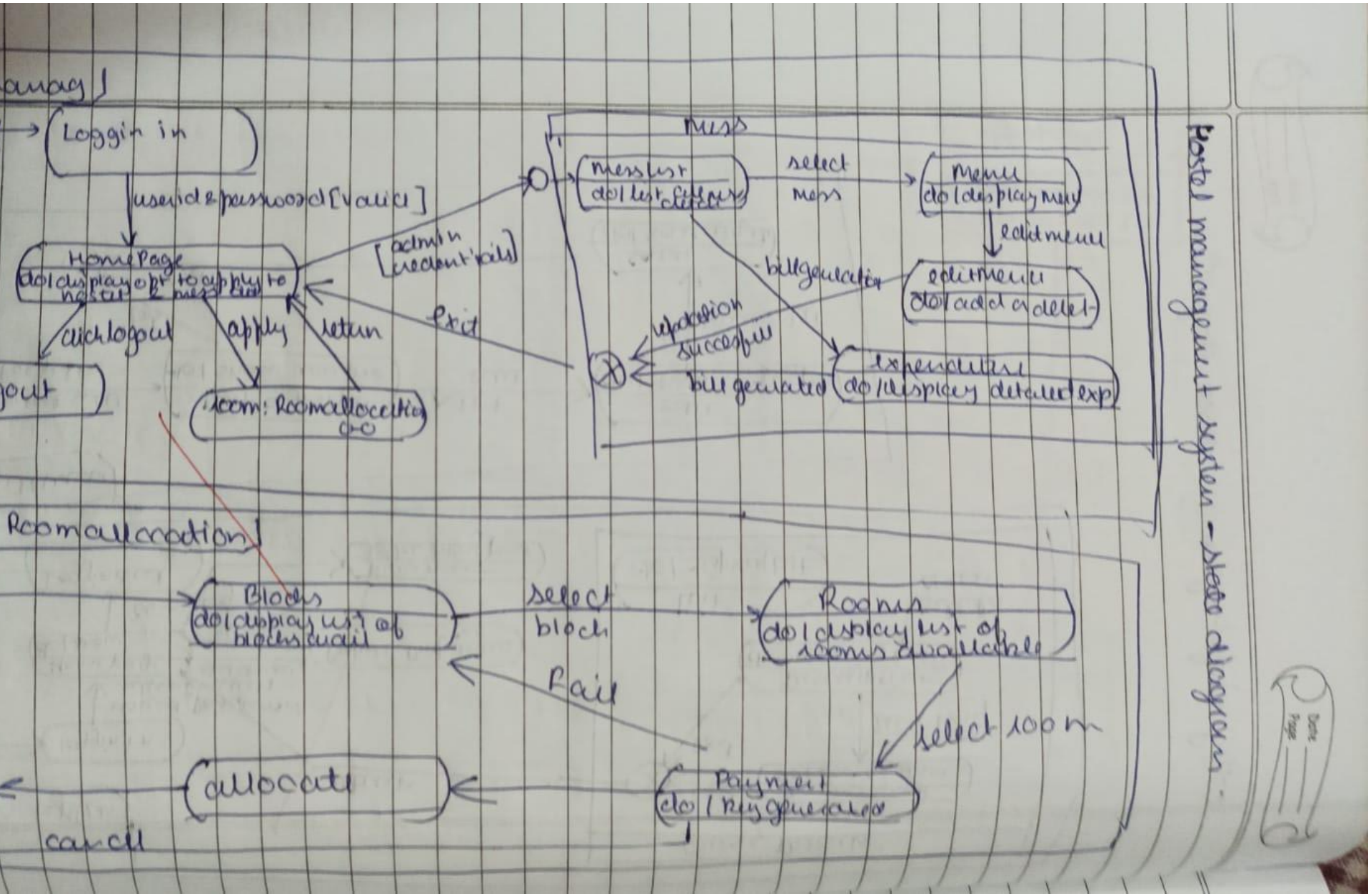

Management System
login
[valid OTP]
HomePage
do/ display Main
loggin out
waiting
do/ wait for confirm
[confirmed]
exit
confirmation
[available]
payment
success
fail
availabl
book ticket
do/ select seat
choose coach
do/ choose type
seller
do/ opt to seller
availabl
Railways - state diag

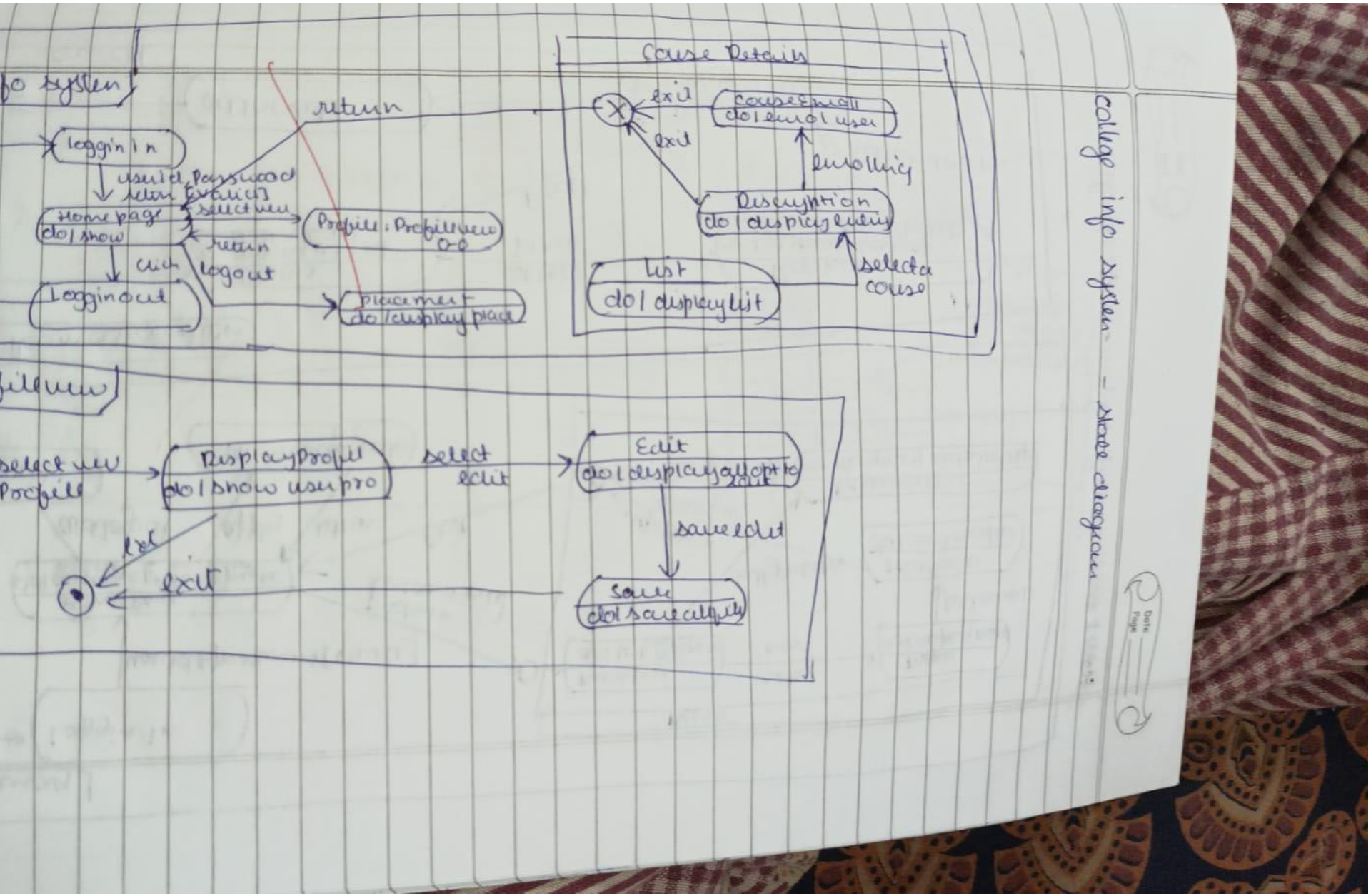

Shopping system
click login
login
[valid username & password]
HomePage
do/display catalogue
click logout
Logging out
select
catalogue
do/display item
item
do/display product det
Add to cart
payment: PaymentSystem
success
pay
order
fail
order
cart
do/display item
quantity
do/change qu
remove item
change quantity
Online shopping system - state diagram
Payment system
method
do/choose payment
select
card
do/ validation
validation
do/ validate det
fail
success
Processing
do/process amount
success
/fail

Coffee vending - State diag

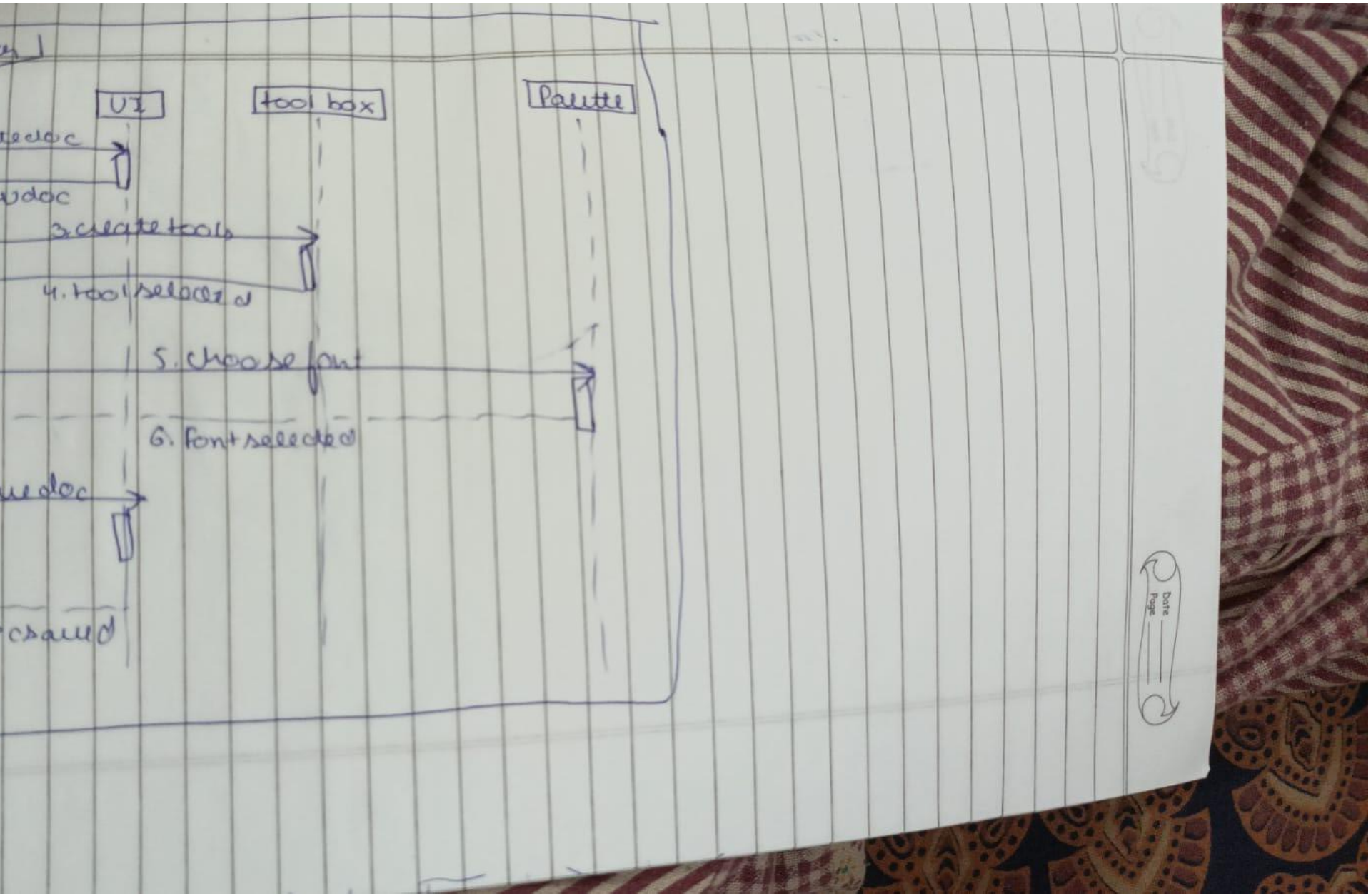


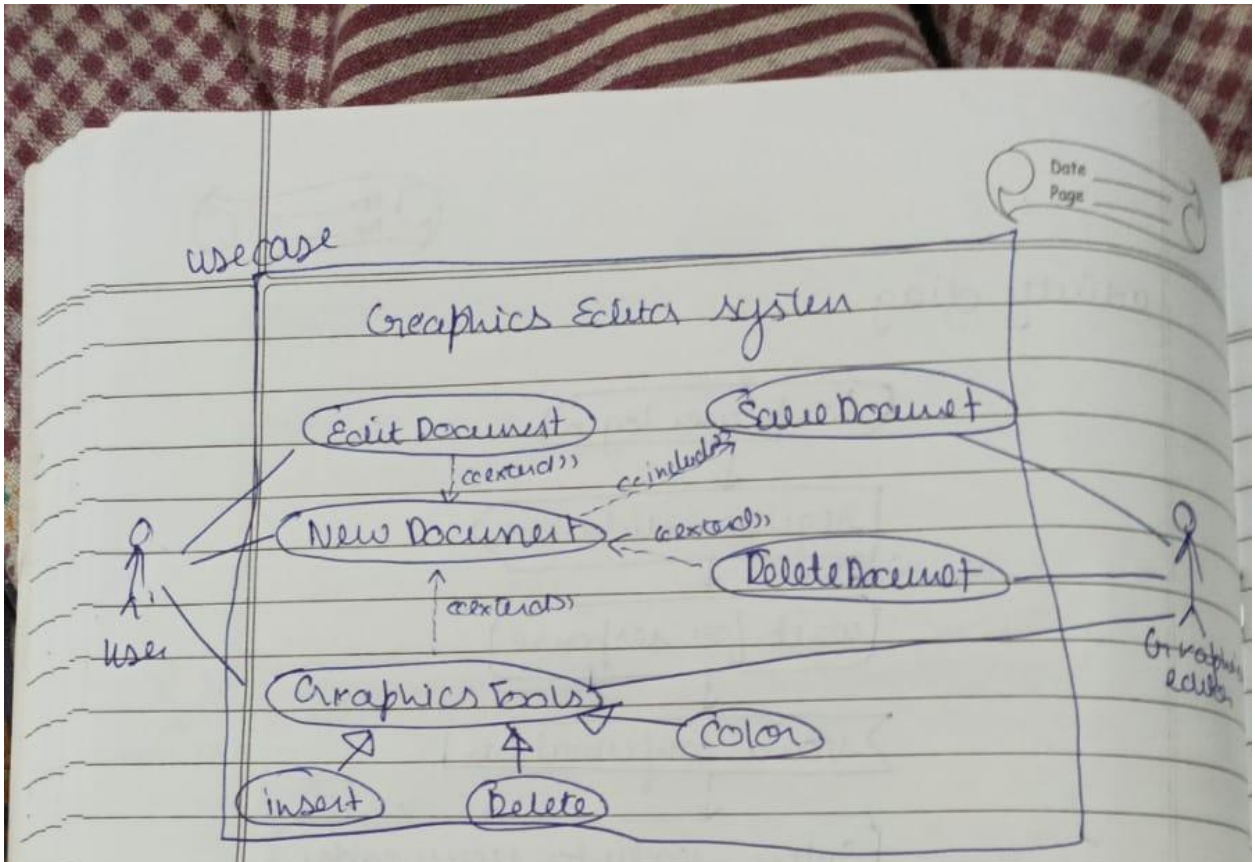

maintainence system
click login
loggingin
[valid username & pass]
HomePage
do / displayallstock
select
loggingout
Purchase
: StockPurch
stock purchase
Stock Status
query / stock status up
do / check status still
Maintaining
maintainRecord
do / auto updated record
[stock empty]
order stock
do / order more stock
[stocks ordered]
order record
orderstock
do / check order status
waiting
Stock maintenence system - state diag


Loggin in
userid & password [valid]
HomePage
[admin credentials]
click logout
apply
return
Exit
Mess
select mess
Menu
edit menu
bill generation
updation successful
bill generated
Expenditure
Room allocation
Blocks
select block
Rooms
Fail
select room
allocate
Payment
cancel
Hostel management system - state diagram

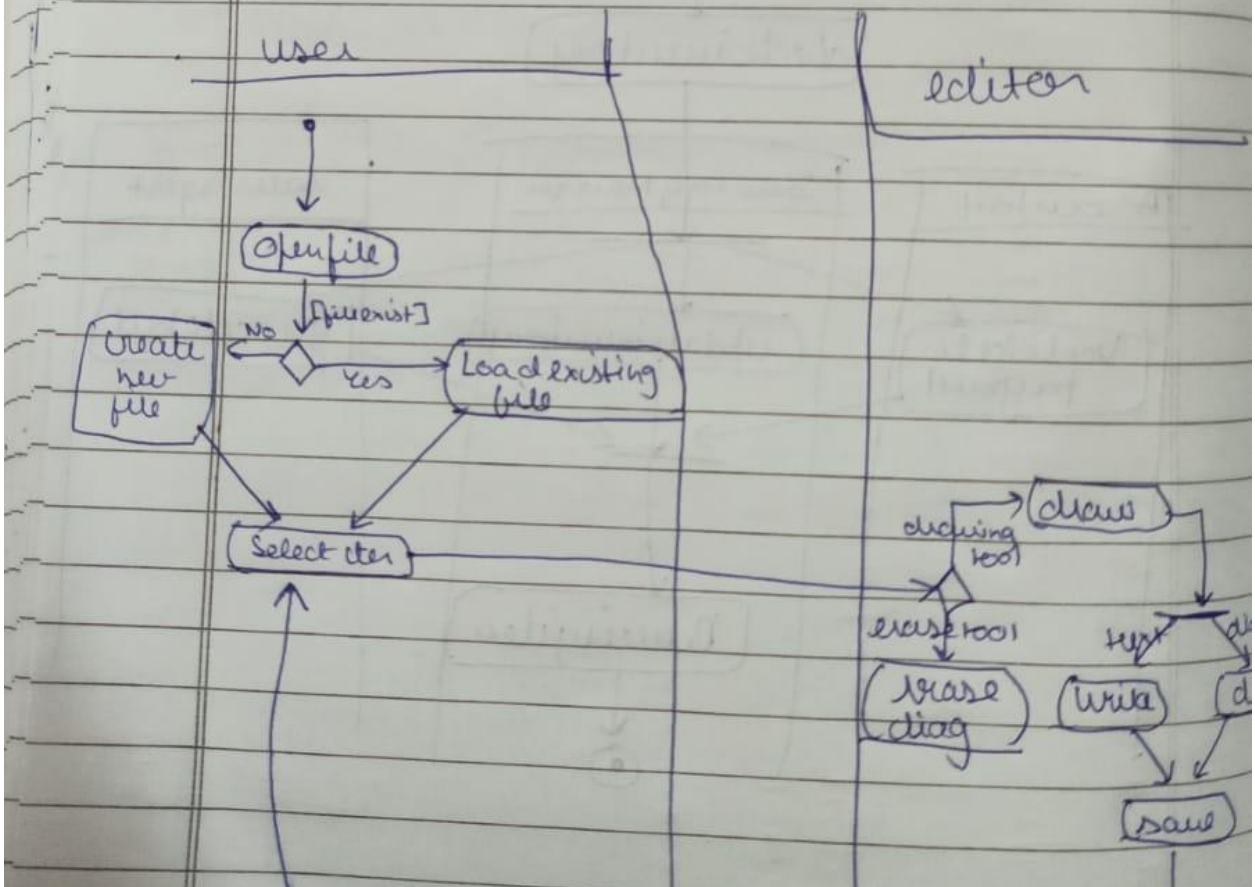

Course Details
exit
exit
Description
do / displayDesc
Enrollment
Course Enroll
do / enroll user
List
do / displayList
select a course
return
Login
Home page
do / show
Login out
Profile : ProfileView
Placement
do / displayplace
Display Profile
do / show usrpro
select view Profile
select edit
Edit
do / displayallopt to edit
save edit
Save
do / saveupdt
exit
exit
college info system - state diagram

Written usecase sequence and activity diags:

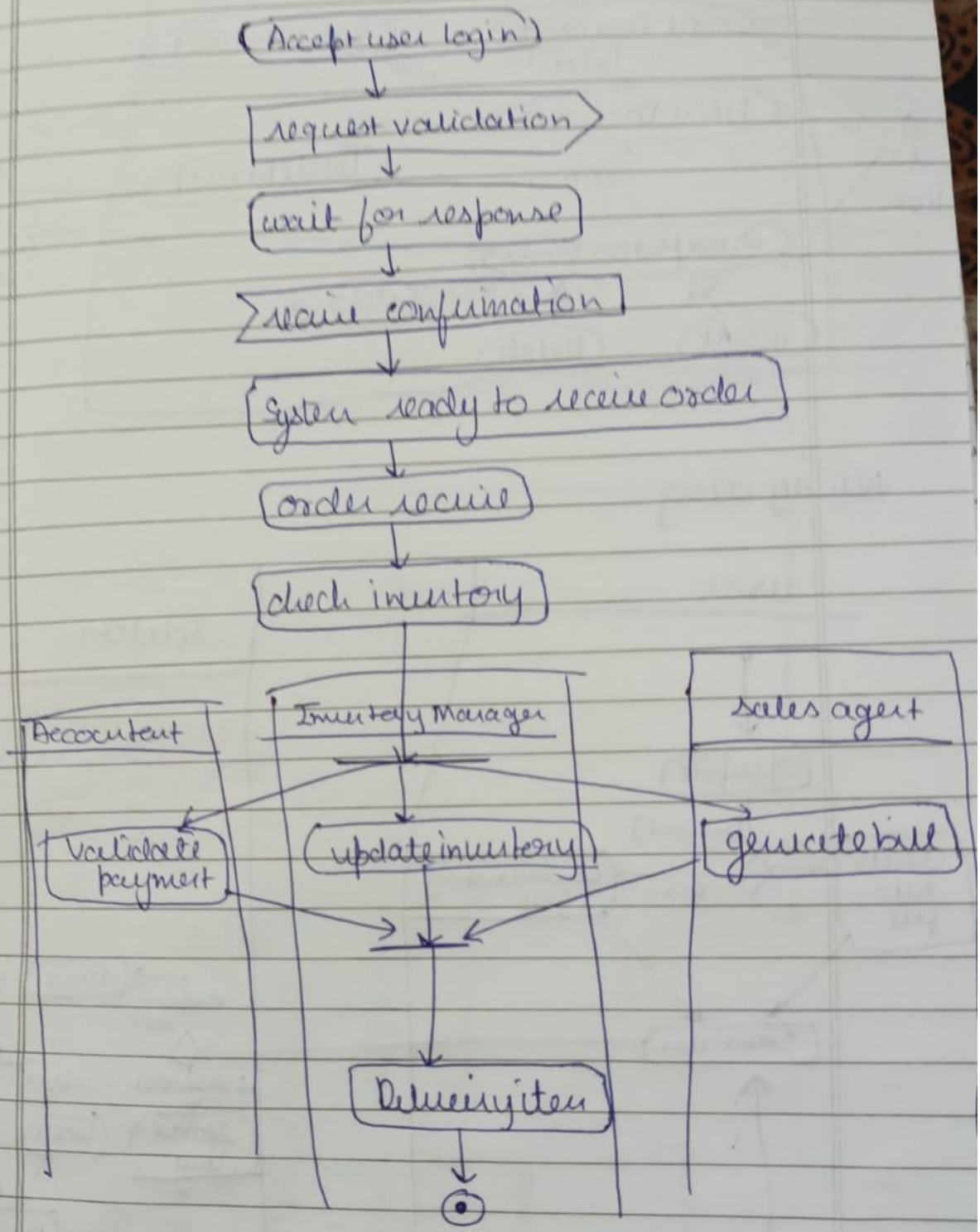

UI
tool box
Palette
3. create tools
4. tool selected
5. choose font
6. font selected
Date
Page



Activity diag

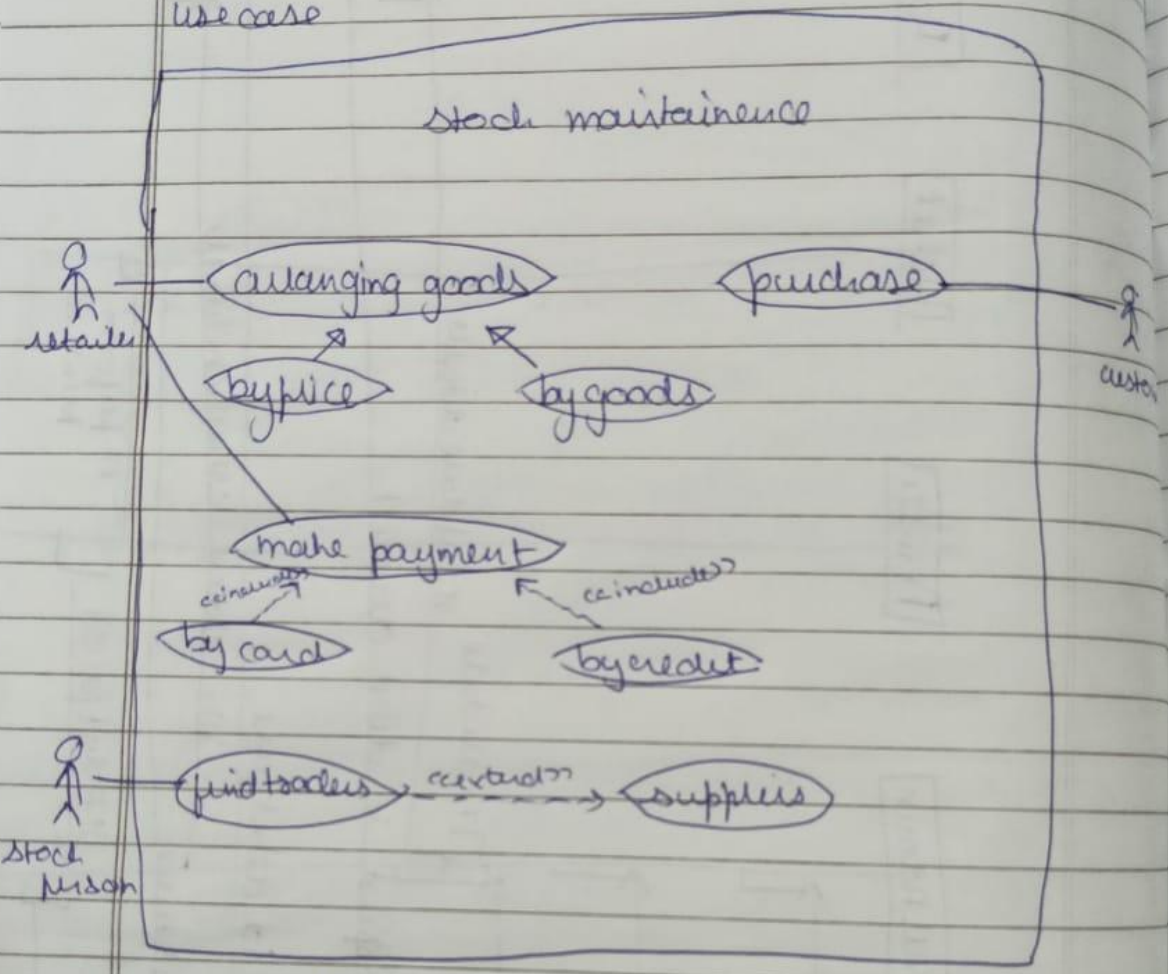


activity diag

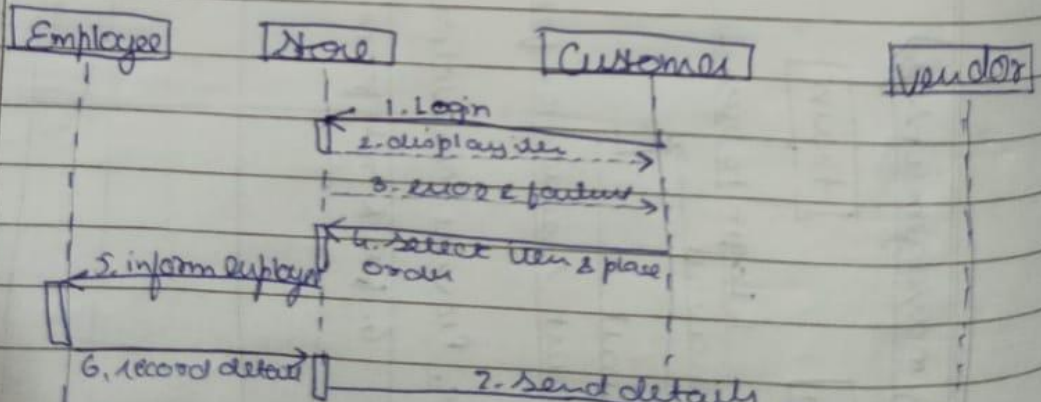


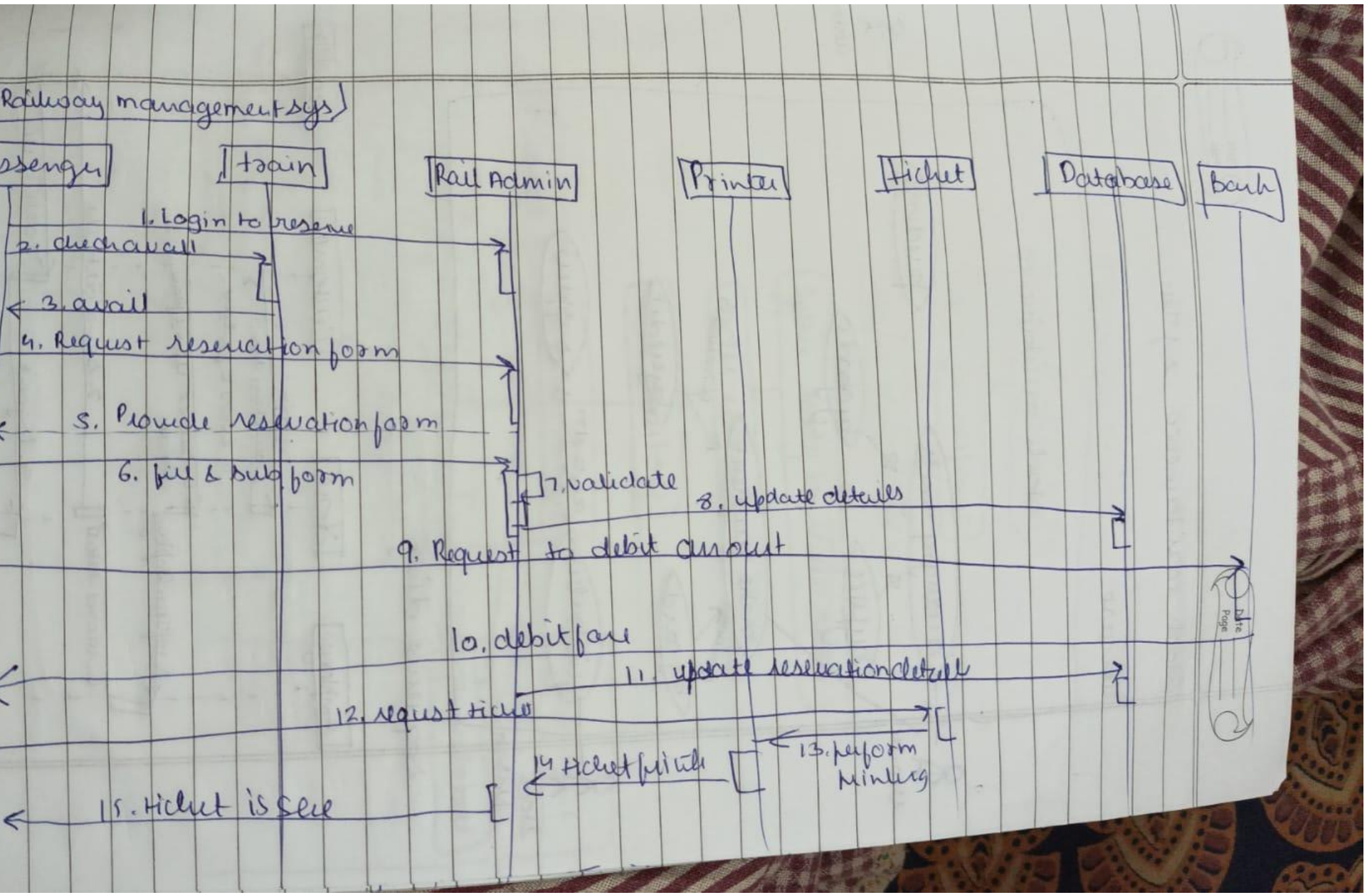
Stock maintainence system

use case



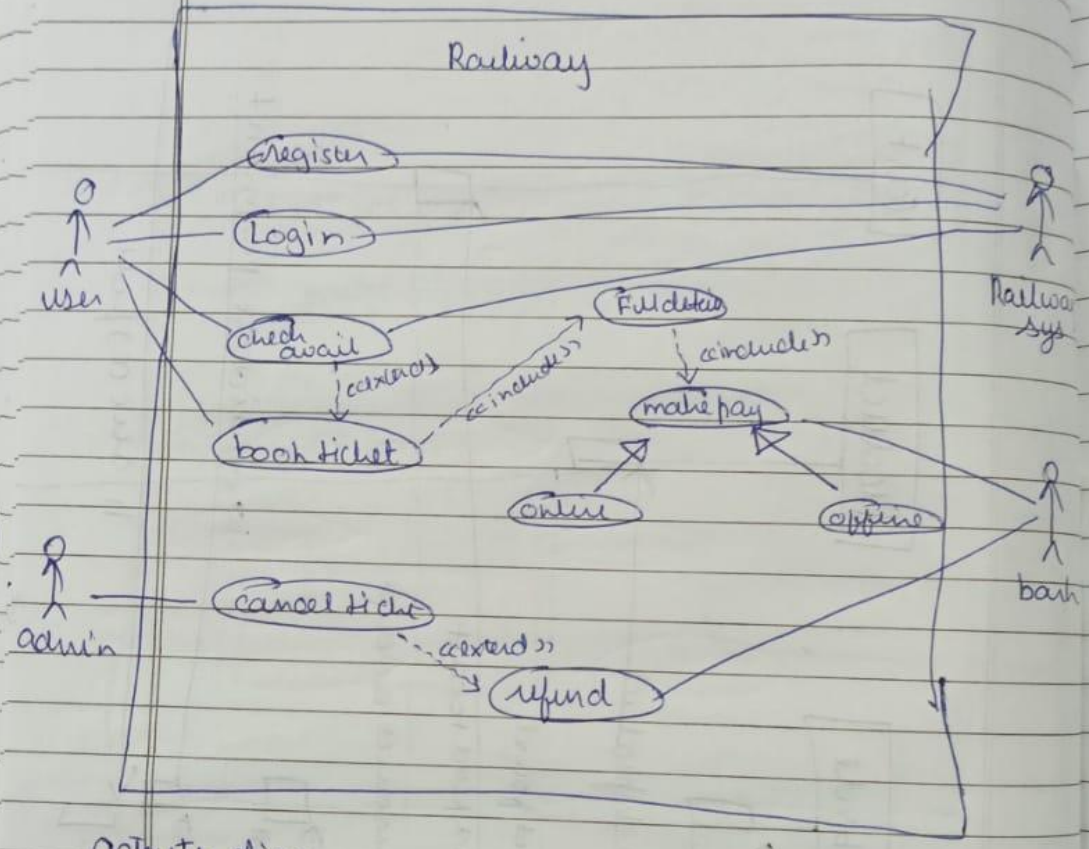
sequence diag



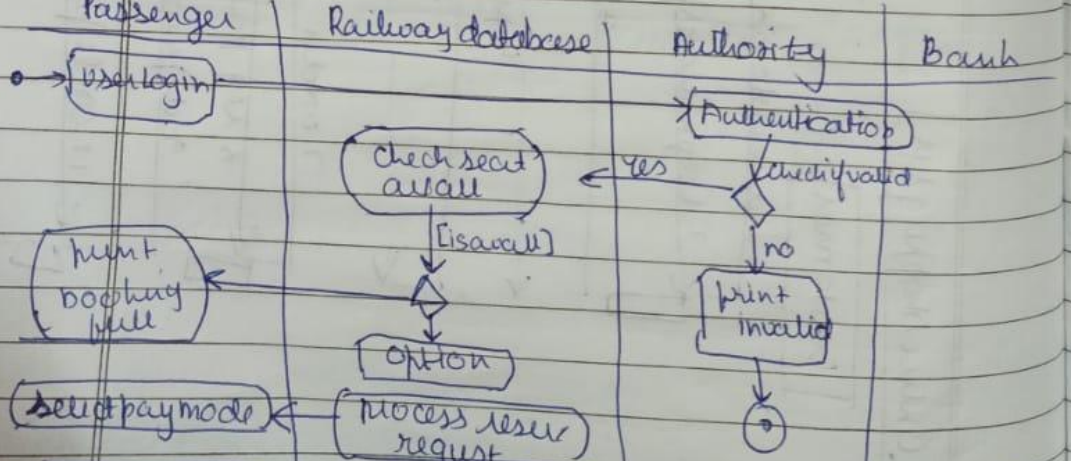

Railway management sys
ssenger
Train
Rail Admin
Printer
Ticket
Database
Bank
1. Login to reserve
2. check avail
3. avail
4. Request reservation form
5. Provide reservation form
6. fill & sub form
7. validate
8. update details
9. Request to debit amount
10. debit fare
11. update reservation detail
12. request ticket
13. perform Printing
14 Ticket print
15. Ticket is see

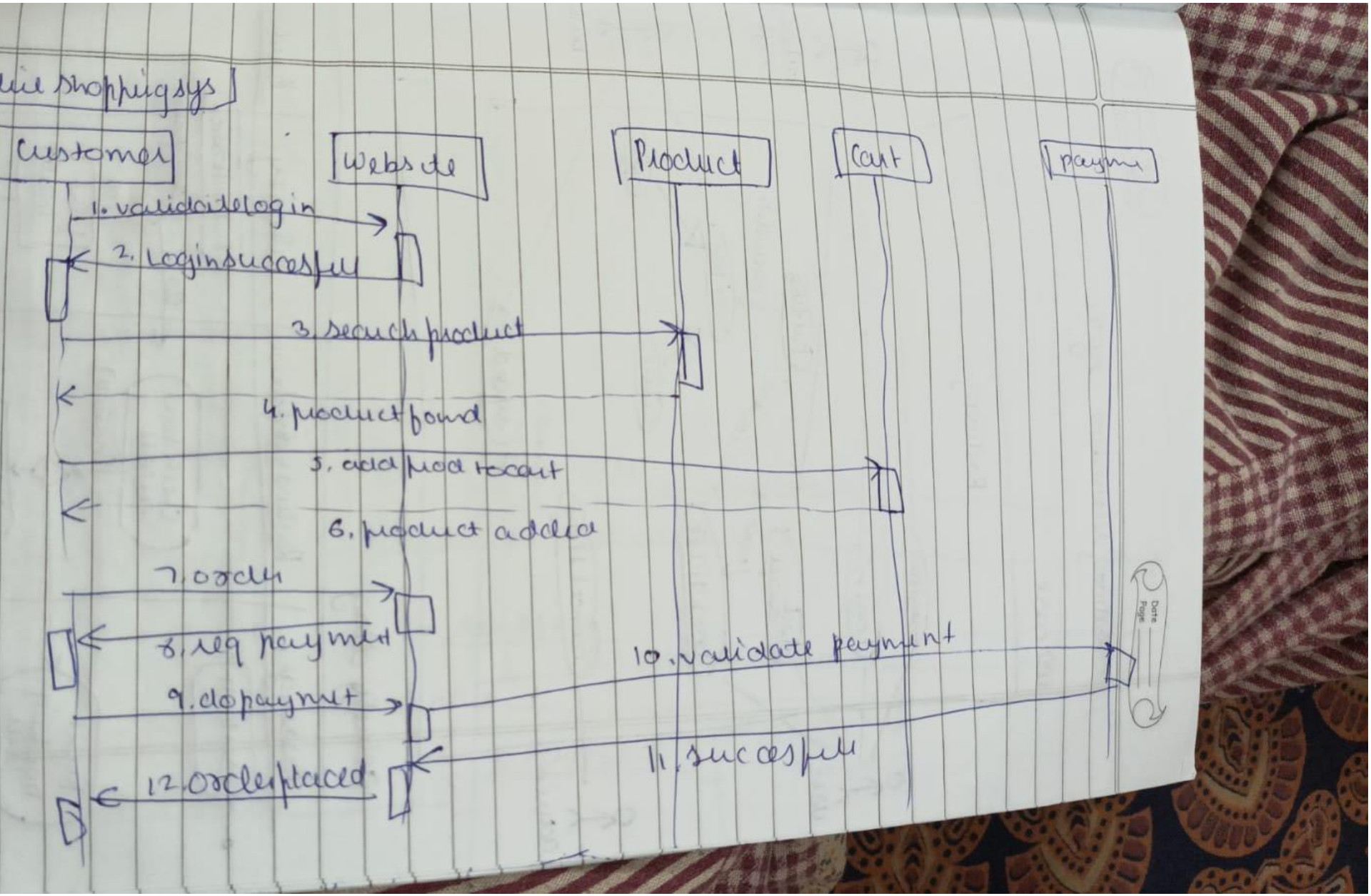
Railway reservation syster

usecase



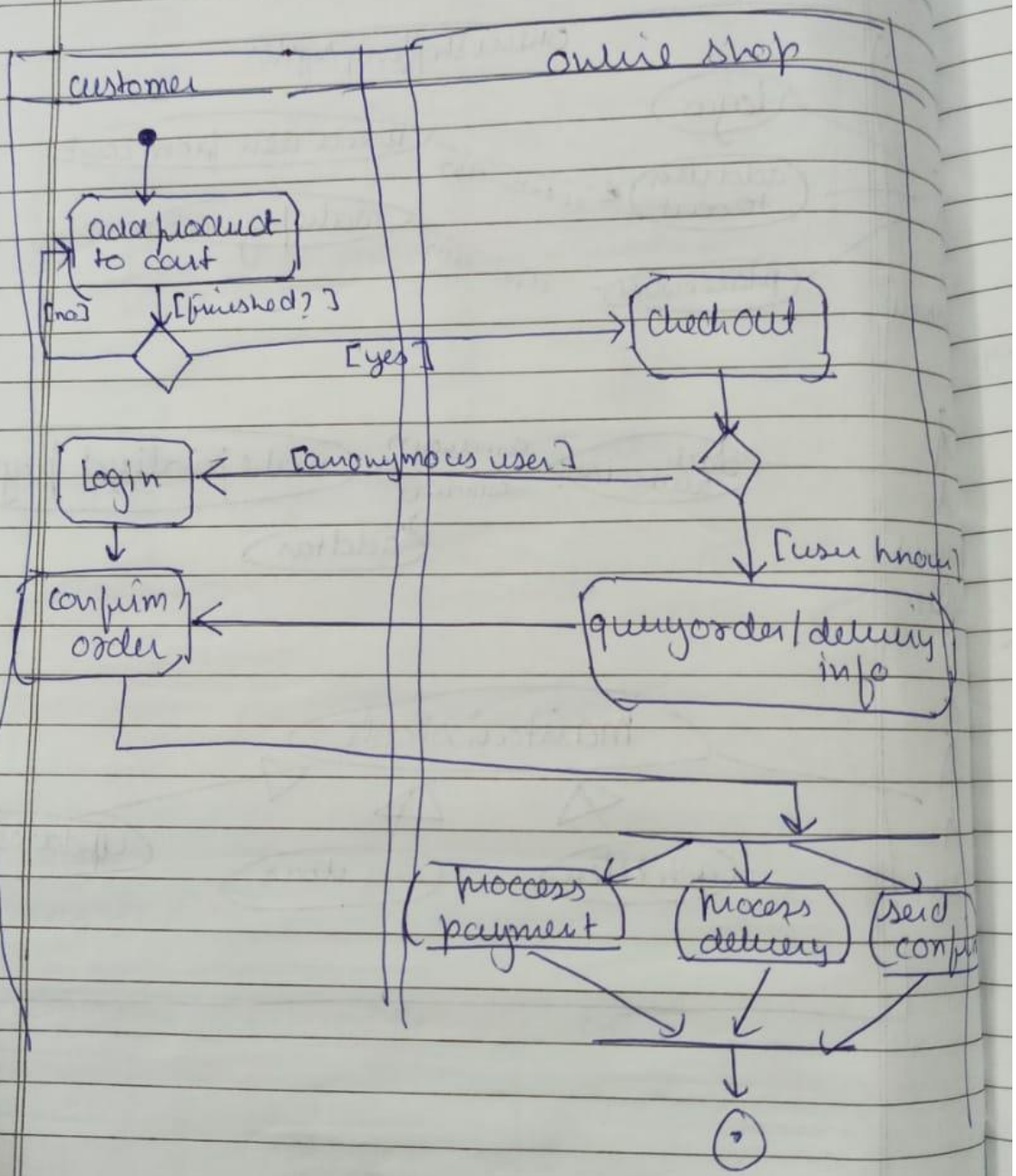
activity diag



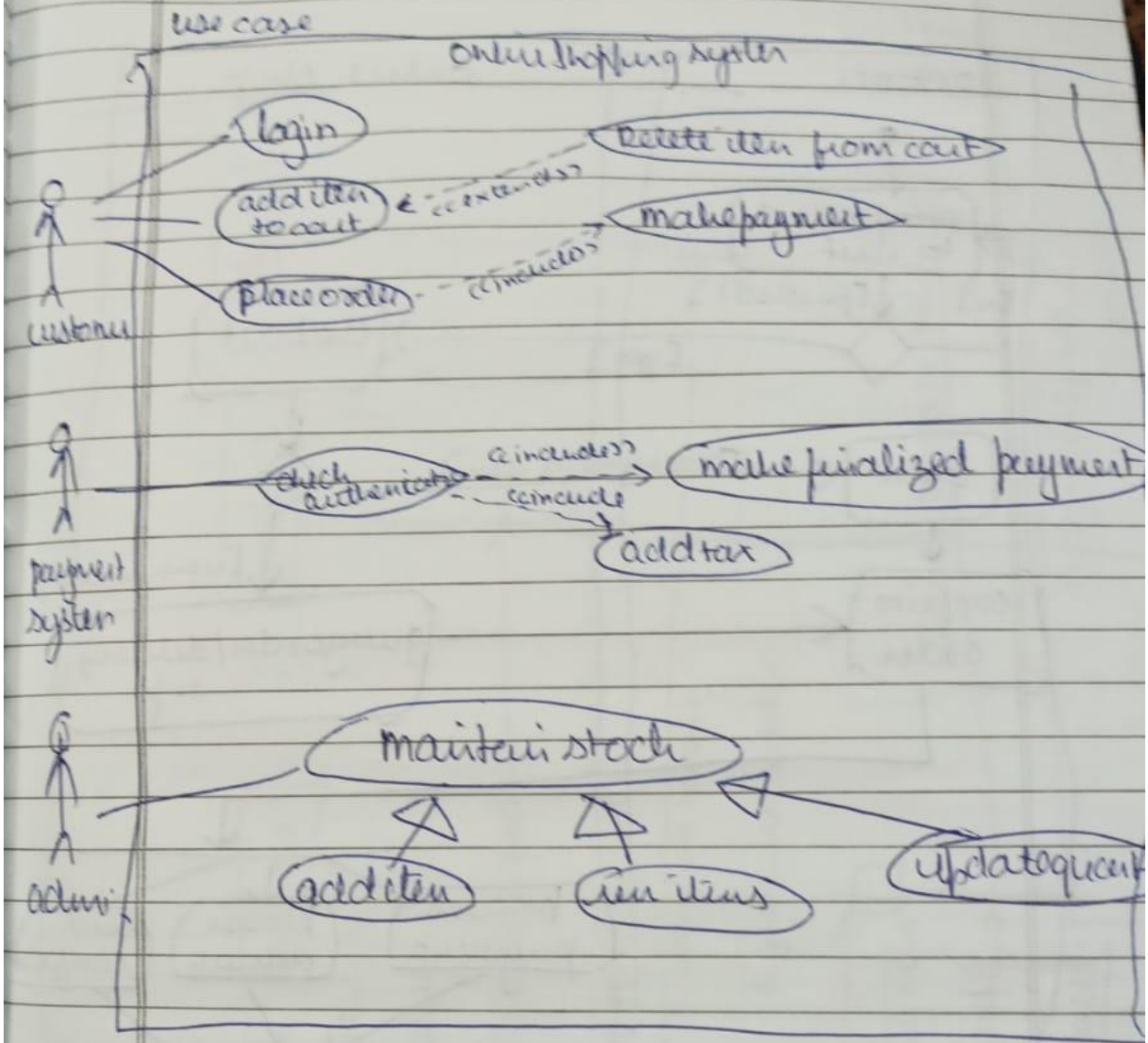

shopping sys
Customer
Website
Product
Cart
payme
1. validate login
2. Login successful
3. search product
4. product found
5. add prod to cart
6. product added
7. order
8. req payment
9. do payment
10. validate payment
11. successful
12. order placed

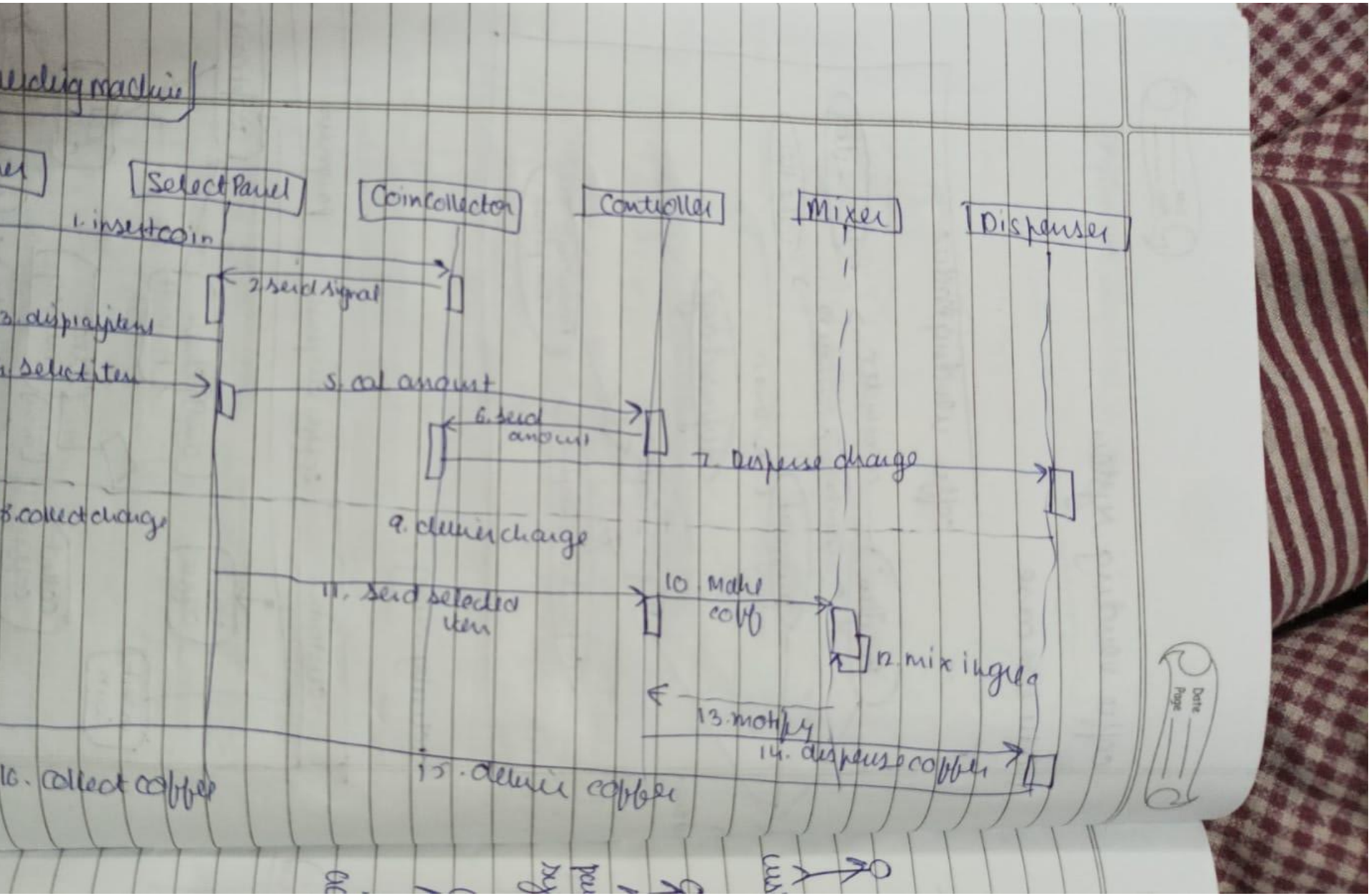
Date
Page

Activity diag



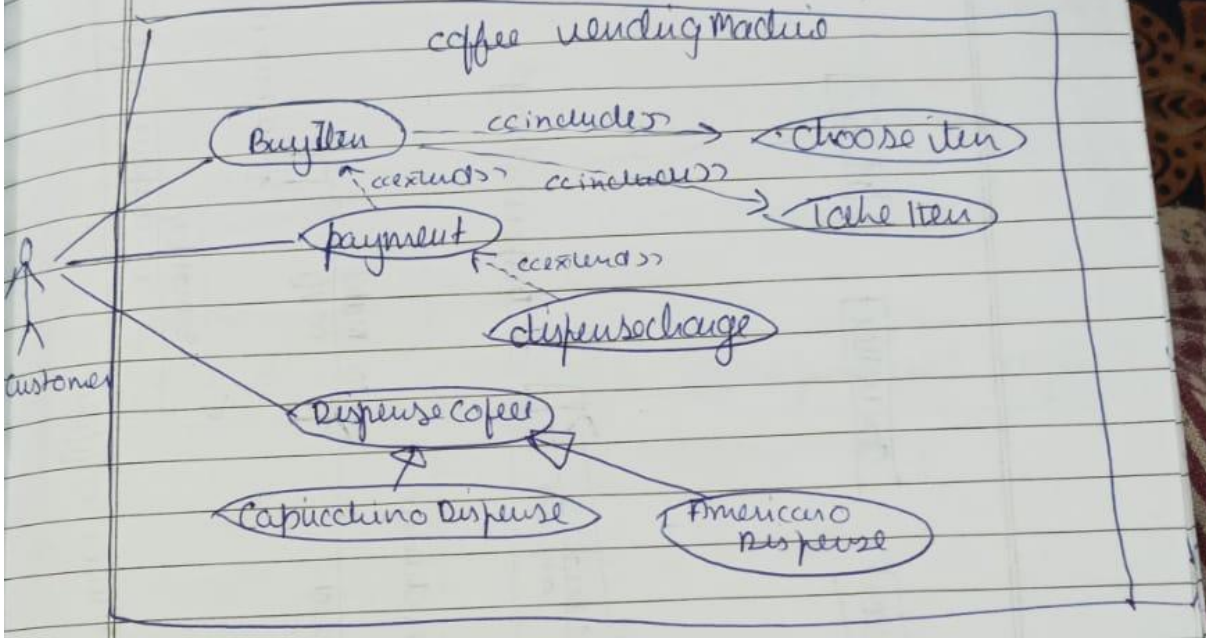
Online shopping system



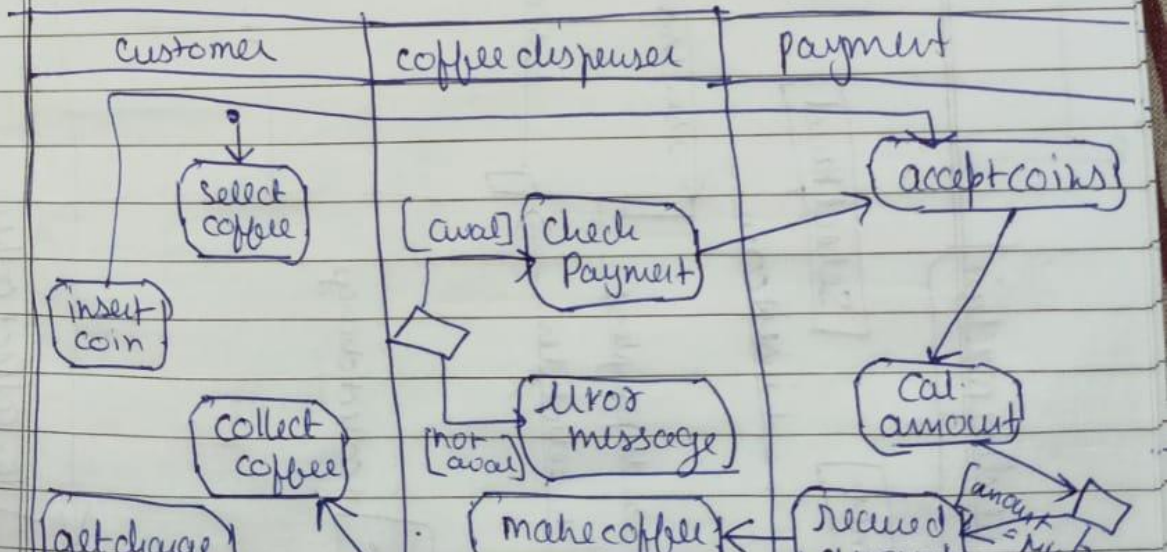

SelectPanel
CoinCollector
Controller
Mixer
Dispenser
1. insertcoin
2. send signal
3. displayitem
4. selectitem
5. collamount
6. send amount
7. dispense change
8. collectchange
9. deliverchange
10. make coffe
11. send selected item
12. mixingred
13. notify
14. dispense coffee
15. deliver coffee
16. collect coffee

coffee vending system

add use case



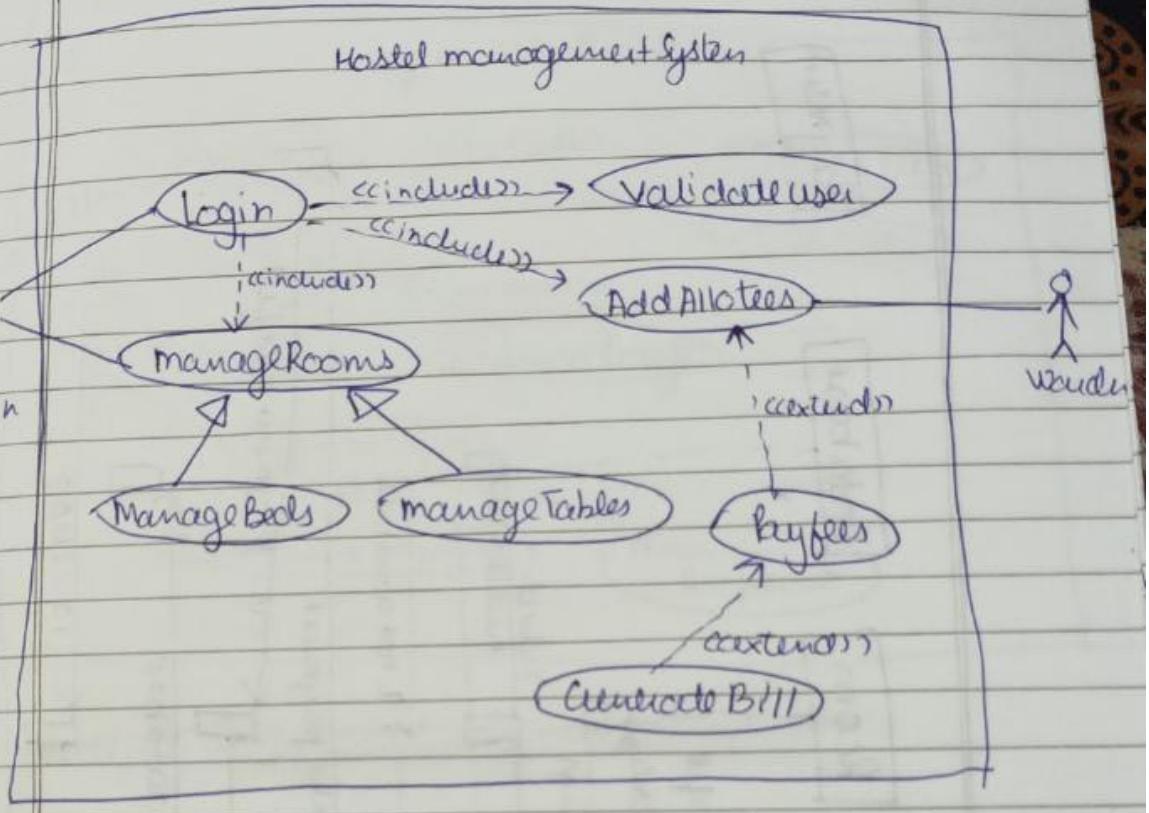
activity diag



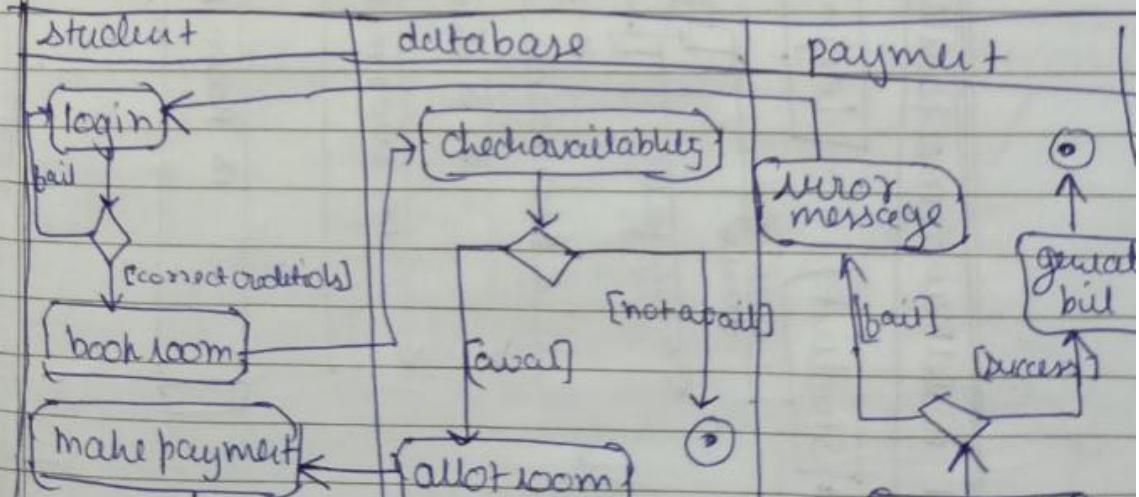
advanced sequence diag

Hostel management system

usecase

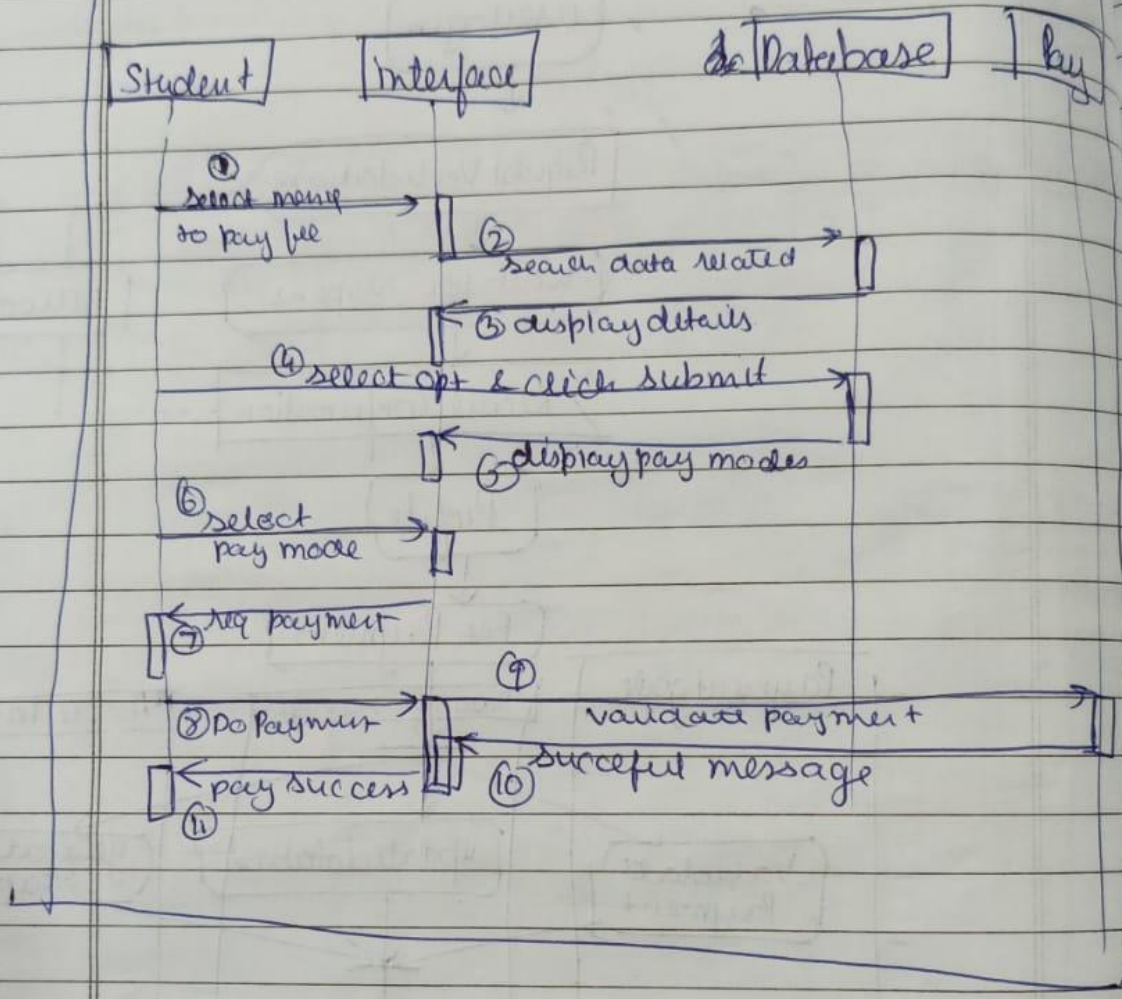


activity diag

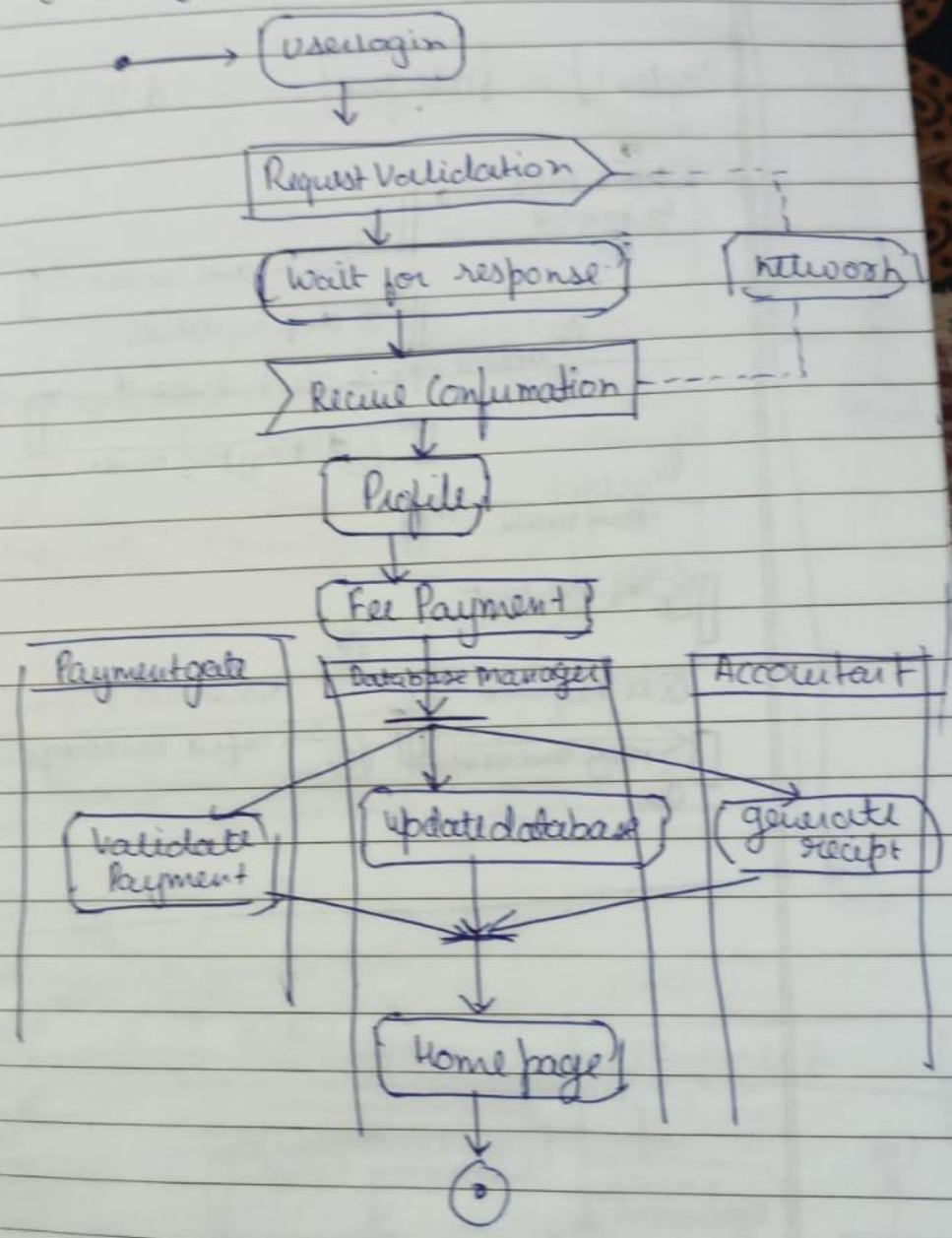


adv sequence diag

sd college manage sys



activity diag



college info system →→

use case