B.M.S. COLLEGE OF ENGINEERING BENGALURU

Autonomous Institute, Affiliated to VTU

Lab Record

Object Oriented Modelling and Design

Submitted in partial fulfillment for the 6th Semester Laboratory

Bachelor of Technology
in
Computer Science and Engineering

Submitted by:

ABHISHIKAT KUMAR SONI
(1BM18CS004)

Department of Computer Science and Engineering

B.M.S. College of Engineering
Bull Temple Road, Basavanagudi, Bangalore 560 019
Mar-June 2021

B.M.S. COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Object-Oriented Modelling and Design(20CS6PCOMD) laboratory has been carried out by ABHISHIKAT KUMAR SONI(1BM18CS004) during the 6th Semester Mar-June-2021.

Signature of the Faculty Incharge:

NAME OF THE FACULTY:
Pradeep Sadanand
Assistant Professor
Department of Computer Science and Engineering
B.M.S. College of Engineering, Bangalore

Table of Contents

1. College Information System

1.1 Problem Statement

To build an efficient system to manage information about students, college department, faculties and all other activities taking place in college.

1.2 Software Requirements Specification of College Information System

- Purpose

To automate and centralize the database for the College Information System. We are attempting to improve our existing system that runs on pen and paper.

- Scope

It is more efficient and convenient for the colleges. It reduces the manpower needed to perform the entire administrative tasks. Since all the work is done by computer, there is less chance of error.

- Functional Requirements

i) Admin can login using credentials provided to him.

ii) Student details are stored in centralized database by administrator

iii) Admin can add and update the student information

iv) Student can login using the credentials provided

v) Student can check attendance status and marks

vi) Faculty can post assignments

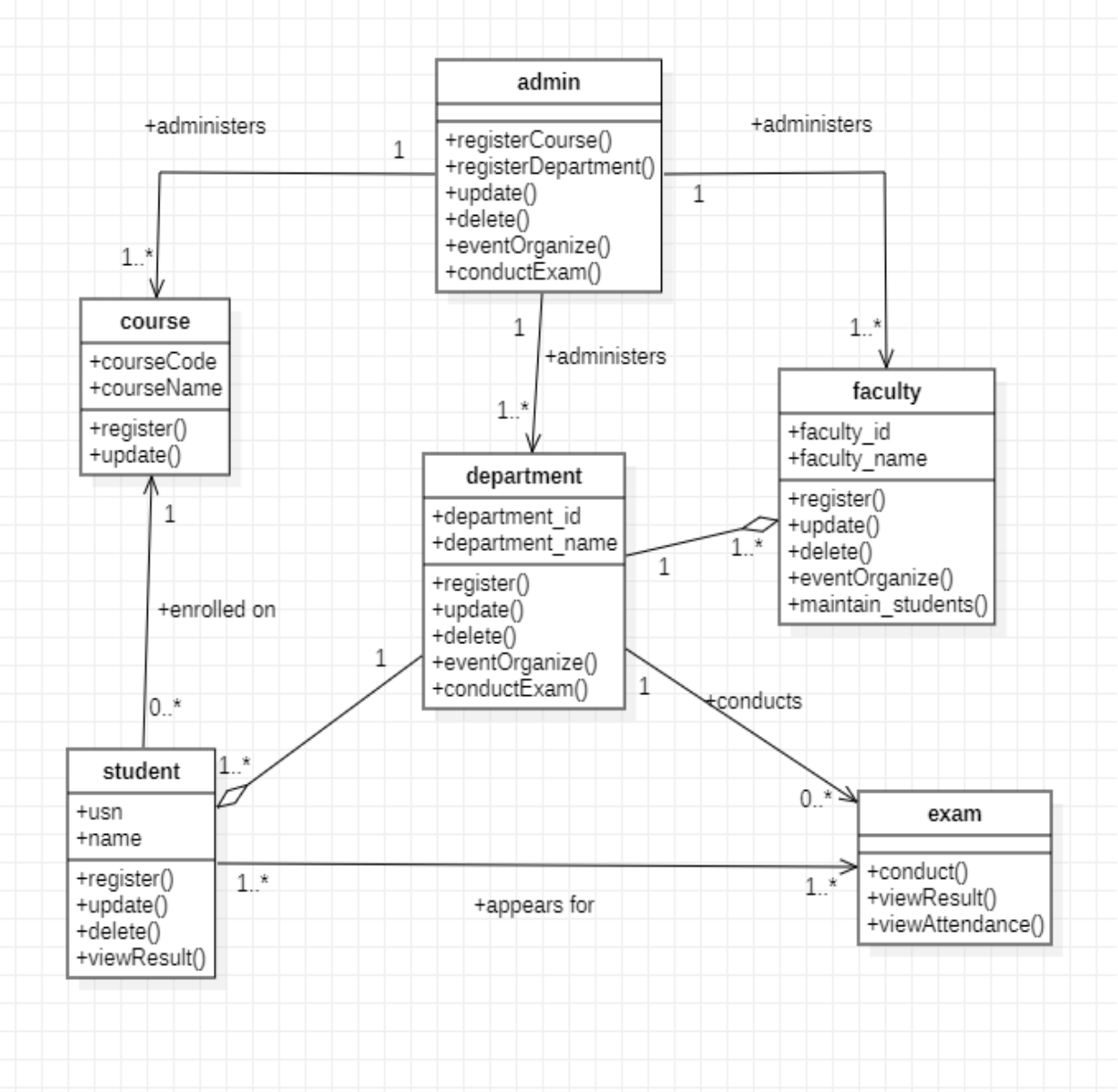
- Non-Functional Requirements

i) The system should be easy to handle

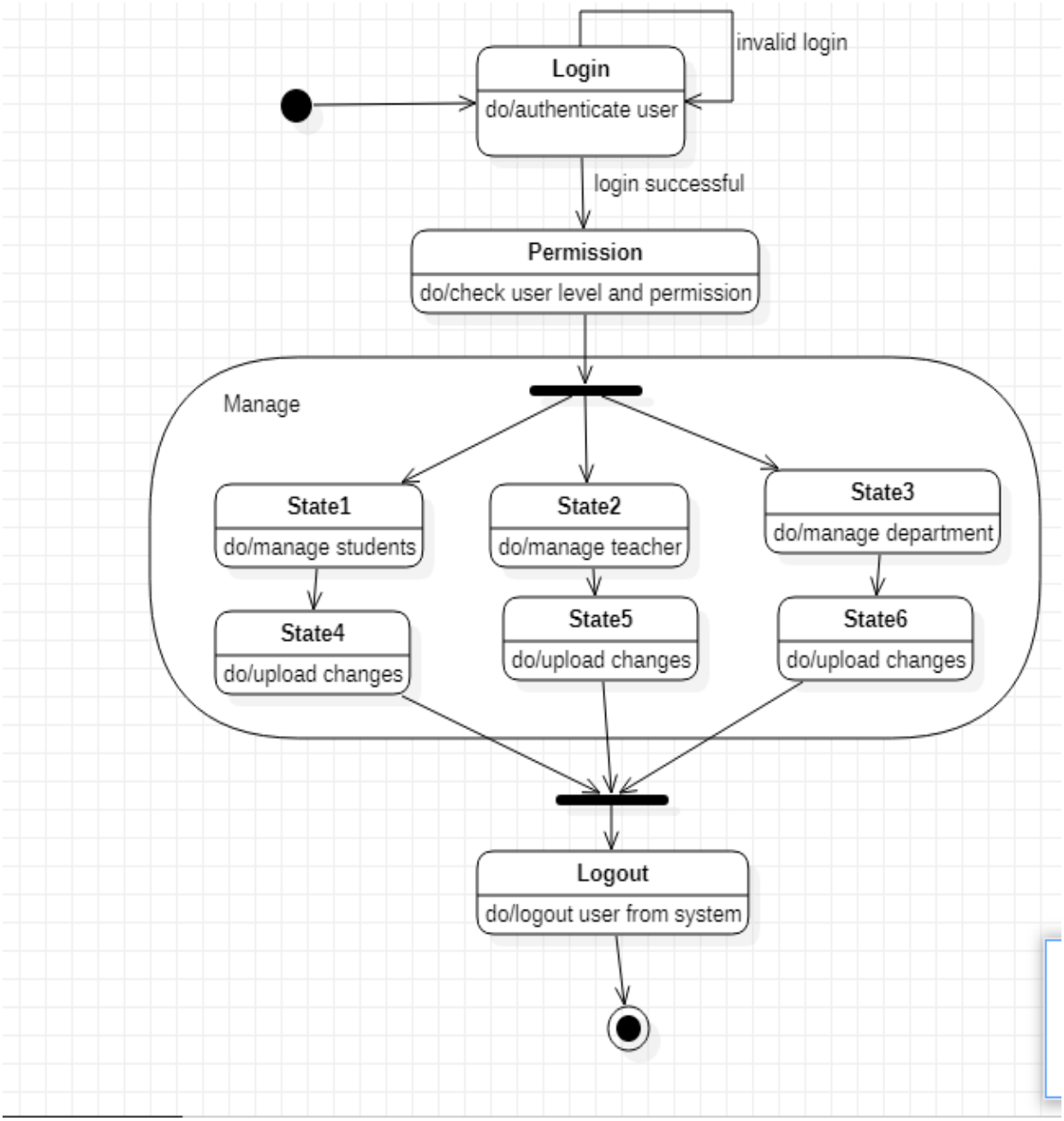
ii) System should give expected performance result

iii) Response time should be less

1.3 Class diagram



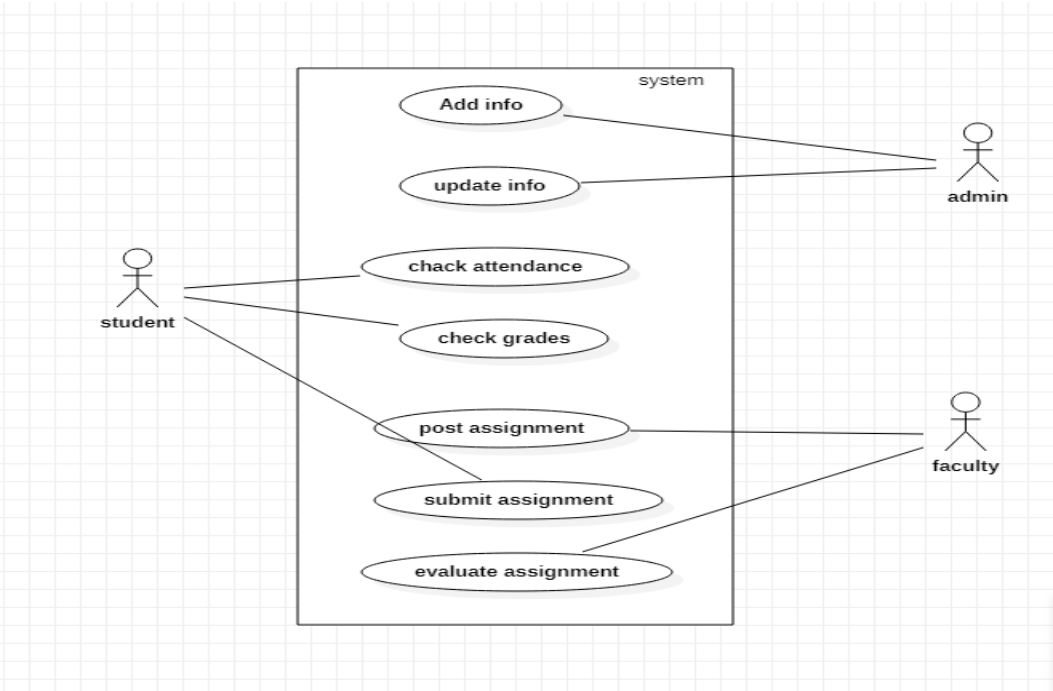
1.4 State diagram



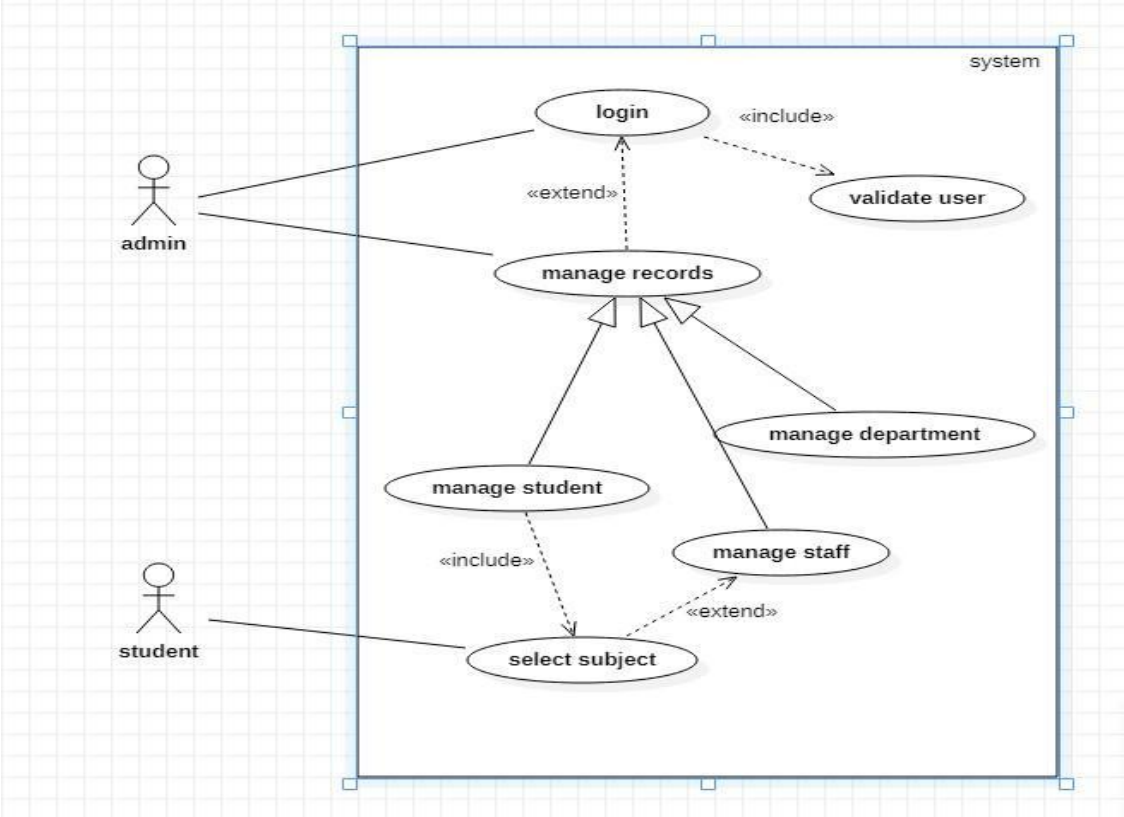
1.5 Use case diagram

Simple use case

diagram:

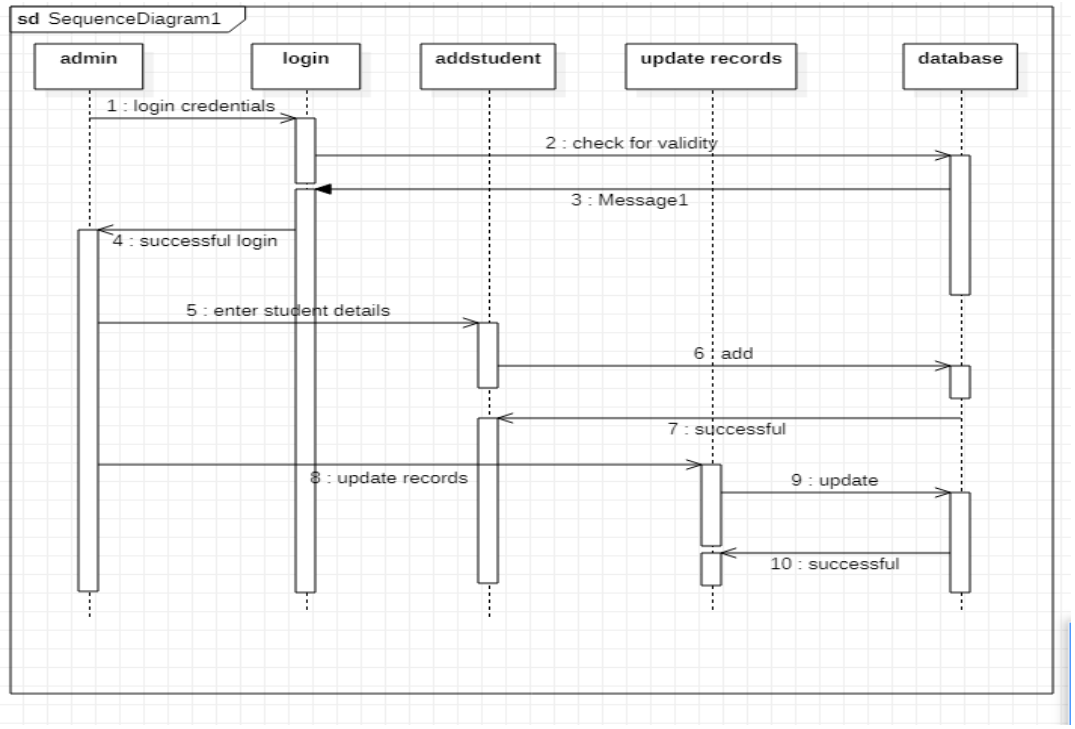


Advanced use case diagram:

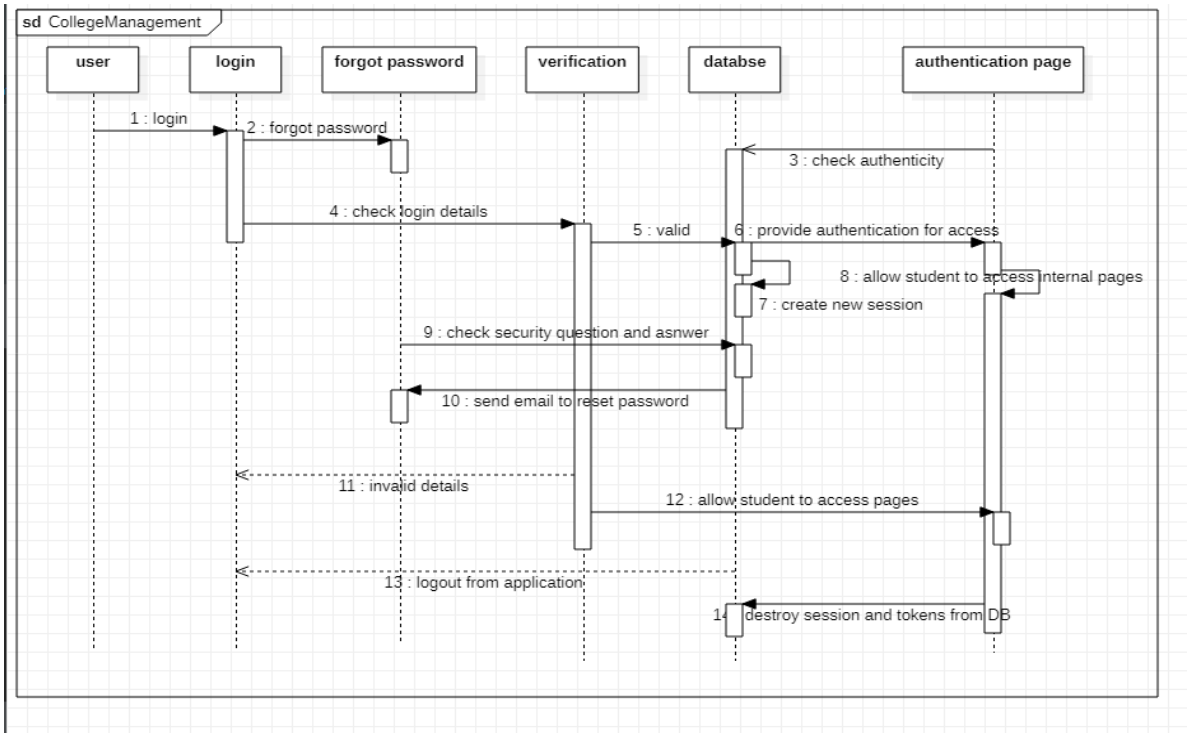


1.6 Sequence diagram

Simple sequence diagram:

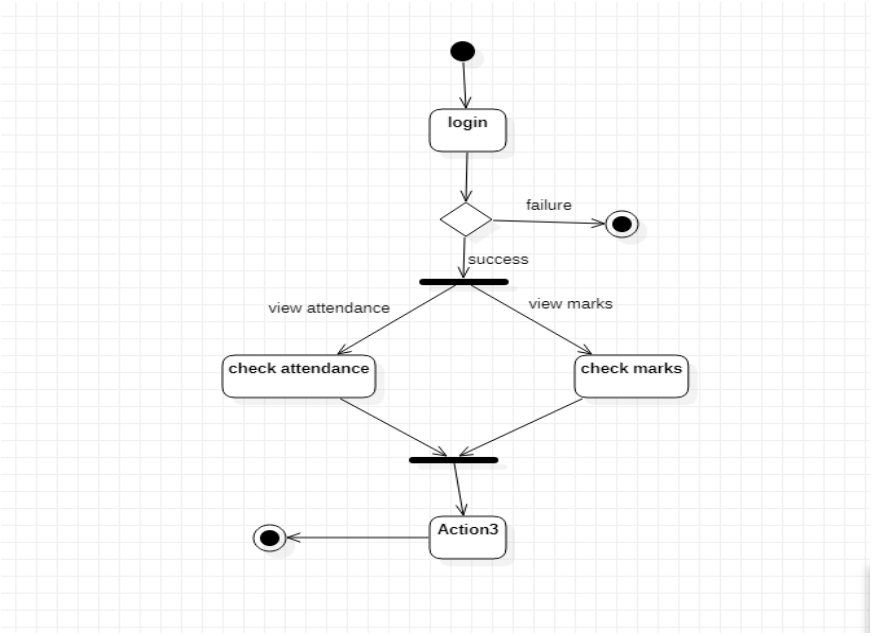


Advanced sequence diagram:

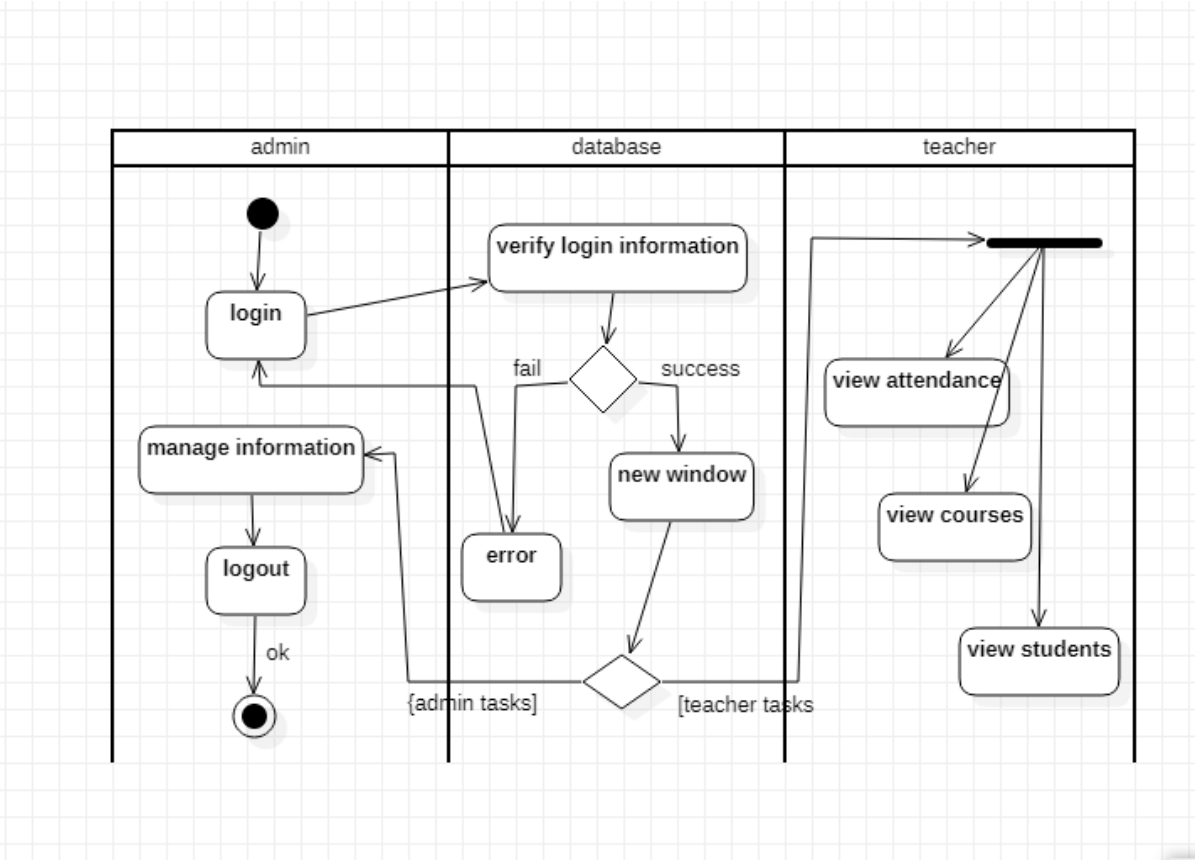


1.7 Activity diagram

Simple activity diagram:



Advanced activity diagram:



2. Hostel Management System

2.1 Problem Statement

To build an efficient system that deals with the problem on managing the hostel and avoids the problem that occurs when carried out manually.

2.2 Software Requirements Specification

- Purpose

To centralize the database and thus provide consistent data to all the employees in the hostel and to make the system more user friendly by providing an intensive user interface.

- Scope

This system is designed for maintaining the records in an organized manner and to replace old paperwork system. Modifications can be done easily and all the calculations and accounting work would be accurate.

- Functional Requirements

i) Admin can allot different students to the different hostels

ii) Admin can control the status of the fee payment

iii) Admin can modify the student records

iv) Students can lodge a complaint about an issue in room, mess food and hygiene

v) Mess manager can view the detailed feedback of the students

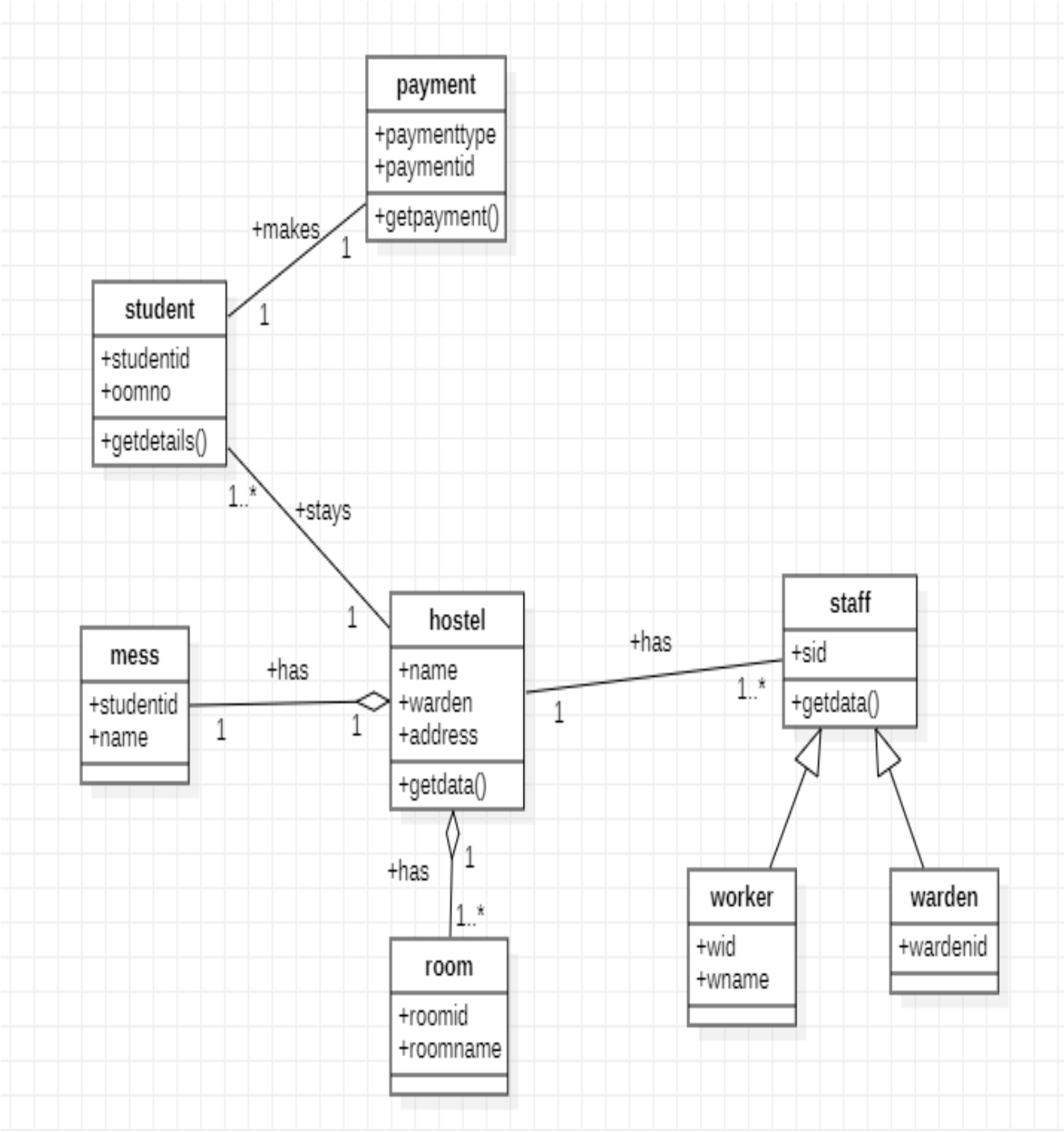
- Non-Functional Requirements

i) Software shall support use of multiple users at a time

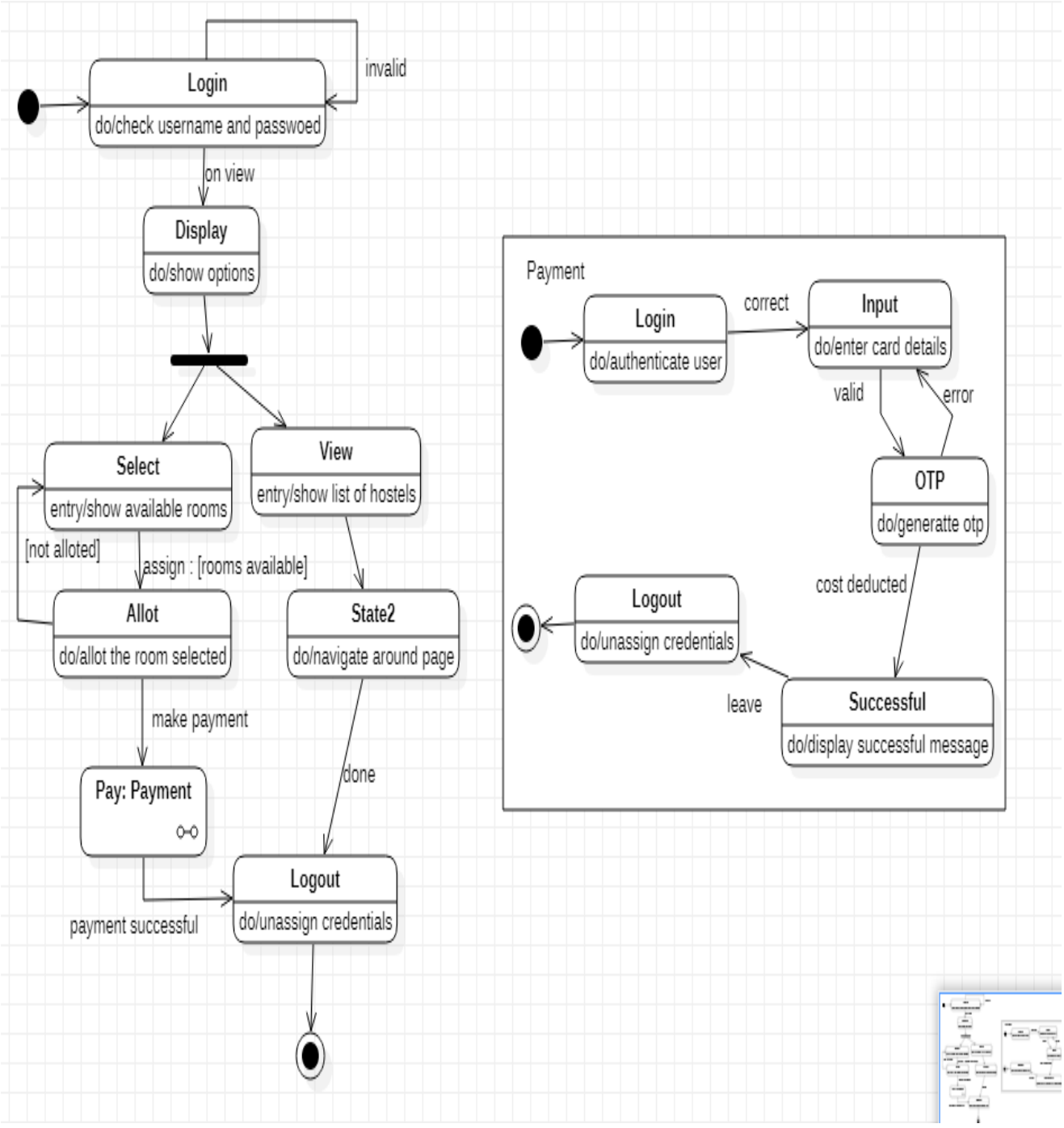
ii) Restrict communications between some areas of the program

iii) Check data integrity for critical variables

2.3 Class diagram



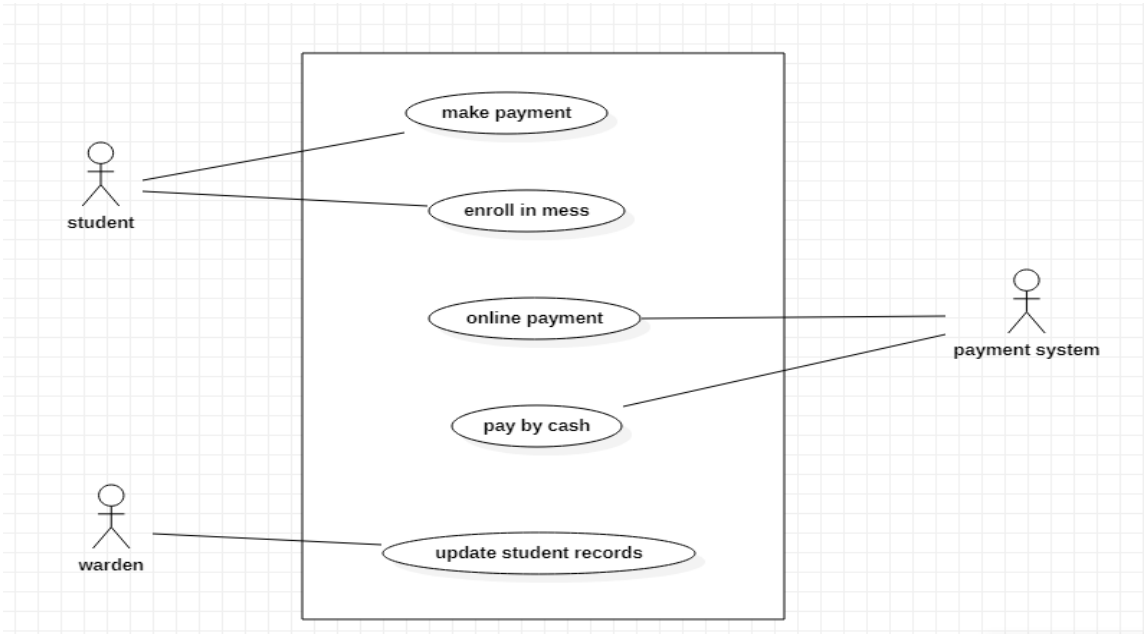
2.4 State diagram



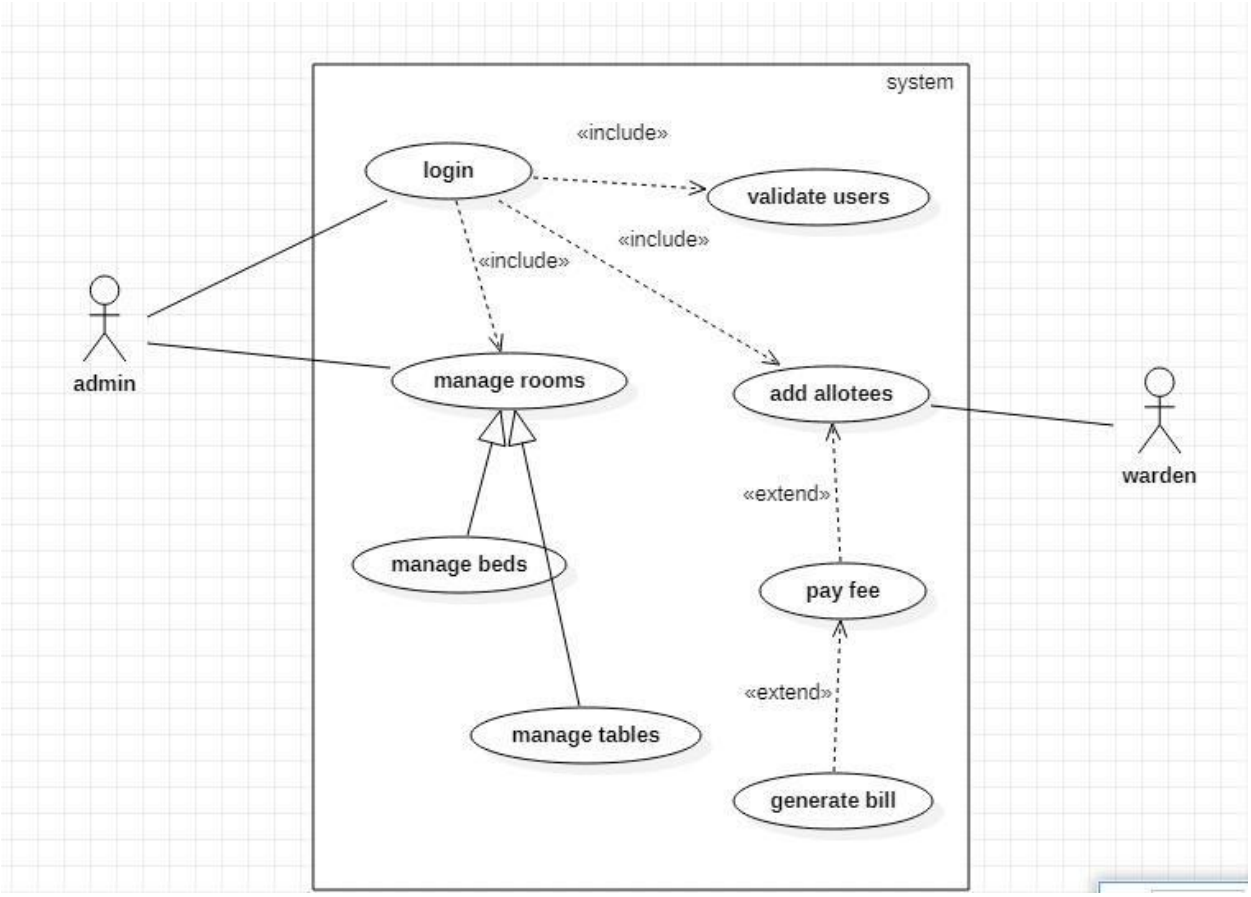
2.5 Use case diagram

Simple use case

diagram:

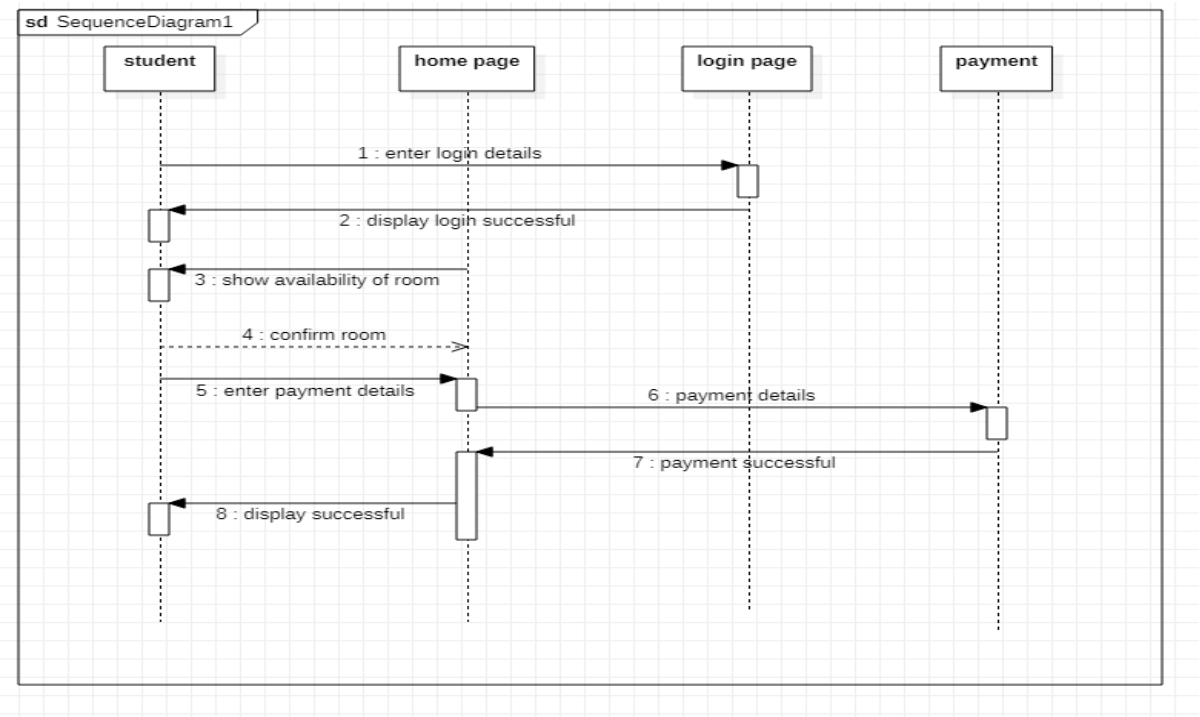


Advanced use case diagram:

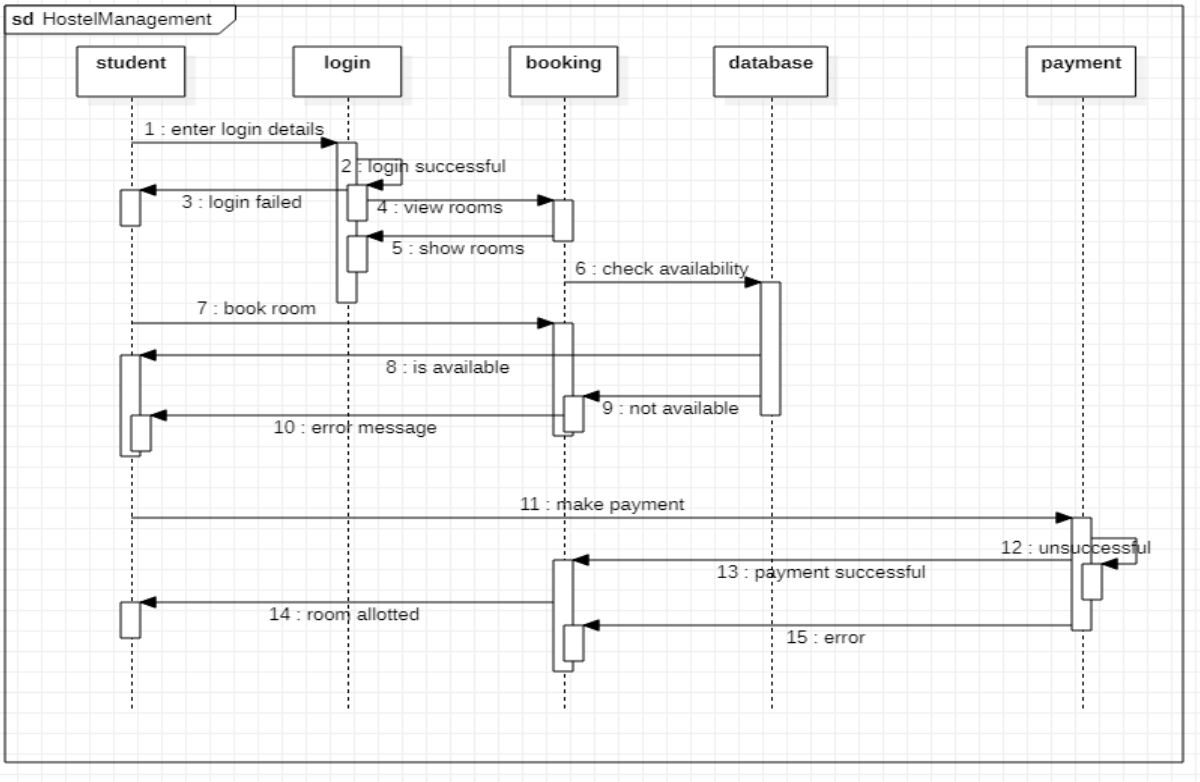


2.6 Sequence diagram

Simple sequence diagram:

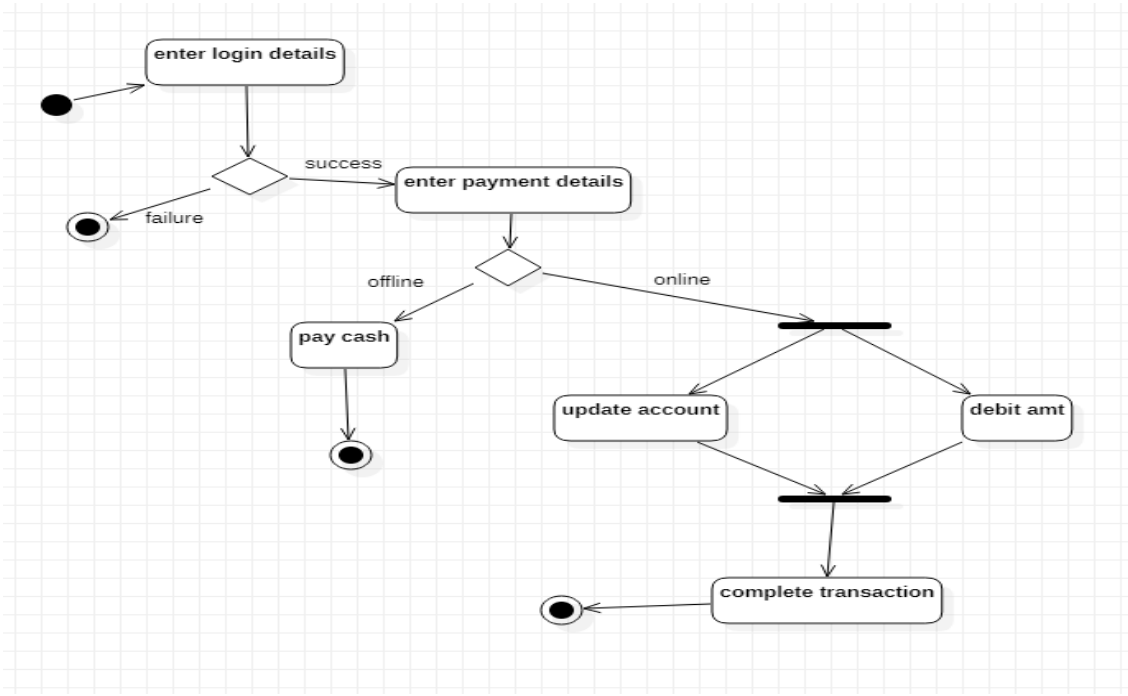


Advanced sequence diagram:

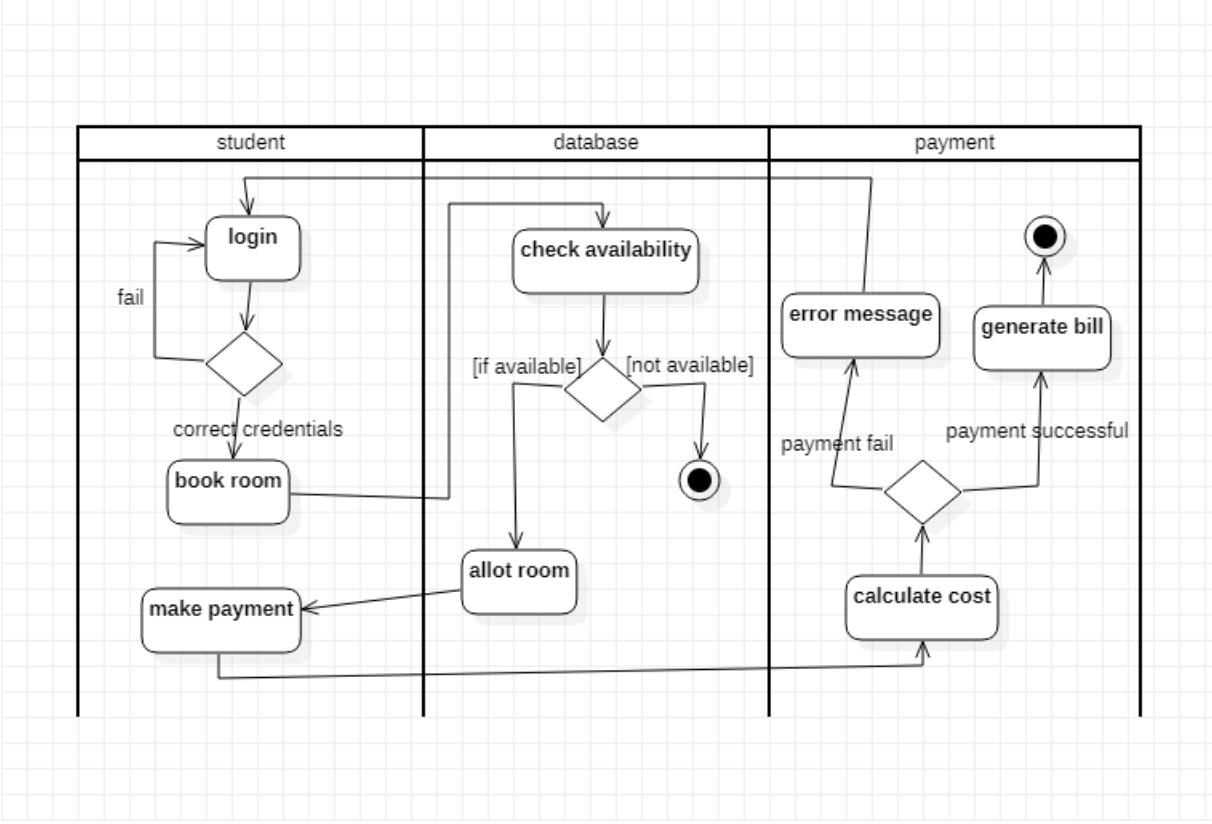


2.7 Activity diagram

Simple activity diagram:



Advanced activity diagram:



3. Stock Maintenance System

3.1 Problem Statement

To design and develop the Stock Maintenance System for processing the stock details of an industry or organization.

3.2 Software Requirements Specifications

- Purpose

The entire process of Stock maintenance is done in a manual manner. Considering the fact that the number of customers for purchase is increasing every year, a maintenance system is essential to meet the demand. So this system uses several programming and database techniques to reduce the work involved in this process.

- Scope

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. It provides a communication platform between the customer and the sales person.

- Functional Requirements

i) Stock Manager can enter the information of sales, purchase orders and create reports.

ii) The customer can view the availability of the required items and the price of the items.

iii) Stock Manager can add the items and update the database.

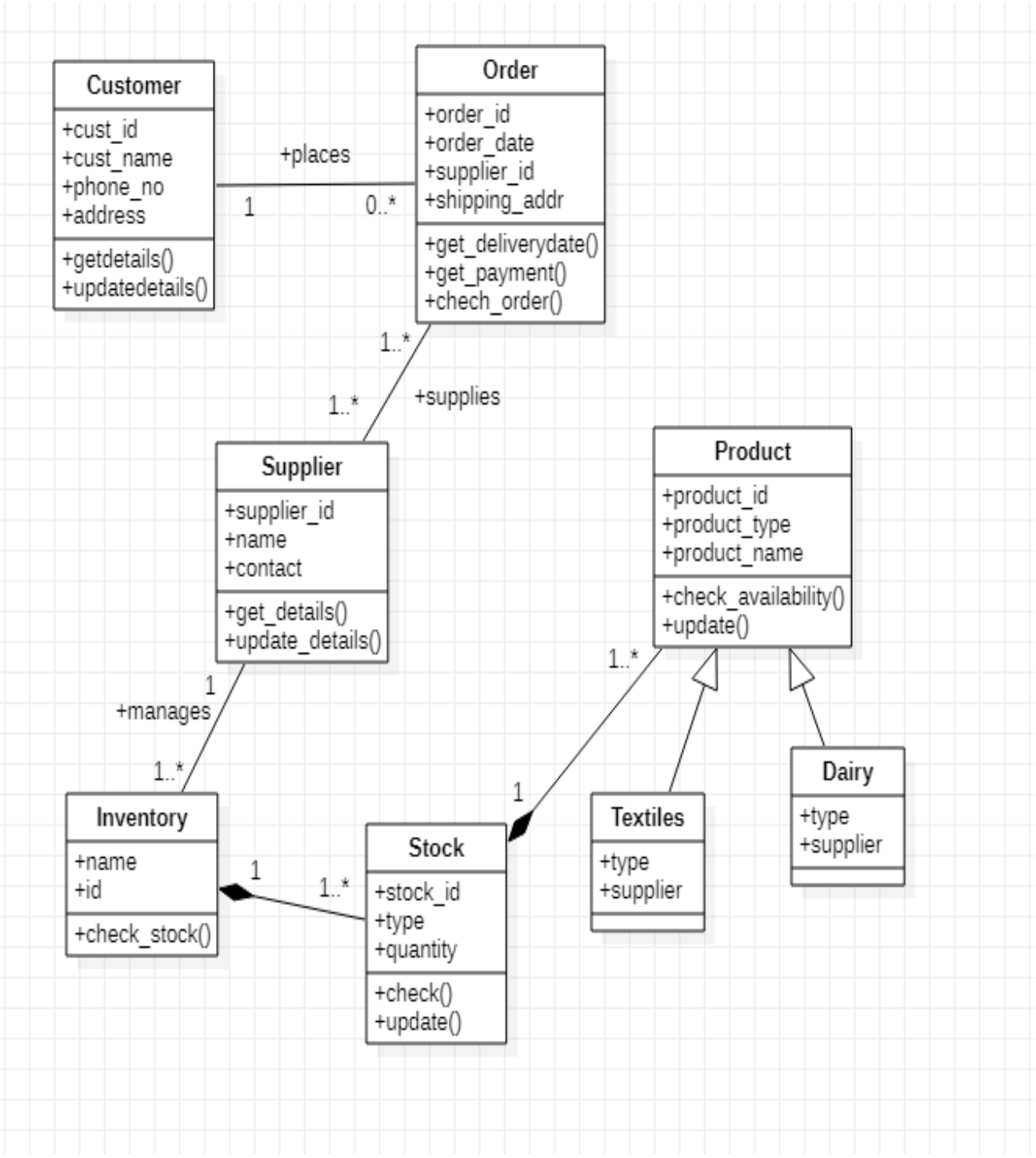
- Non-Functional Requirements

i) The load time for user interface screens will take no longer than two seconds.

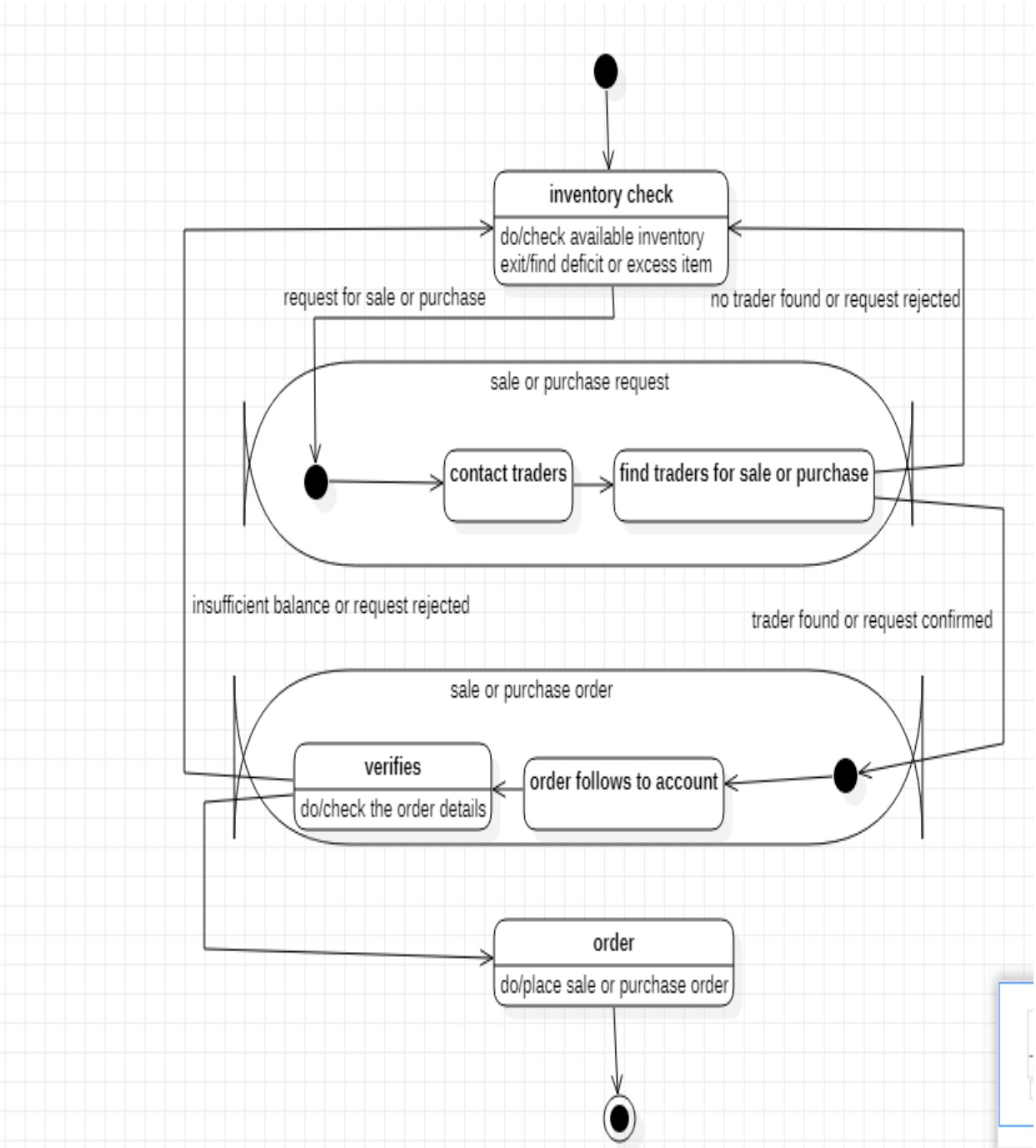
ii) The system should have an availability of 99.99%.

iii) The system shall be easy to migrate or backed up via another use drive.

3.3 Class diagram



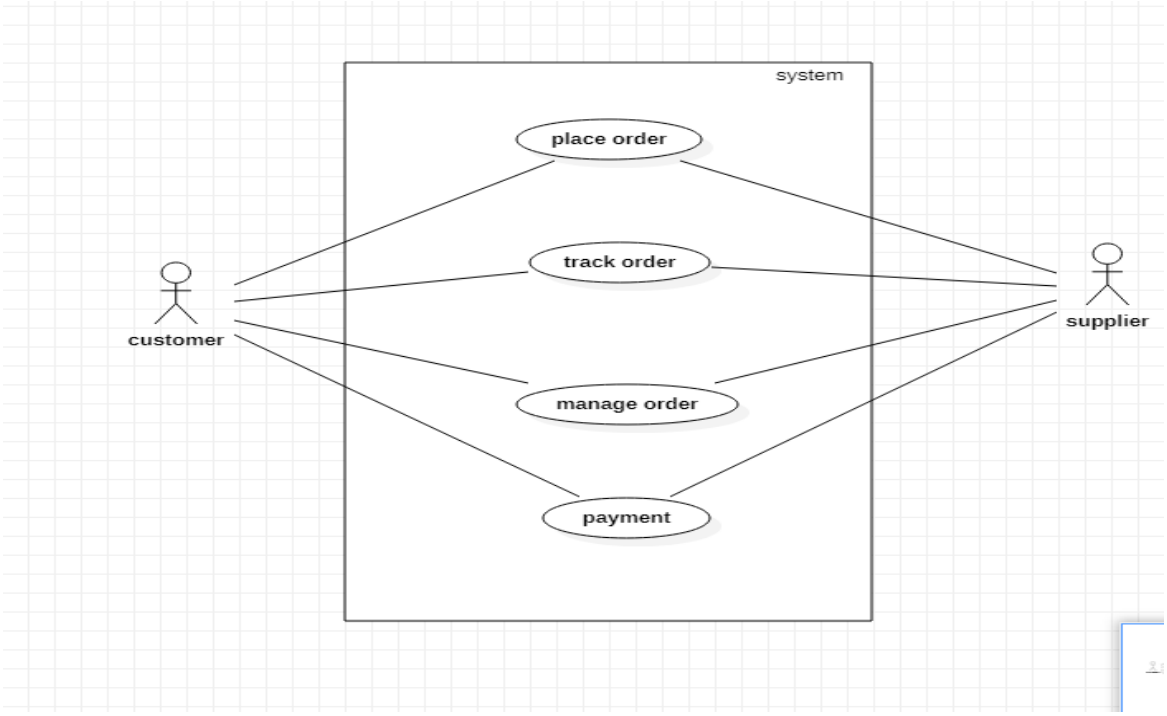
3.4 State diagram



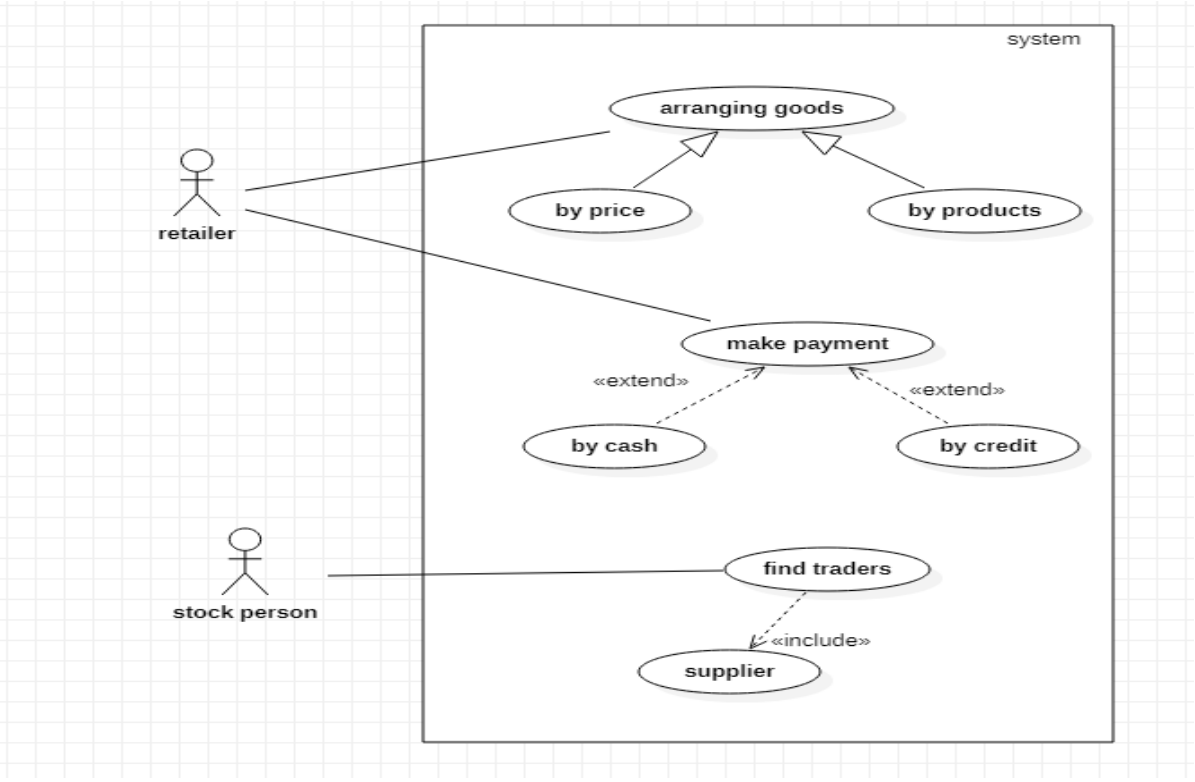
3.5 Use case diagram

Simple use case

diagram:

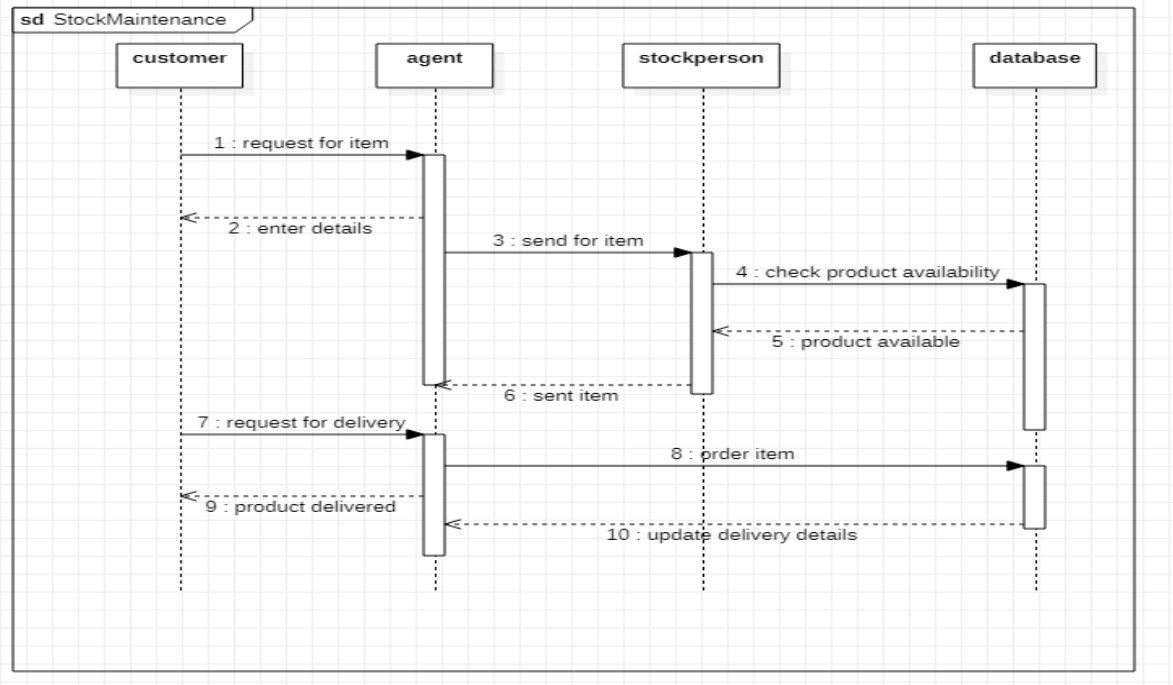


Advanced use case diagram:

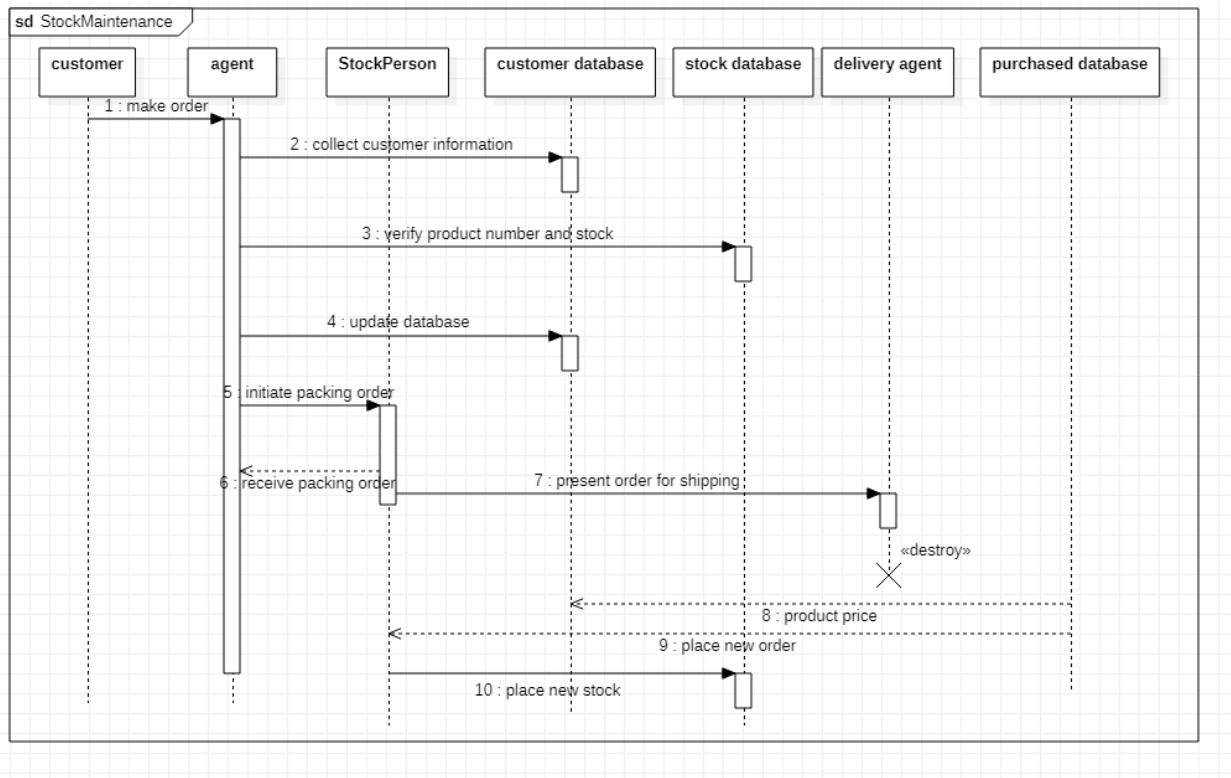


3.6 Sequence diagram

Simple sequence diagram:

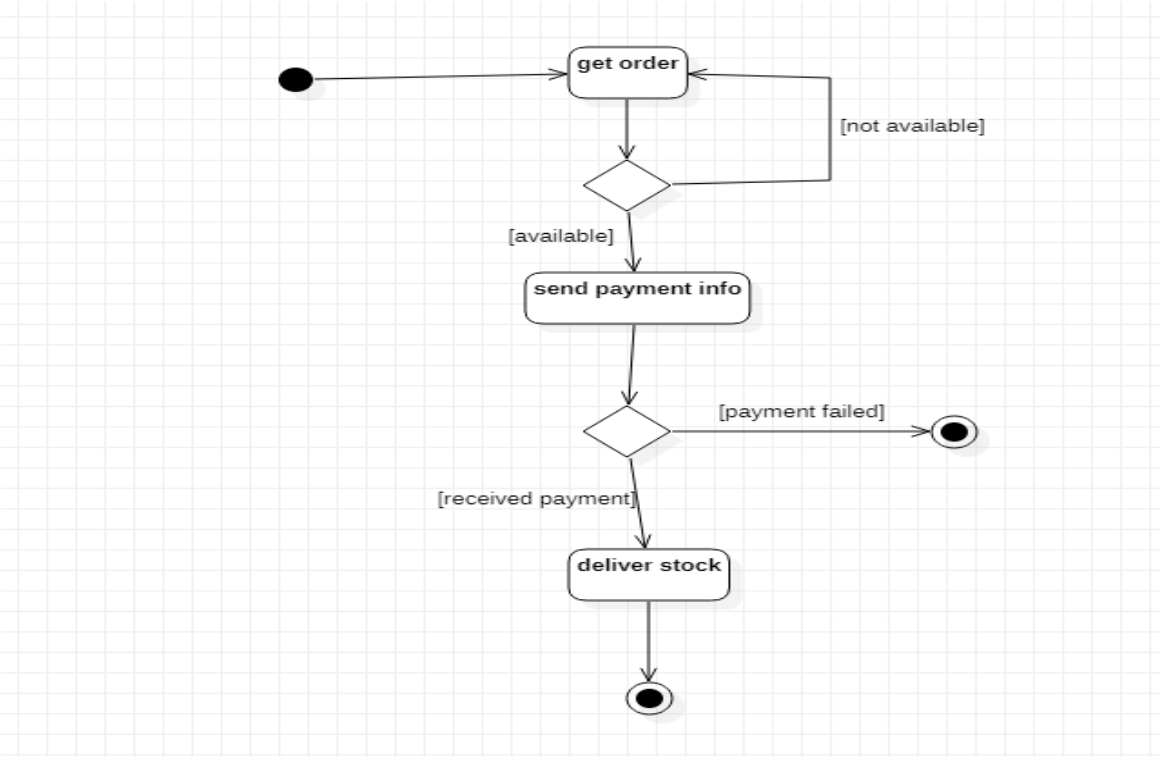


Advanced sequence diagram:

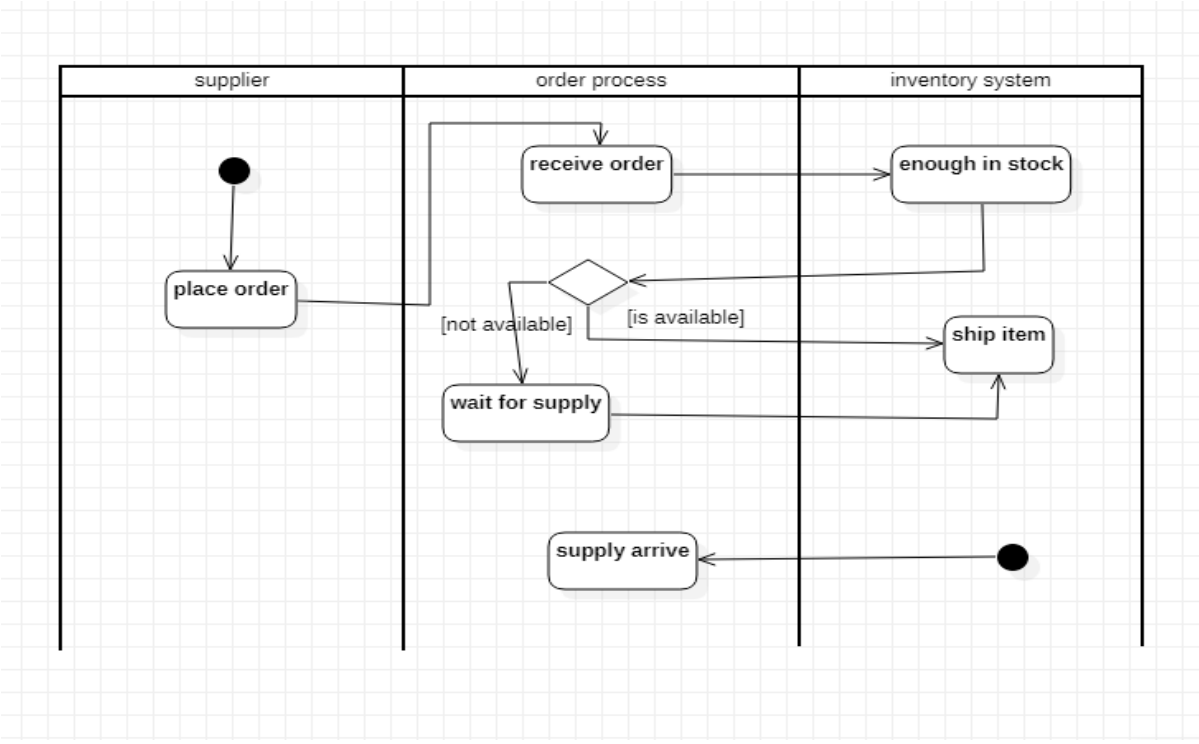


3.7 Activity diagram

Simple activity diagram:



Advanced activity diagram:



4. Coffee Vending Machine

4.1 Problem Statement

To build an efficient coffee vending machine system to provide automated and easy availability of coffee to the customers.

4.2 Software Requirements Specification

- Purpose

To automate the Coffee Vending System. We are attempting to improve our existing system that runs manually.

- Scope

It is more efficient and convenient for the customers. It reduces the manpower needed to perform the entire coffee vending tasks. Since all the work is done by machine, there is less chance of error.

- Functional Requirements

i) Should provide an appropriate display for the customers to avail the purchase

ii) Should support the selection of buttons for different types of coffee

iii) It should have currency detector and money dispenser

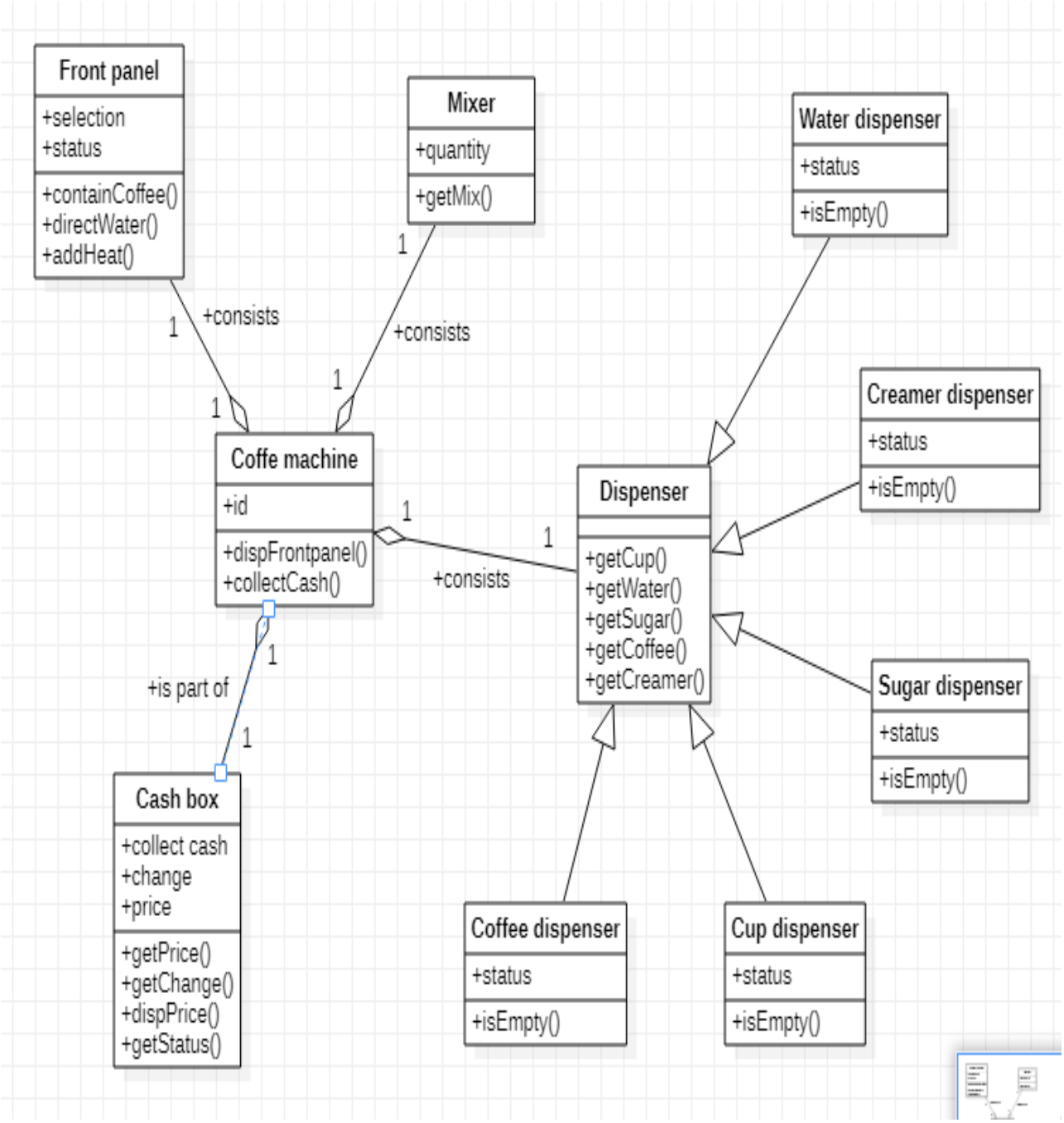
- Non-Functional Requirements

i) The system should be easy to handle

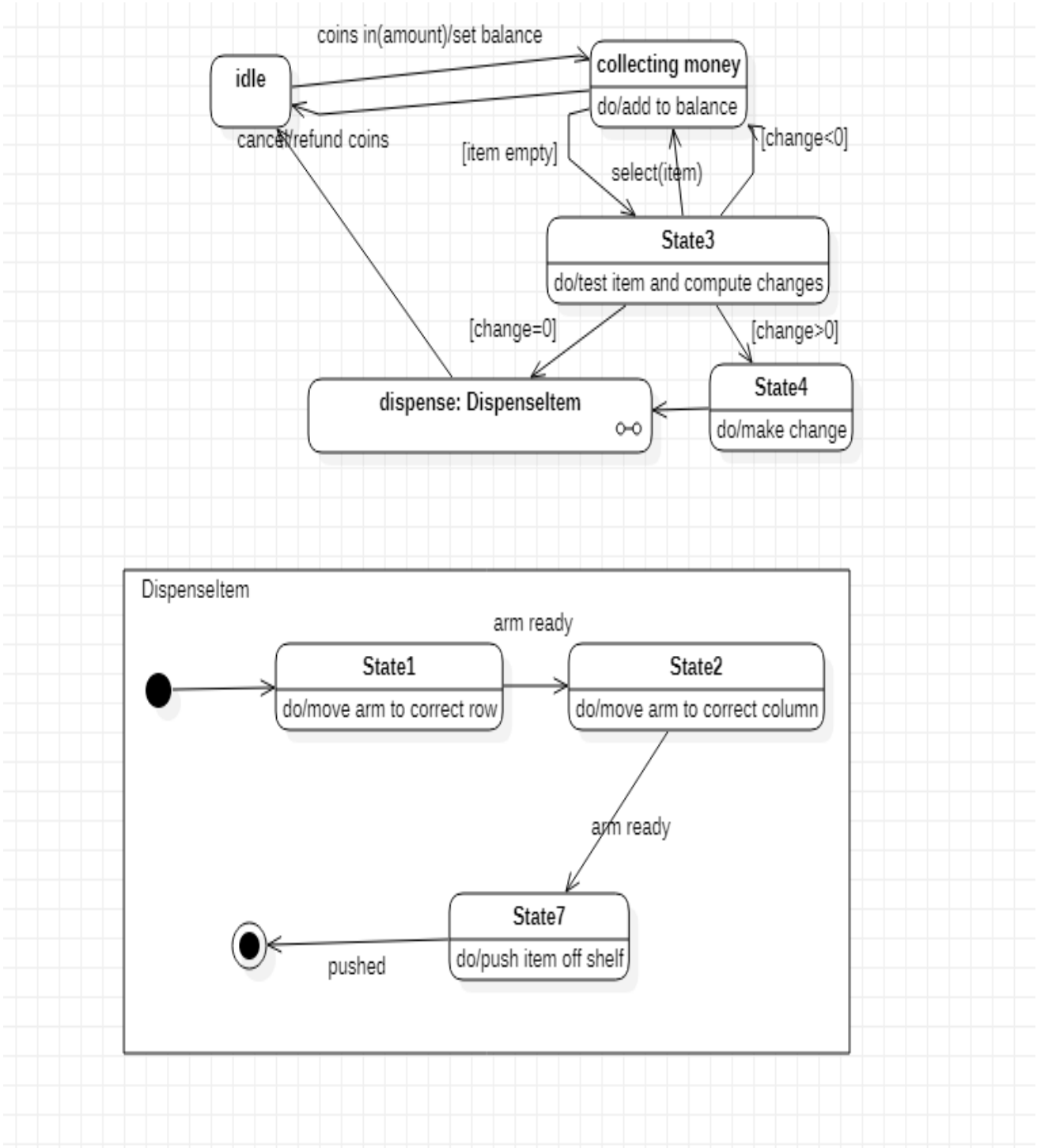
ii) System should give expected performance result

iii) Response time should be less

4.3 Class diagram



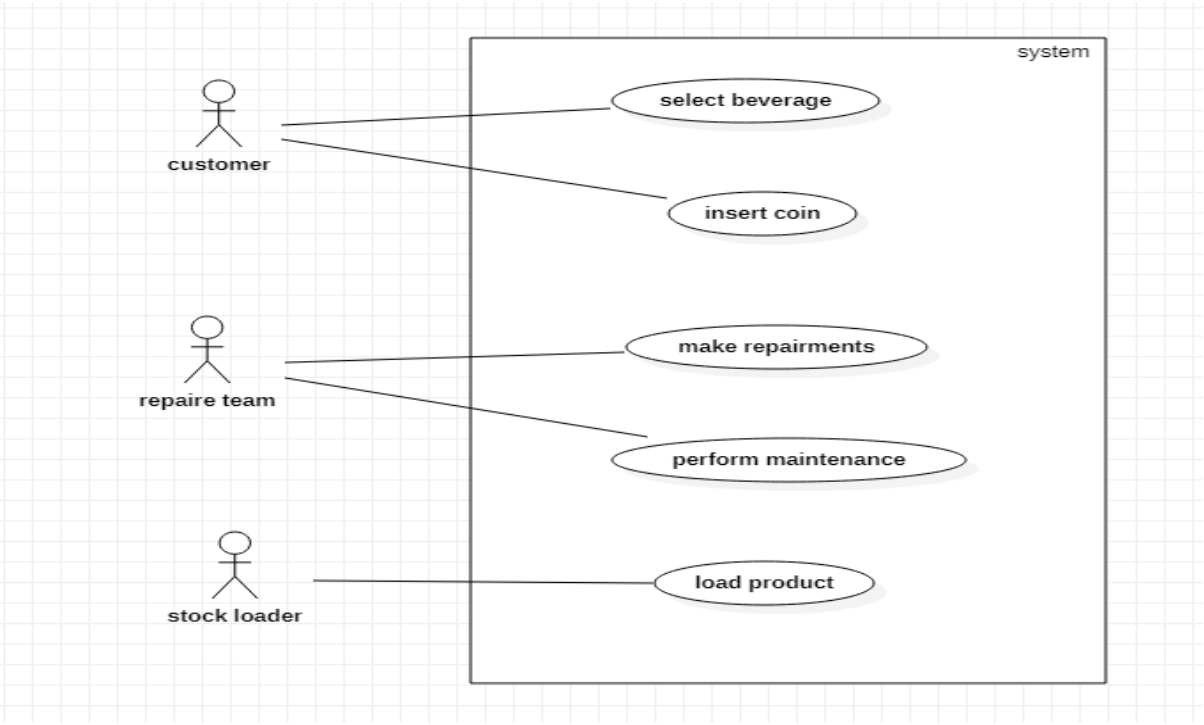
4.4 State diagram



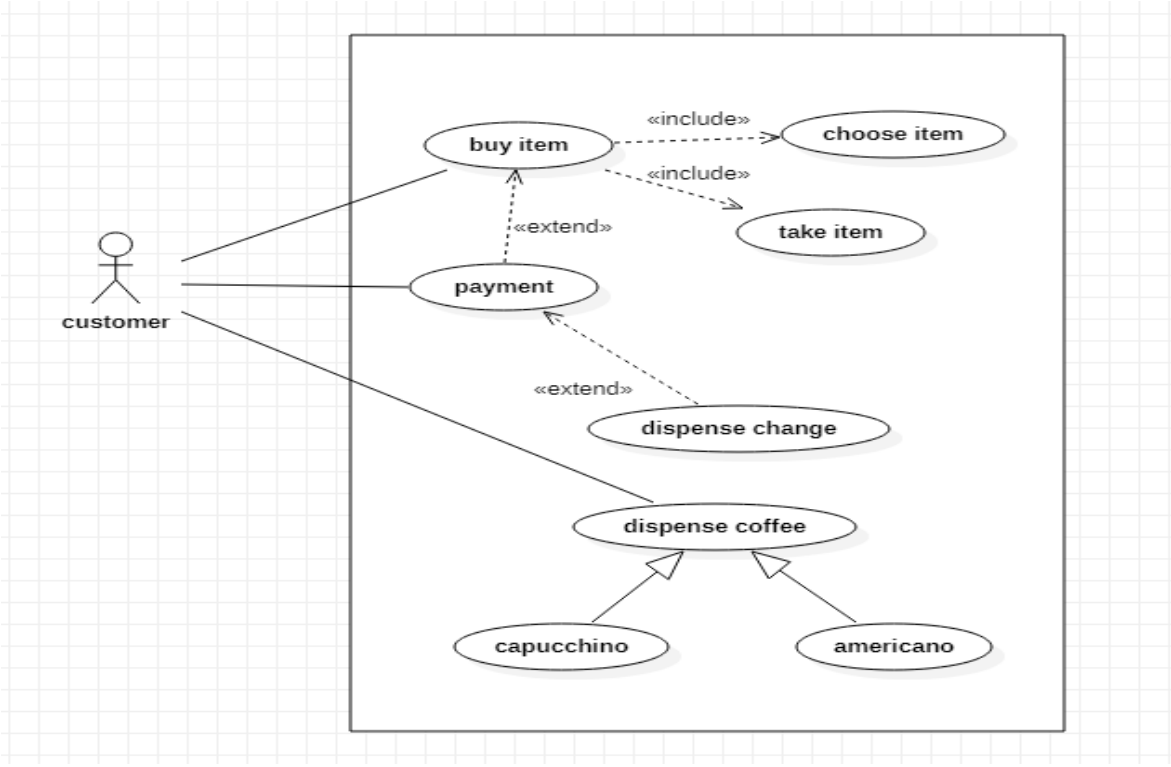
4.5 Use case diagram

Simple use case

diagram:

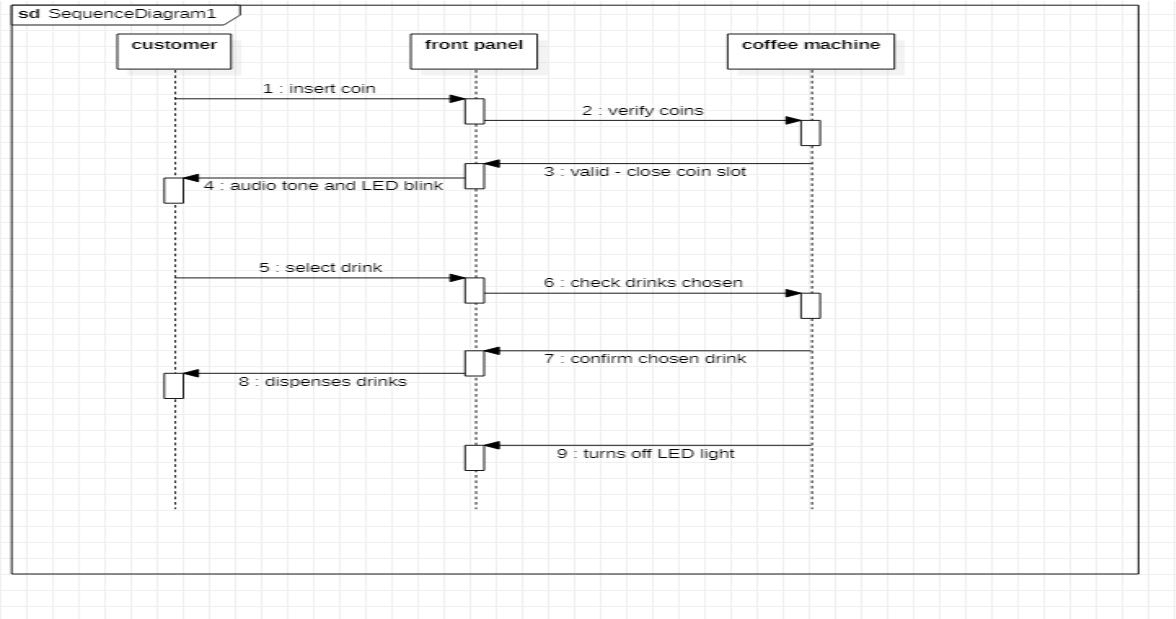


Advanced use case diagram:

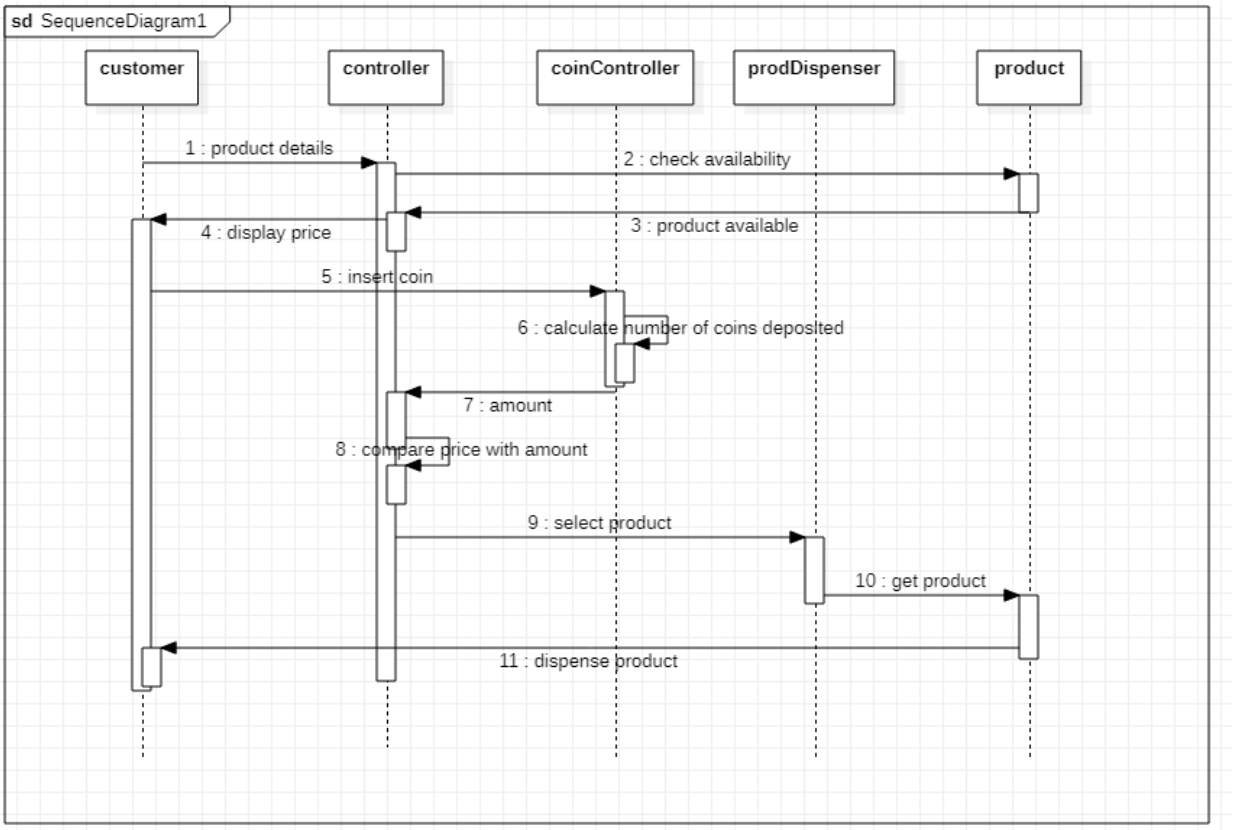


4.6 Sequence diagram

Simple sequence diagram:

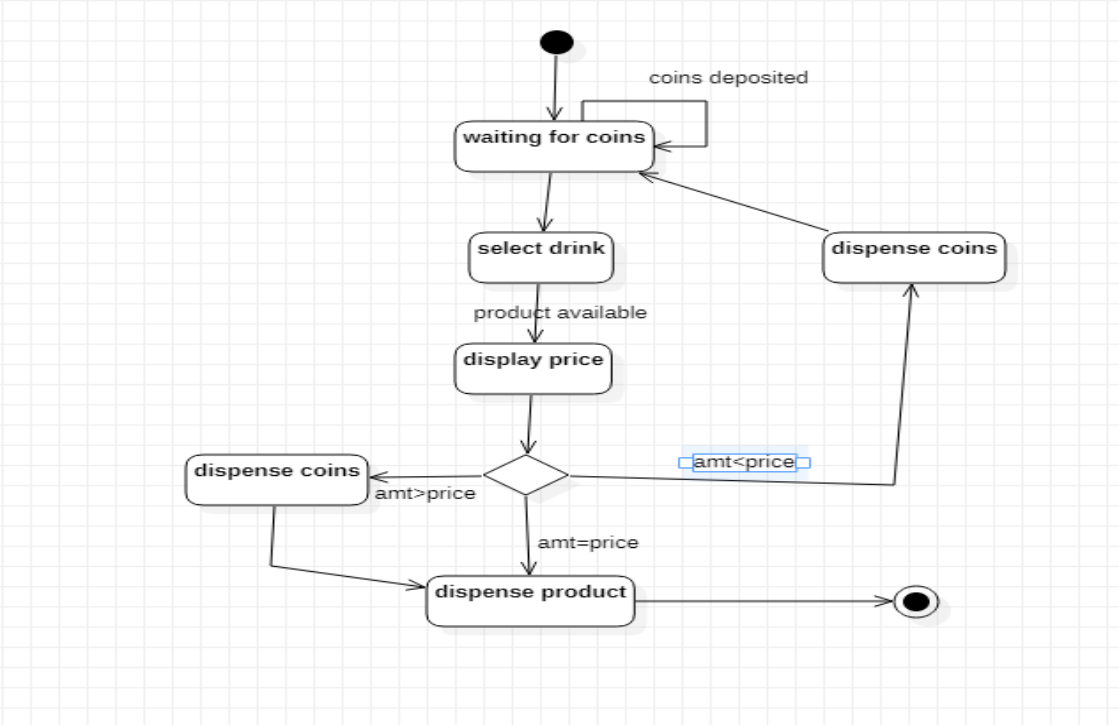


Advanced sequence diagram:

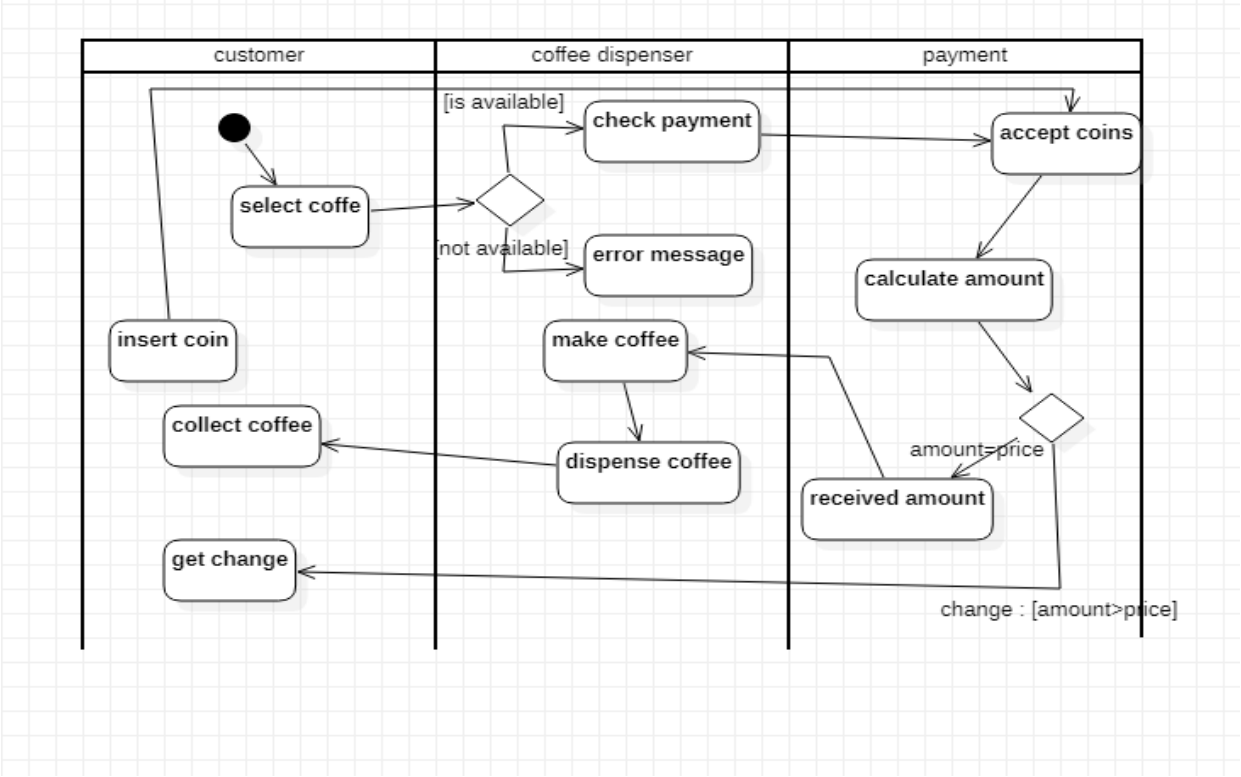


4.7 Activity diagram

Simple activity diagram:



Advanced activity diagram:



5. Online Shopping System

5.1 Problem Statement

To build an efficient online shopping system to provide automated and easy availability of items to the customers.

5.2 Software Requirements Specification

- Purpose

To automate the Online Shopping System. We are attempting to improve our existing system that runs manually.

- Scope

It is more efficient and convenient for the customers. It reduces the manpower needed to perform the entire shopping tasks. Since all the work is done by machine, there is less chance of error.

- Functional Requirements

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

ii) Provides information about the products in categories

iii) Provides email facility for future correspondence

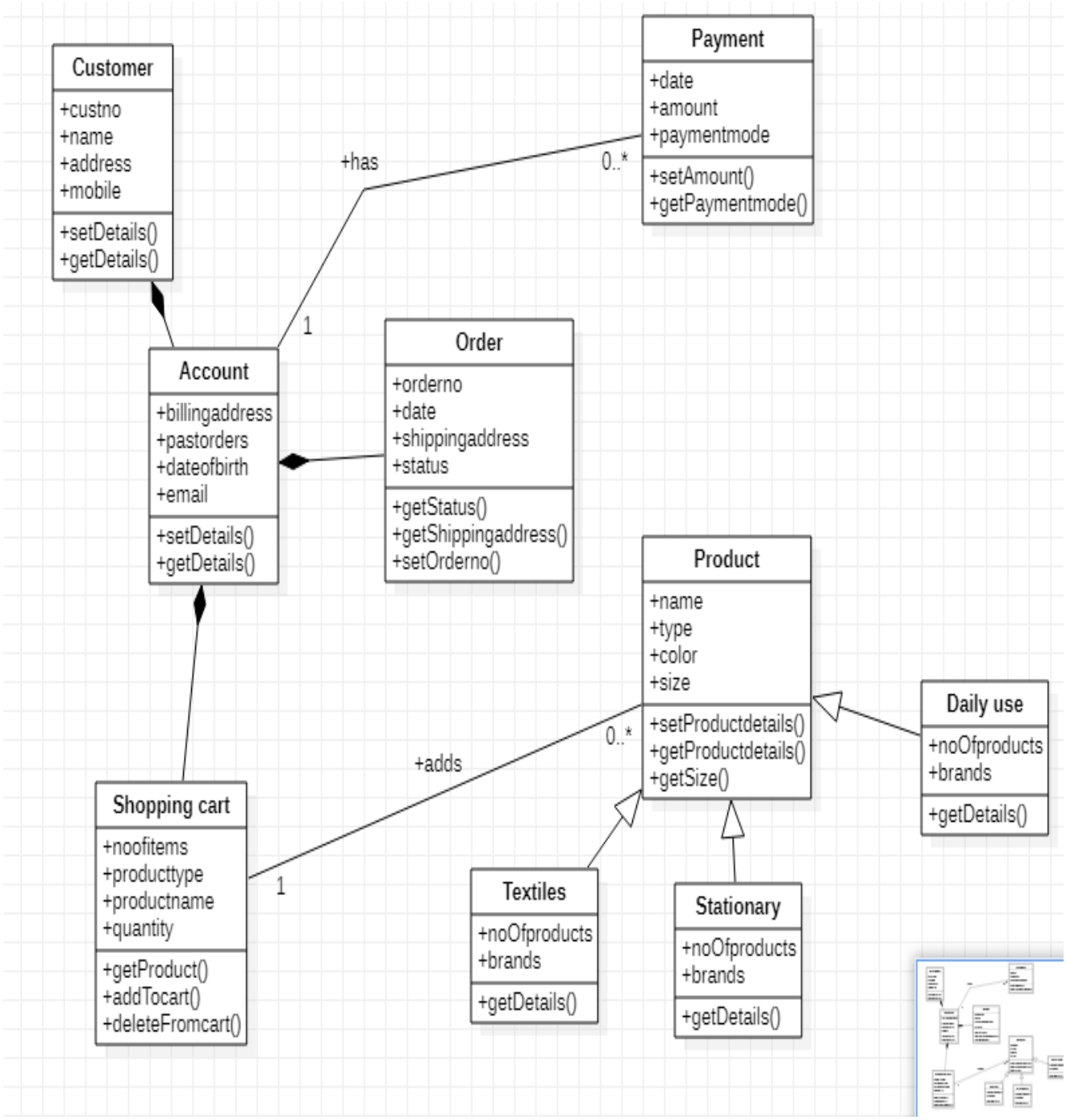
- Non-Functional Requirements

i) The system should be easy to handle

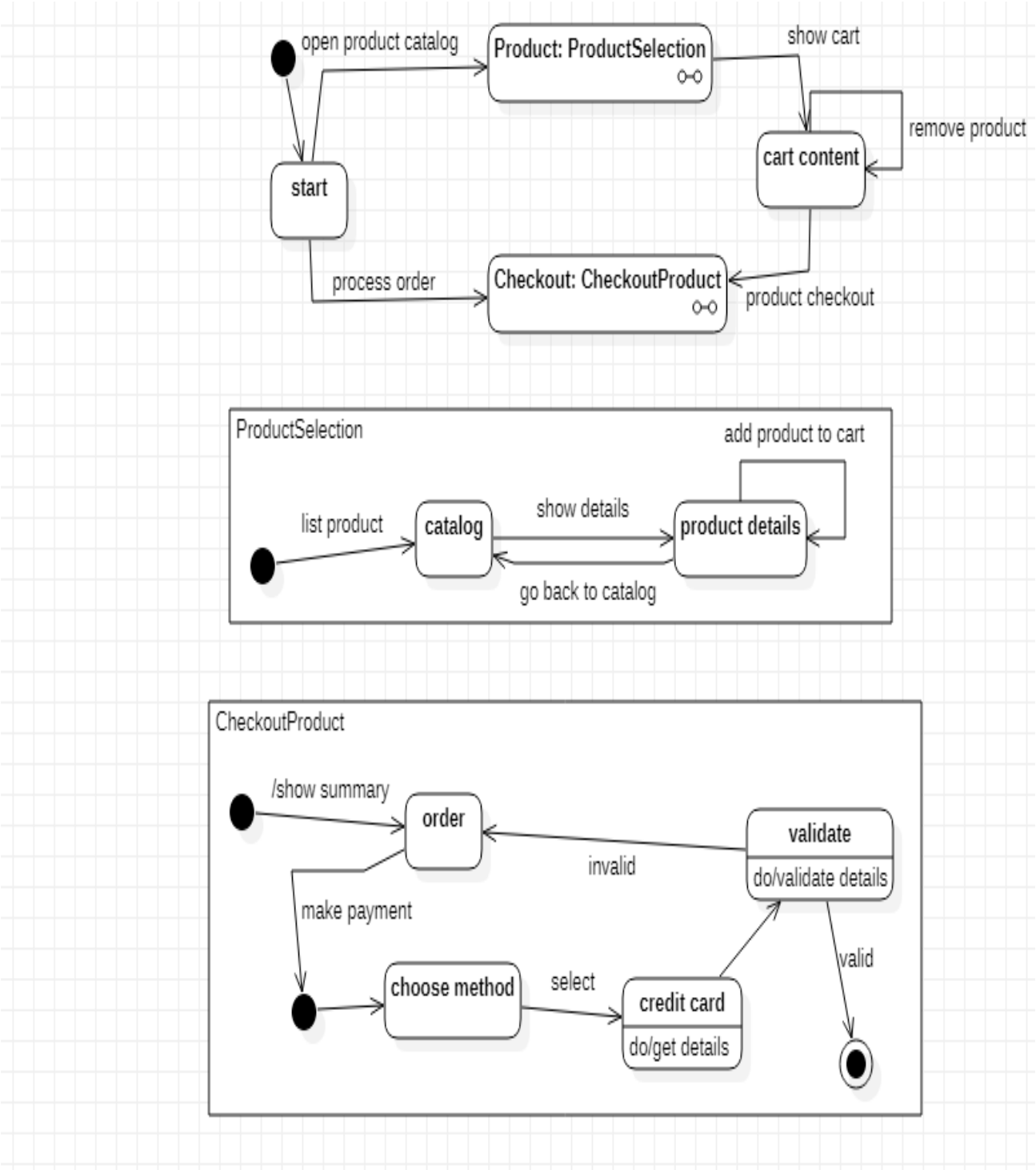
ii) System should give expected performance result

iii) Response time should be less.

5.3 Class diagram



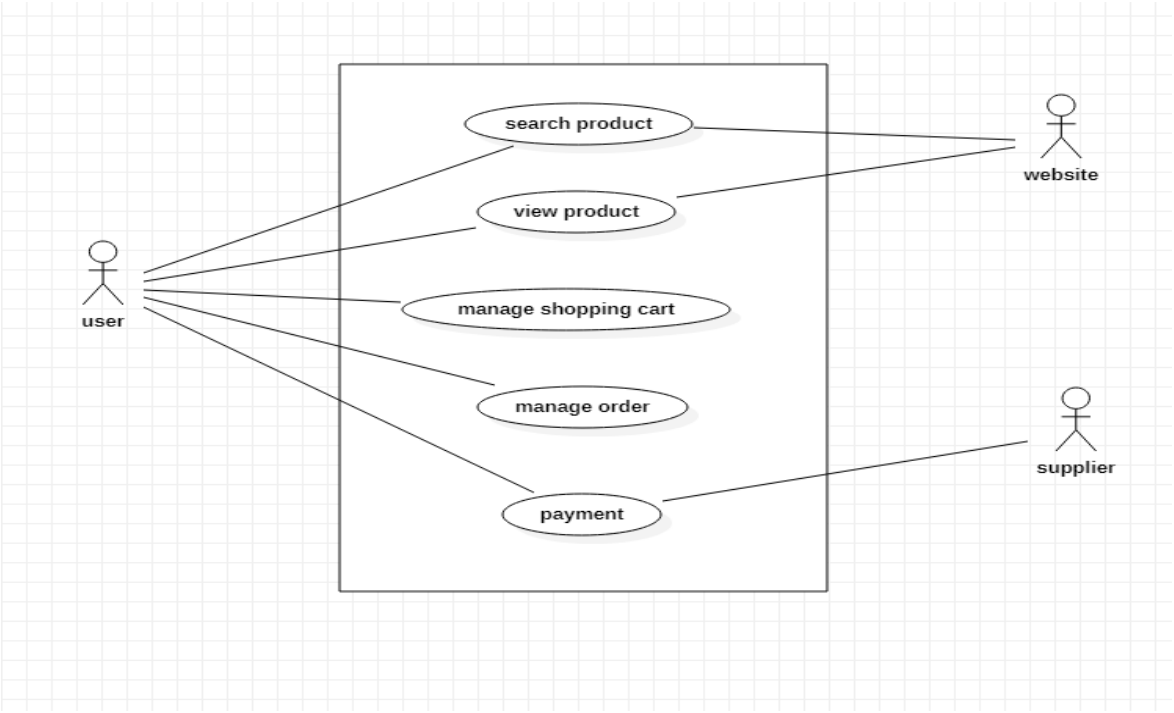
5.4 State diagram



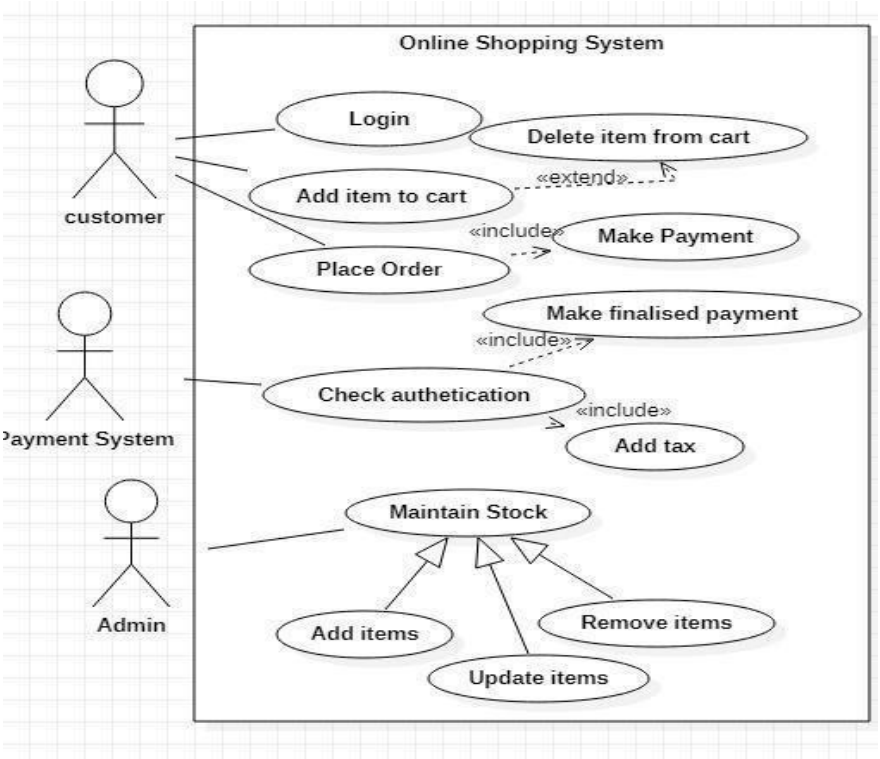
5.5 Use case diagram

Simple use case

diagram:

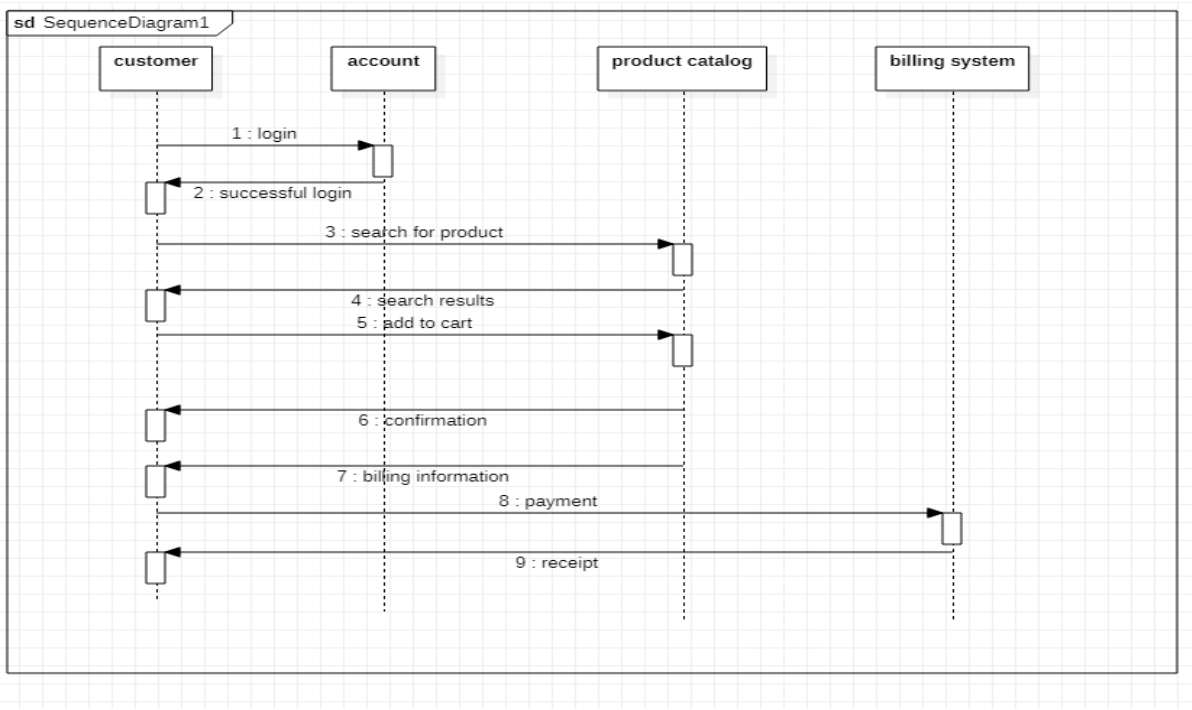


Advanced use case diagram:

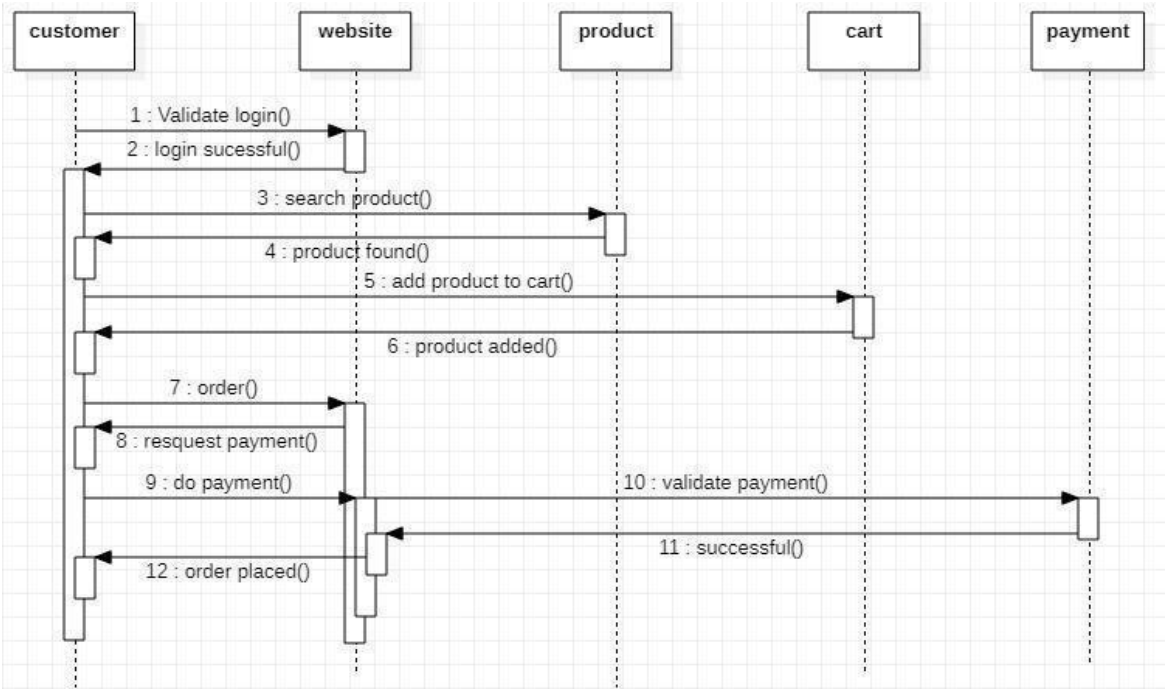


5.6 Sequence diagram

Simple sequence diagram:

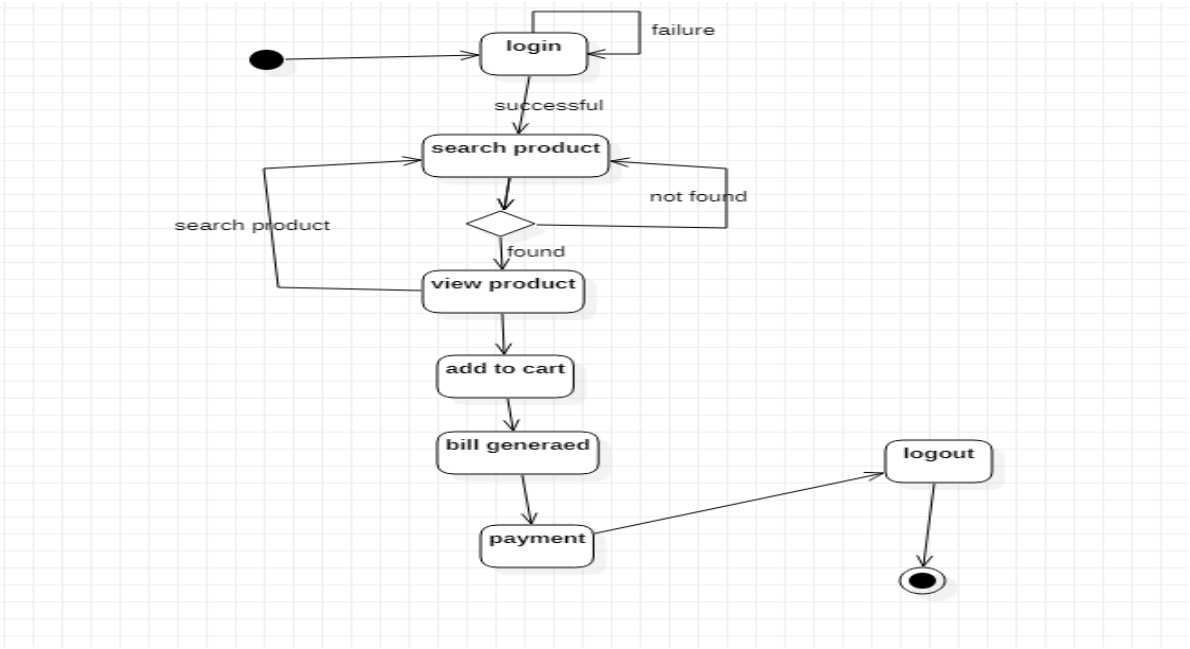


Advanced sequence diagram:

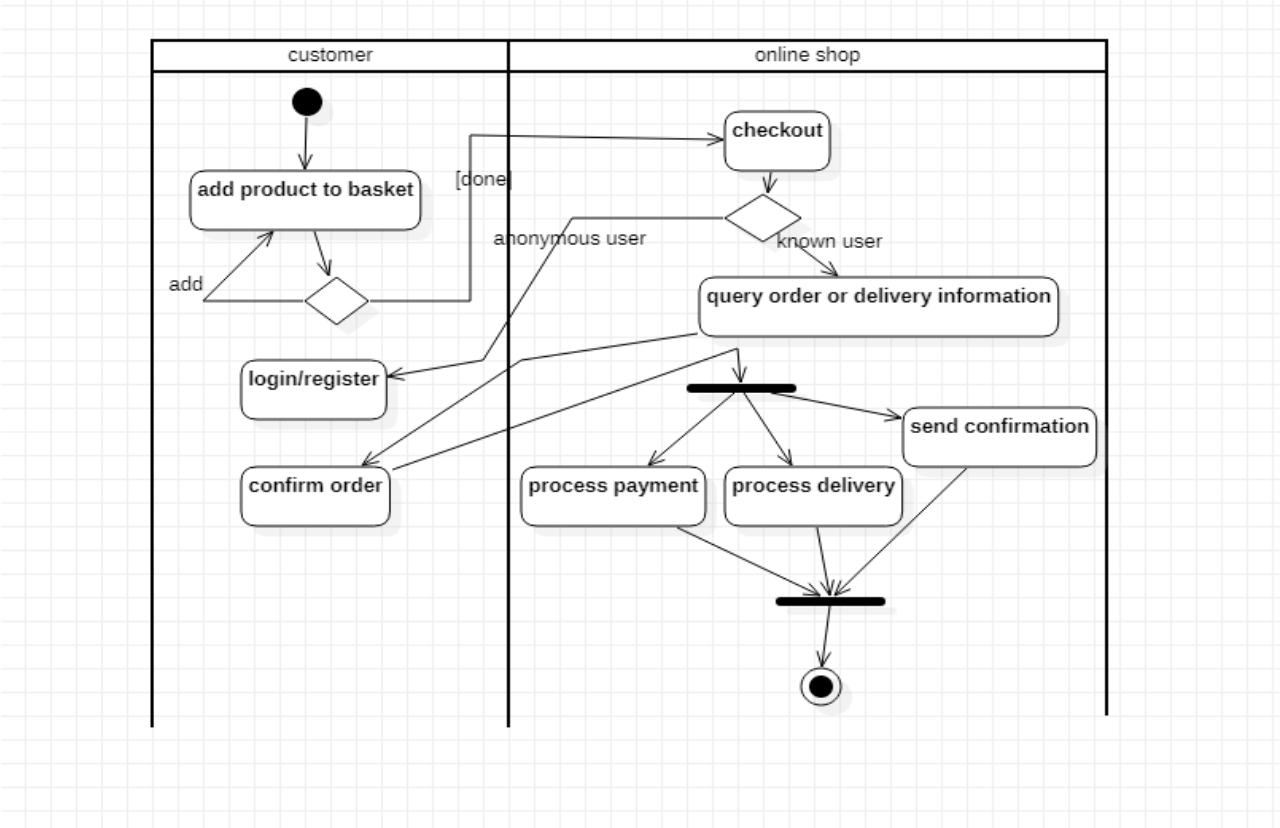


5.7 Activity diagram

Simple activity diagram:



Advanced activity diagram:



6. Railway reservation system

6.1 Problem Statement

To build an efficient railway reservation system to provide automated and easy availability of tickets to the customers.

6.2 Software Requirements Specification

- Purpose

To automate the Railway Reservation System. We are attempting to improve our existing system that runs manually.

- Scope

It is more efficient and convenient for the customers. It reduces the manpower needed to perform the entire reservation tasks. Since all the work is done by machine, there is less chance of error.

- Functional Requirements

i) Confirm reservation for seat

ii) Reservation against cancellations

iii) Tatkal reservation

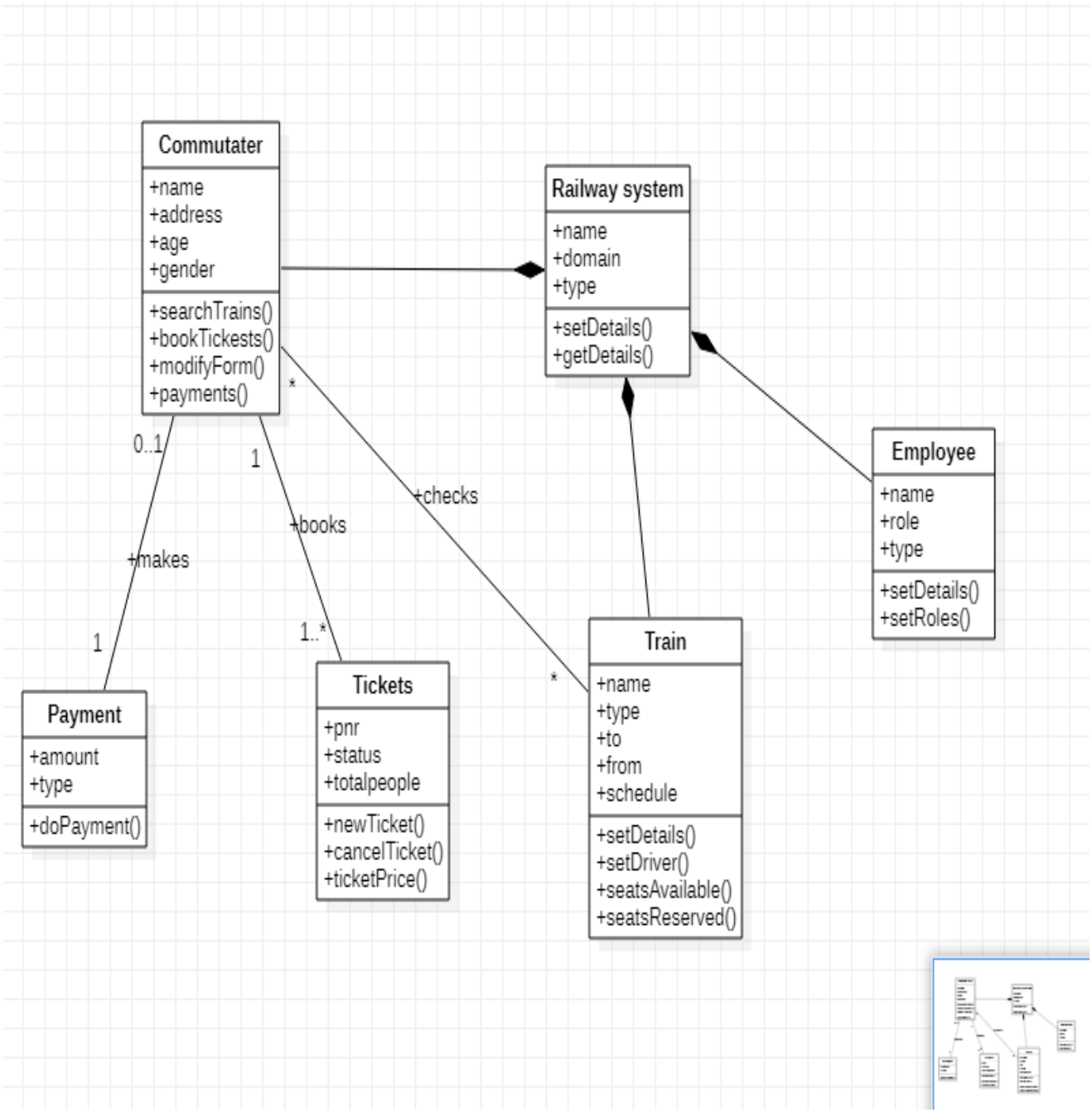
- Non-Functional Requirements

i) The system should be easy to handle

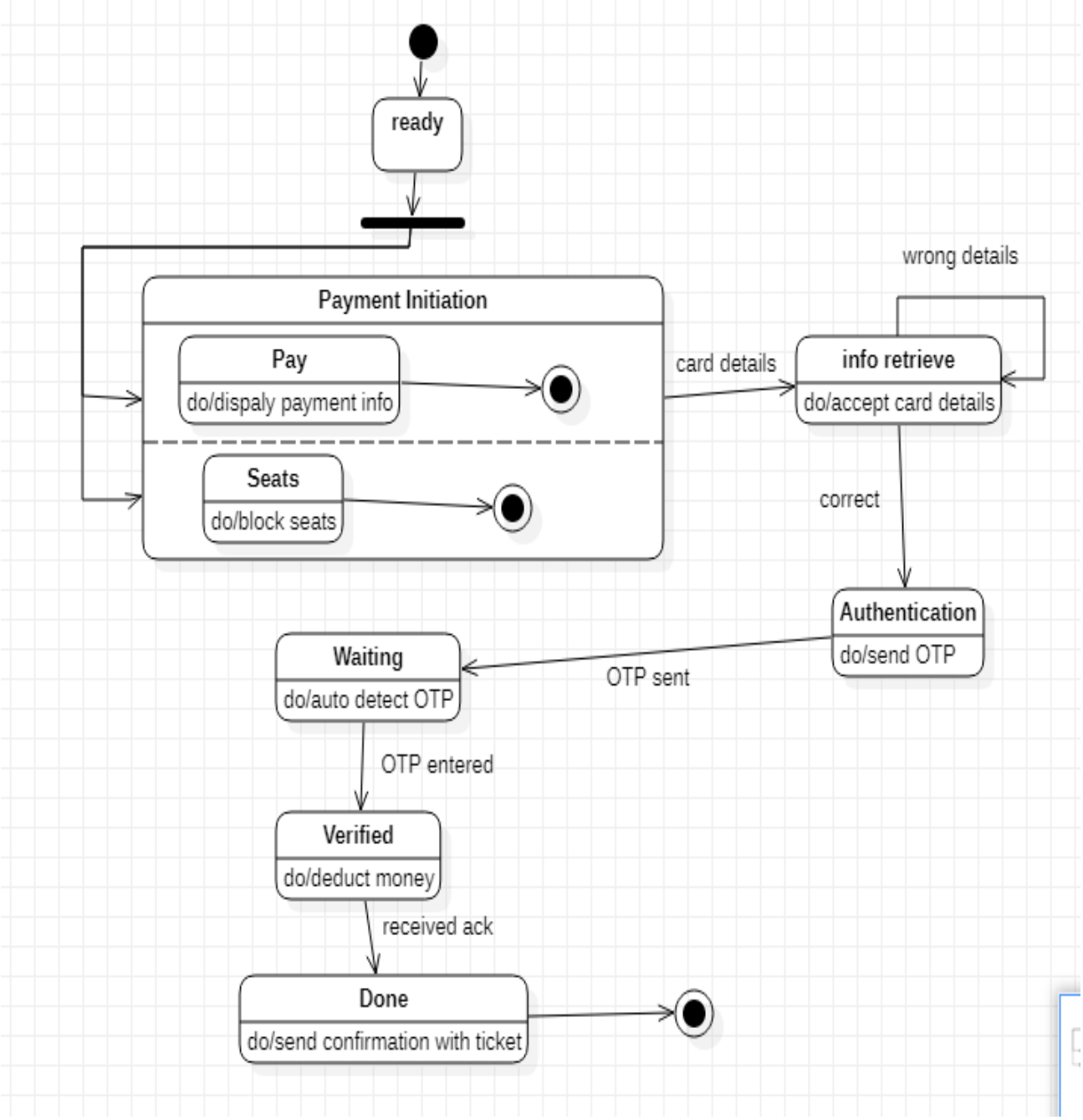
ii) System should give expected performance result

iii) Response time should be less

6.3 Class diagram



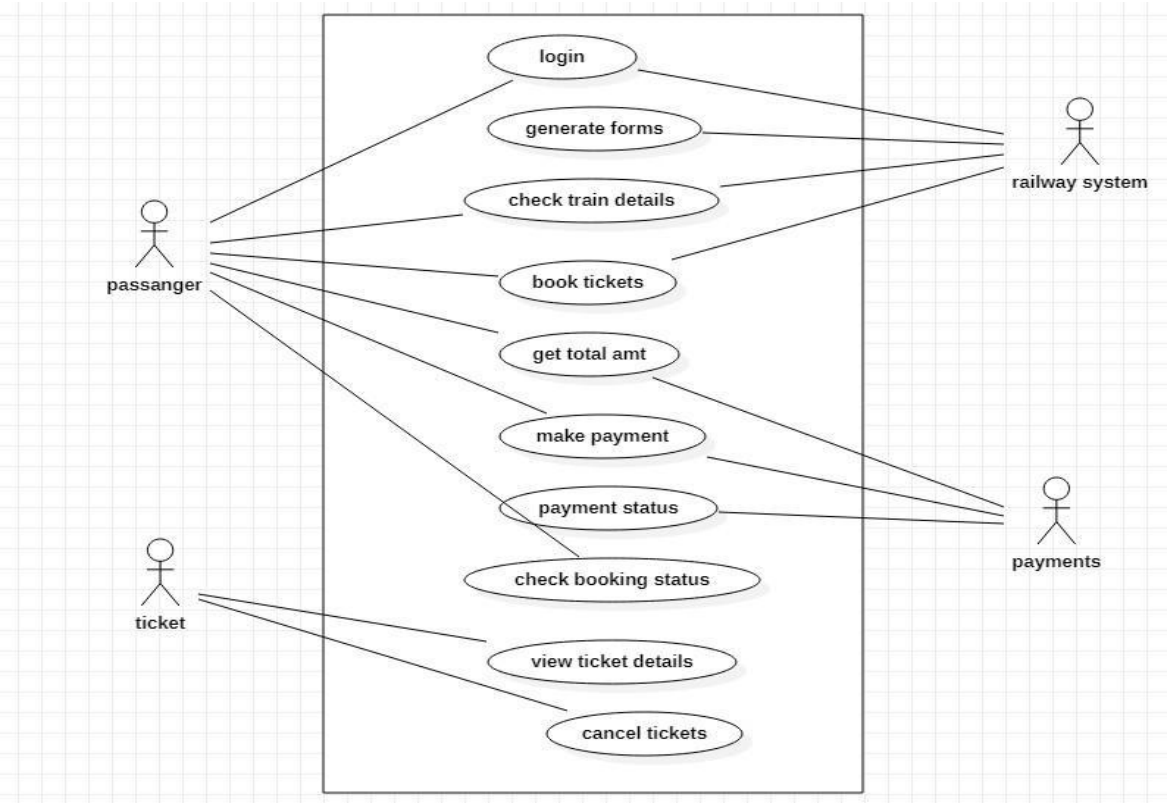
6.4 State diagram



6.5 Use case diagram

Simple use case

diagram:



Advanced use case diagram:

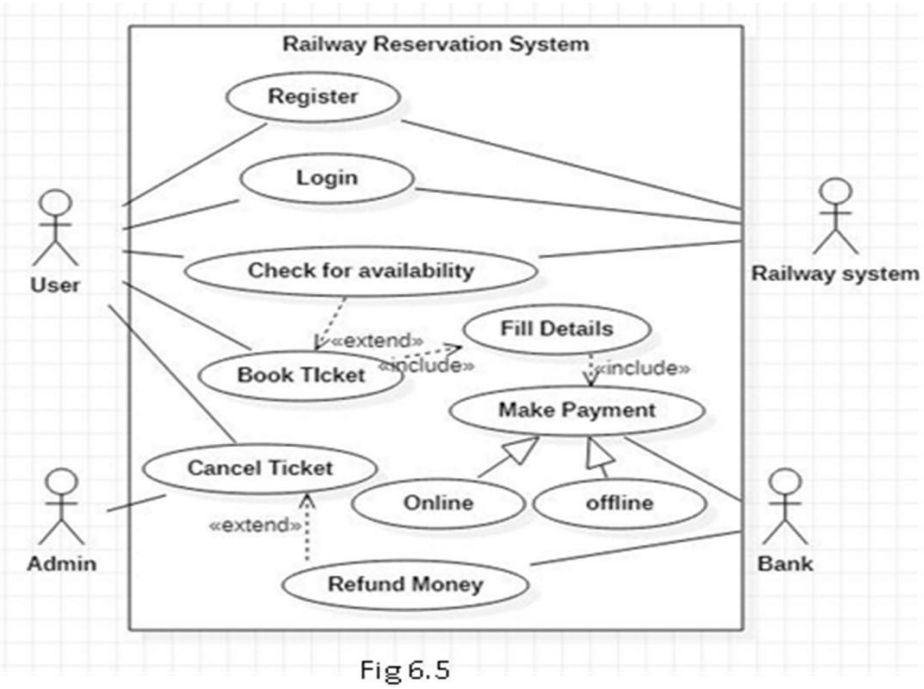
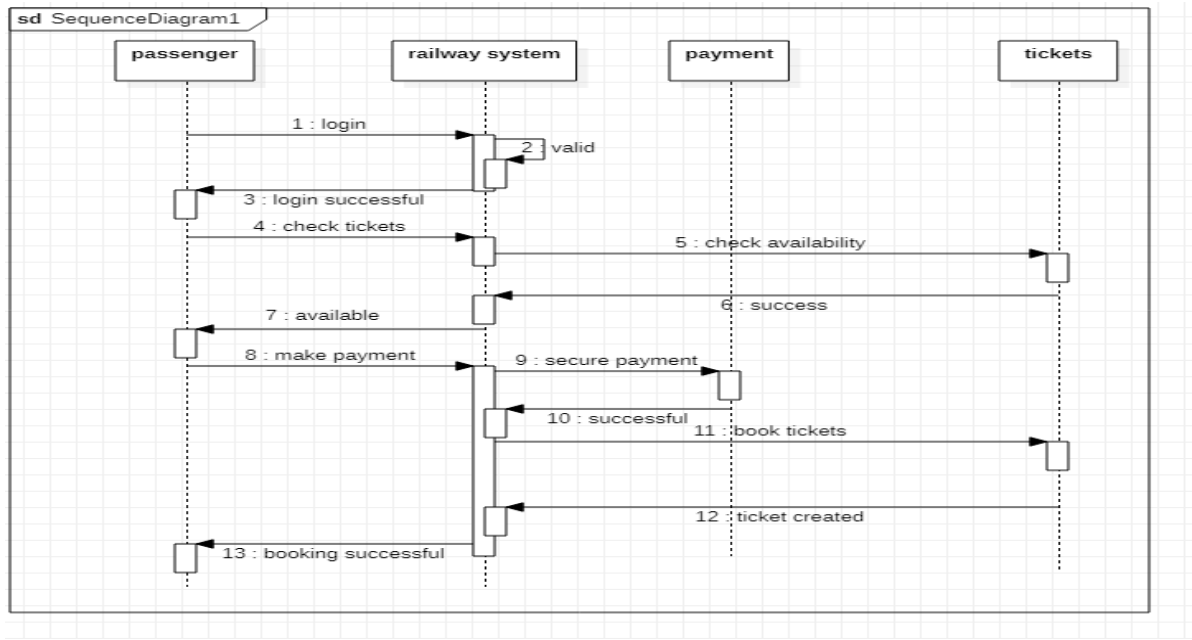


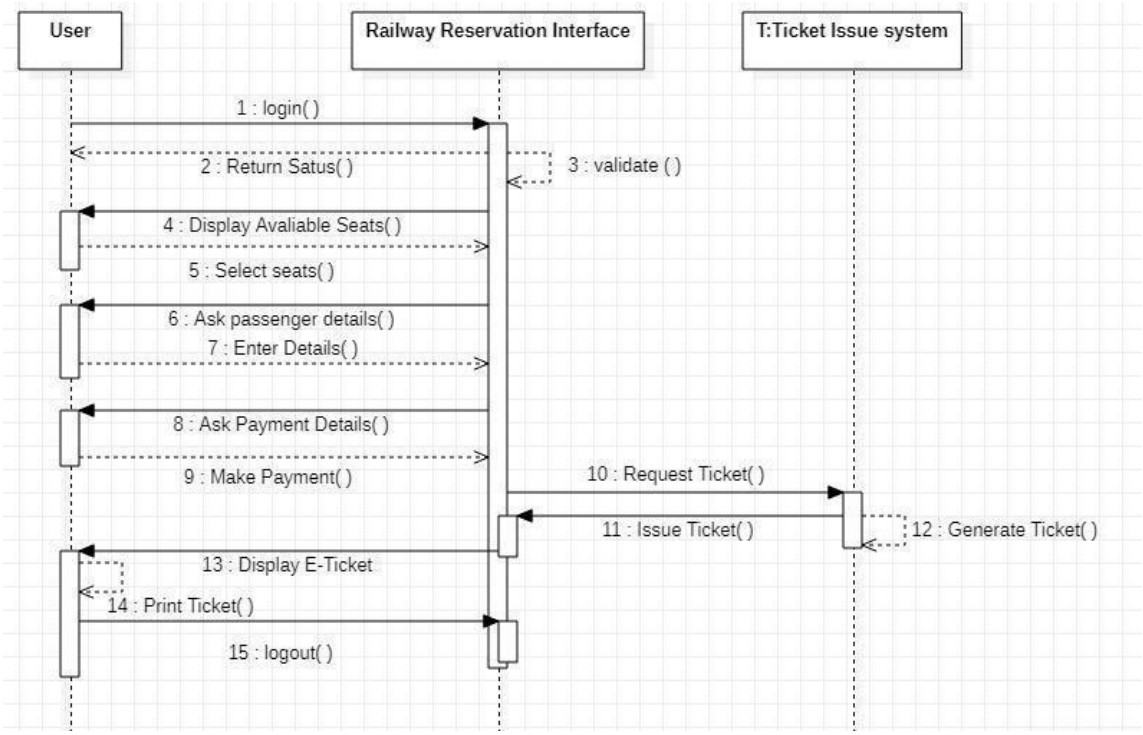
Fig 6.5

6.6 Sequence diagram

Simple sequence diagram:

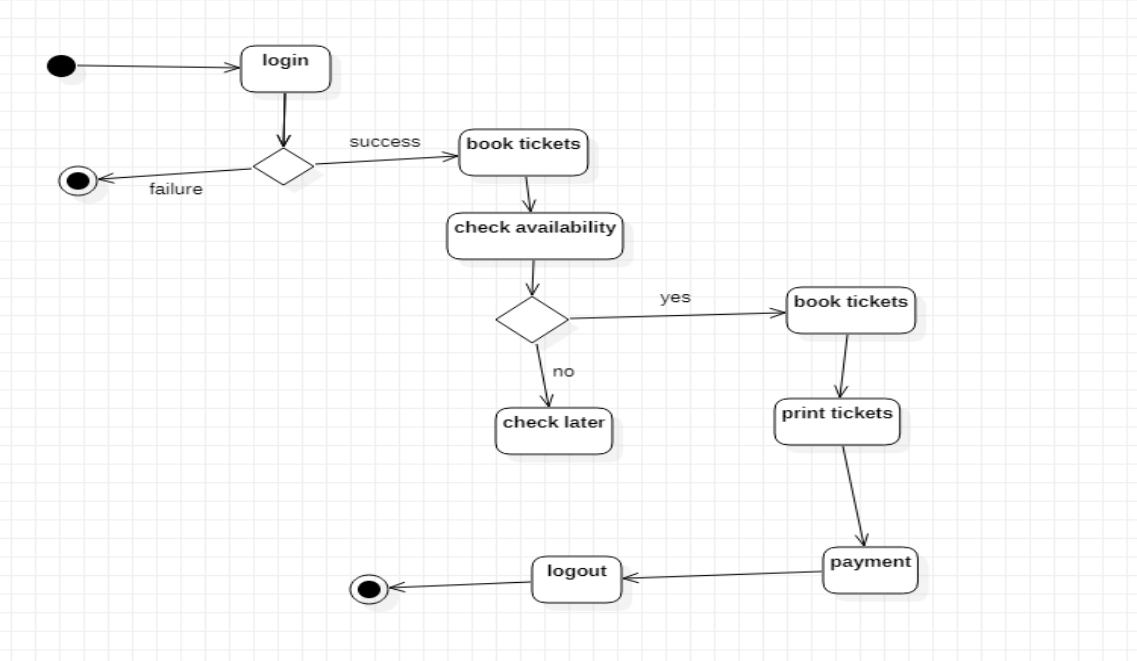


Advanced sequence diagram:

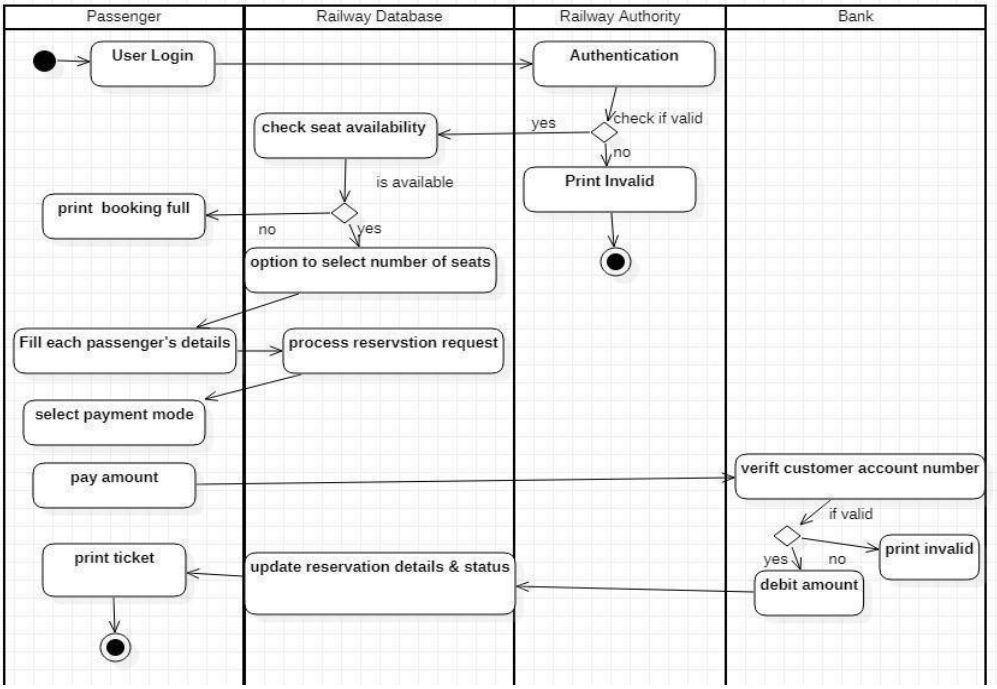


6.7 Activity diagram

Simple activity diagram:



Advanced activity diagram:



7. Graphics Editor

7.1 Problem Statement

To design and develop the Graphics Editor software package to create line drawings involving several types of graphic entities.

7.2 Software Requirements Specification

- Purpose

The entire process of Graphics Editing is done in a manual manner. Considering the fact that the number of customers for purchase is increasing every year, a maintenance system is essential to meet the demand. So this system uses several programming and database techniques to reduce the work involved in this process.

- Scope

The graphics editing system will provides a designing platform between the customer and the editor person.

- Functional Requirements

i) It contains the toolbox which contains tools like line, circle, rectangle, text, eraser, etc

ii) Color box or palette

iii) Standard toolbar with options for New, Open, Save, etc

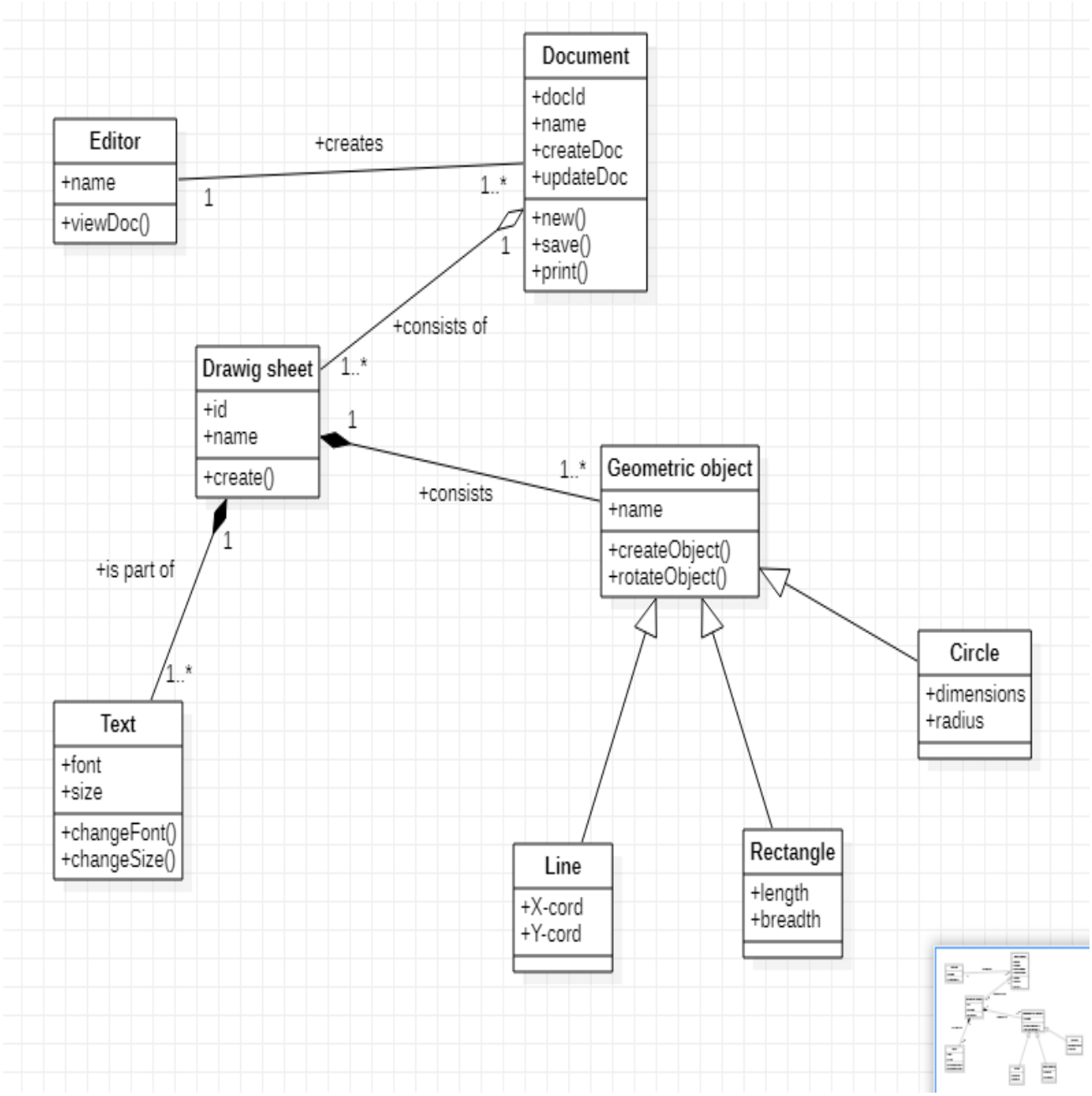
- Non-Functional Requirements

i) Easy handling of tools to users

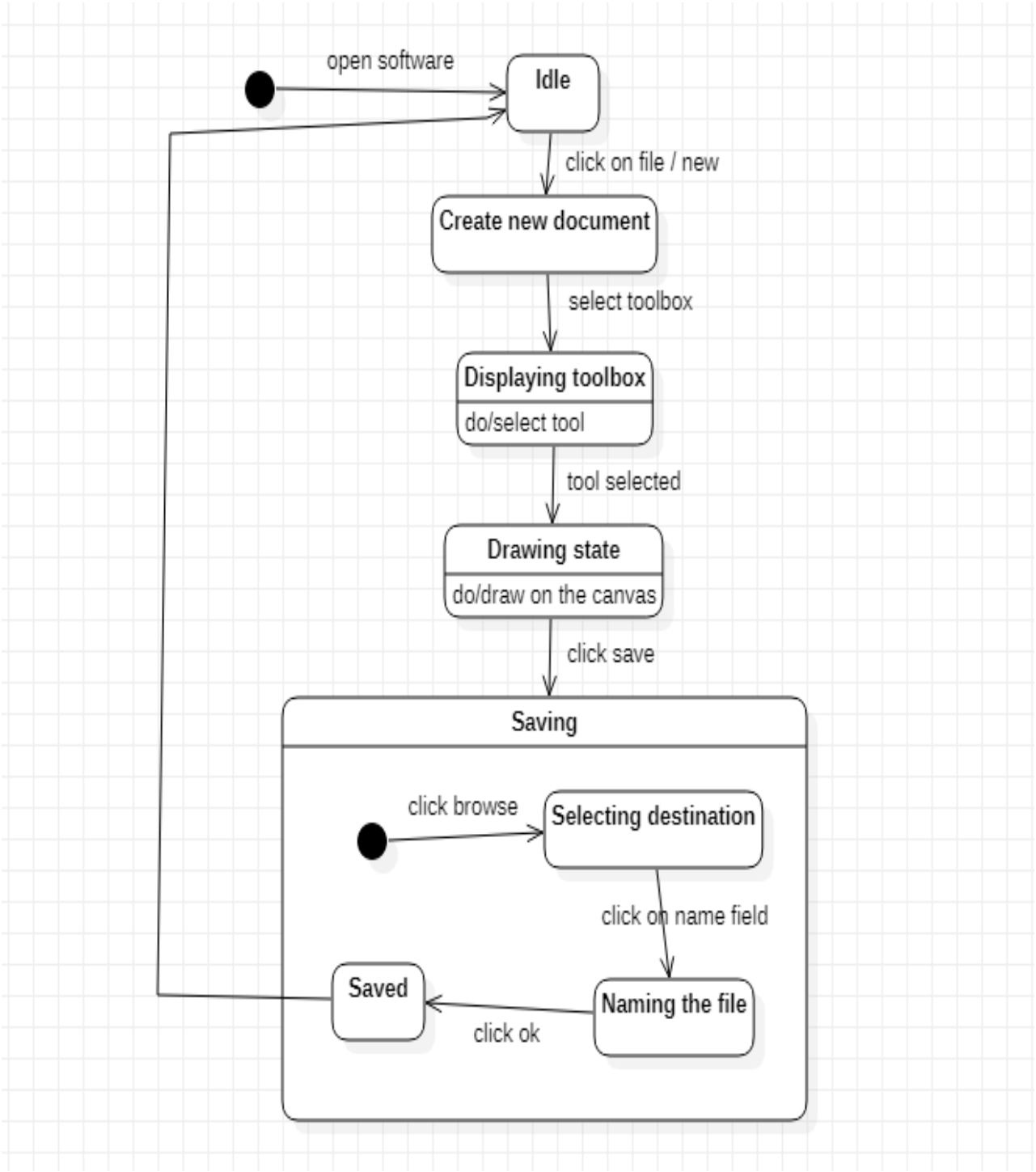
ii) User-friendly

iii) The system shall be easy to migrate or backed up via another use drive.

7.3 Class diagram



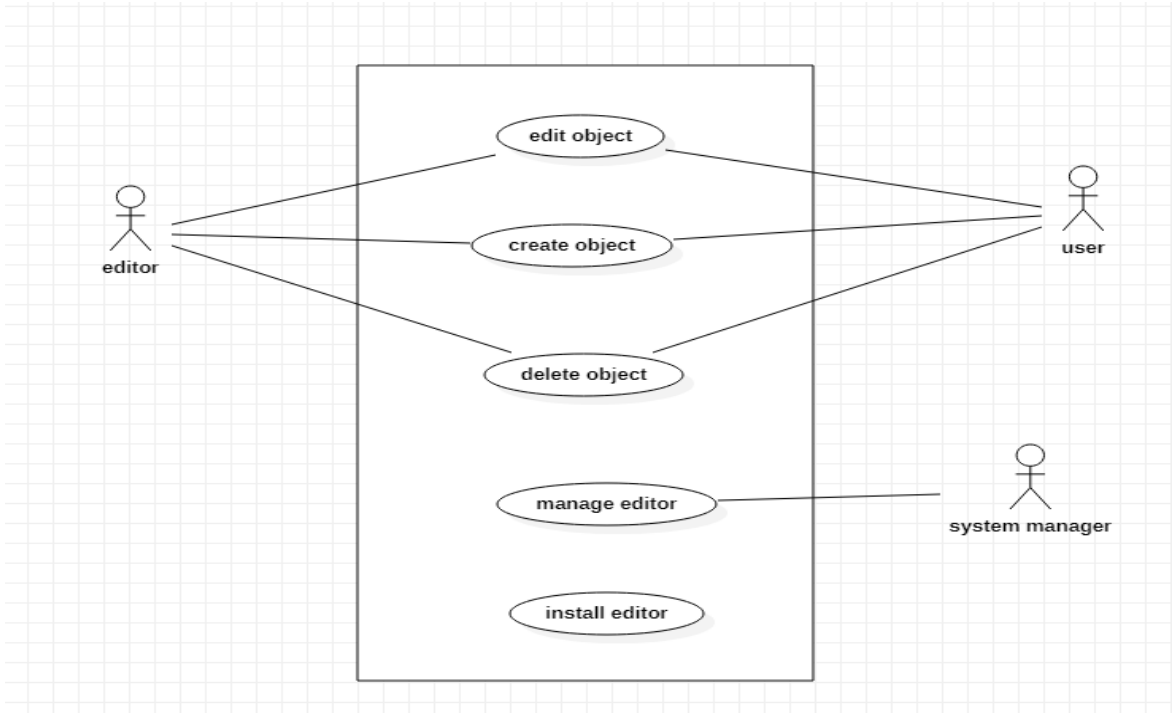
7.4 State diagram



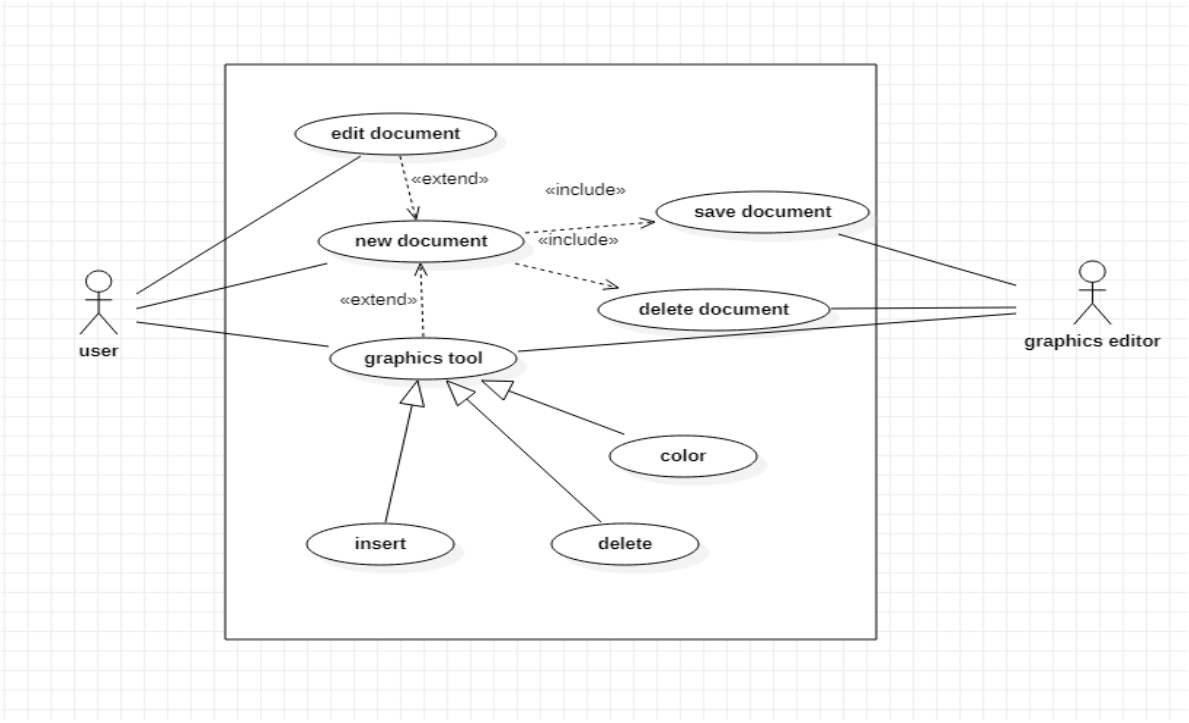
7.5 Use case diagram

Simple use case

diagram:

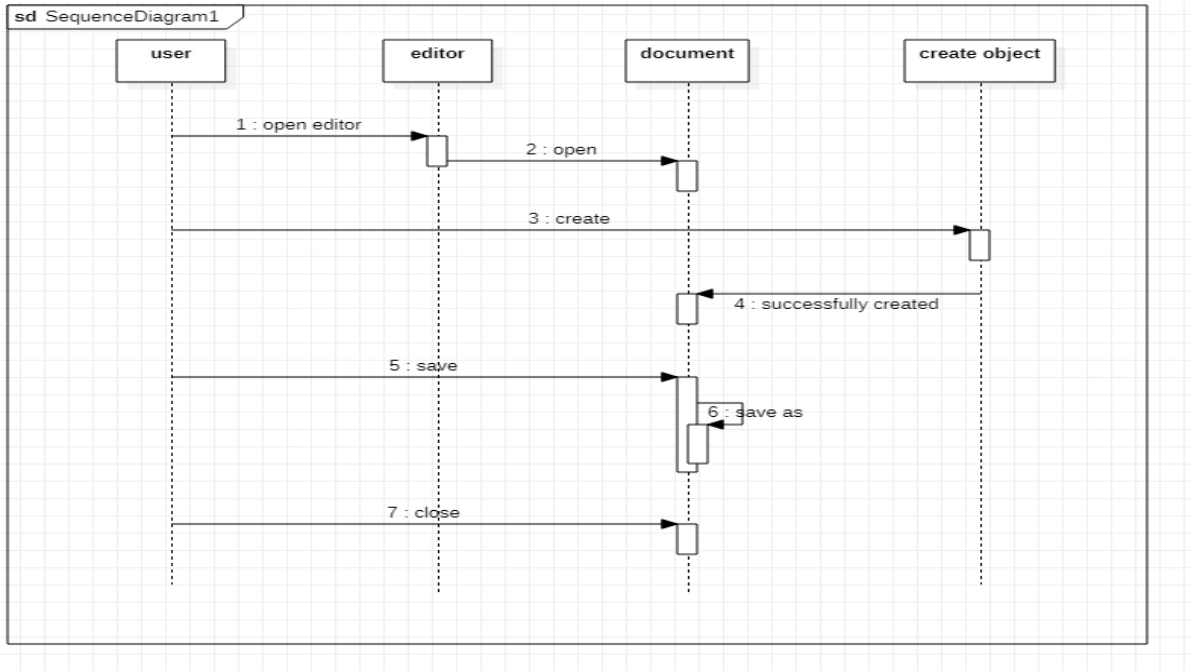


Advanced use case diagram:

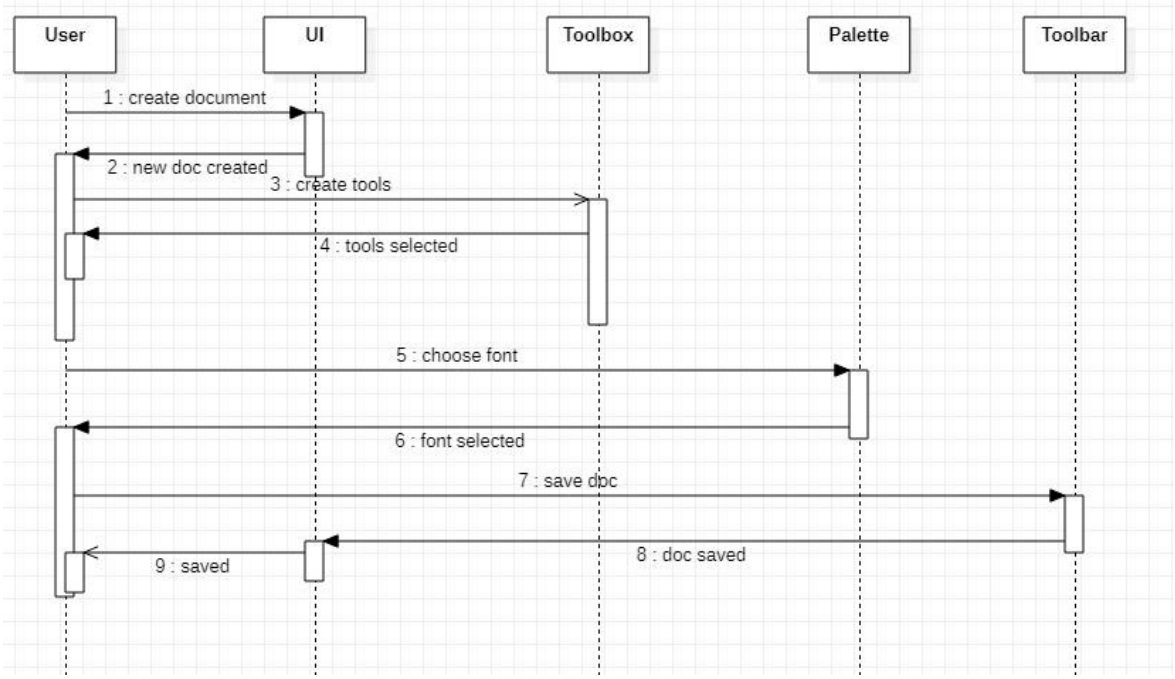


7.6 Sequence diagram

Simple sequence diagram:

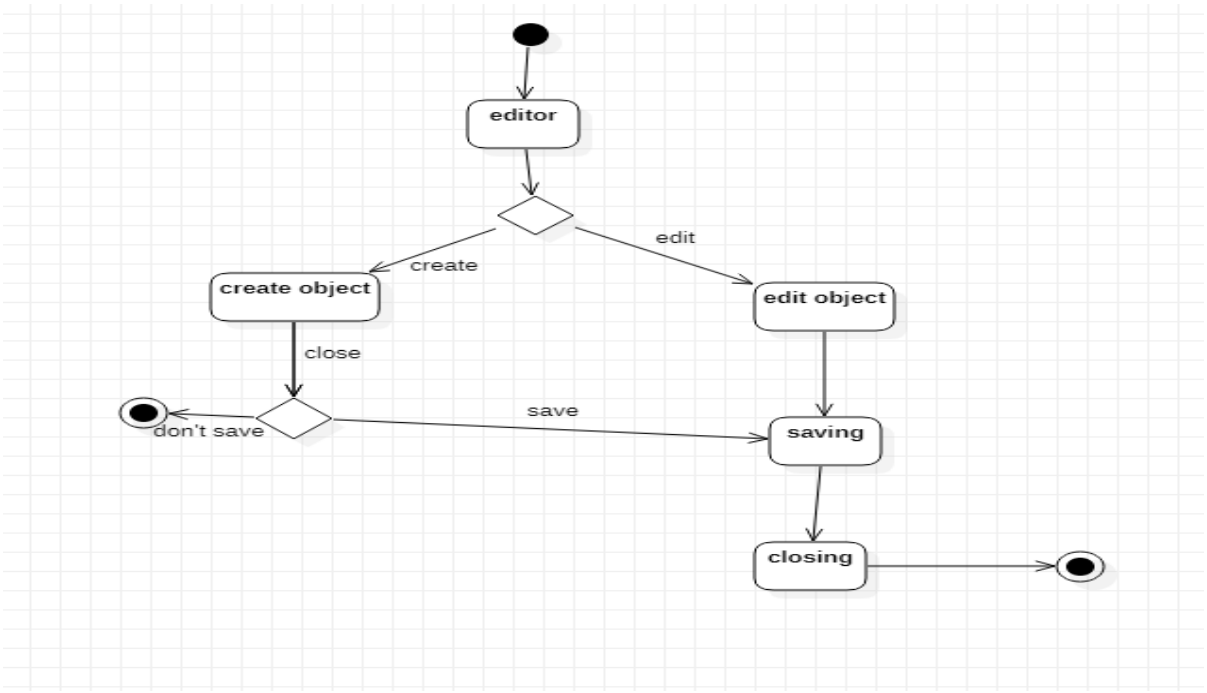


Advanced sequence diagram:



7.7 Activity diagram

Simple activity diagram:



Advanced activity diagram: