SCHOOL OF
COMPUTING

S.LAHARI SAI

CH.SC.U4CSE24143

OBJECT ORIENTED PROGRAMMING (23CSE111)

LAB RECORD

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF COMPUTING, CHENNAI

BONAFIDE CERTIFICATE

This is to certify that the Lab Record work for 23CSE111- Object Oriented Programming Subject submitted by ***CH.SC.U4CSE24113 - S.LAHARI SAI*** in **"Computer Science and Engineering"** is a Bonafide record of the work carried out under my guidance and supervision at Amrita School of Computing, Chennai.

This Lab examination held on

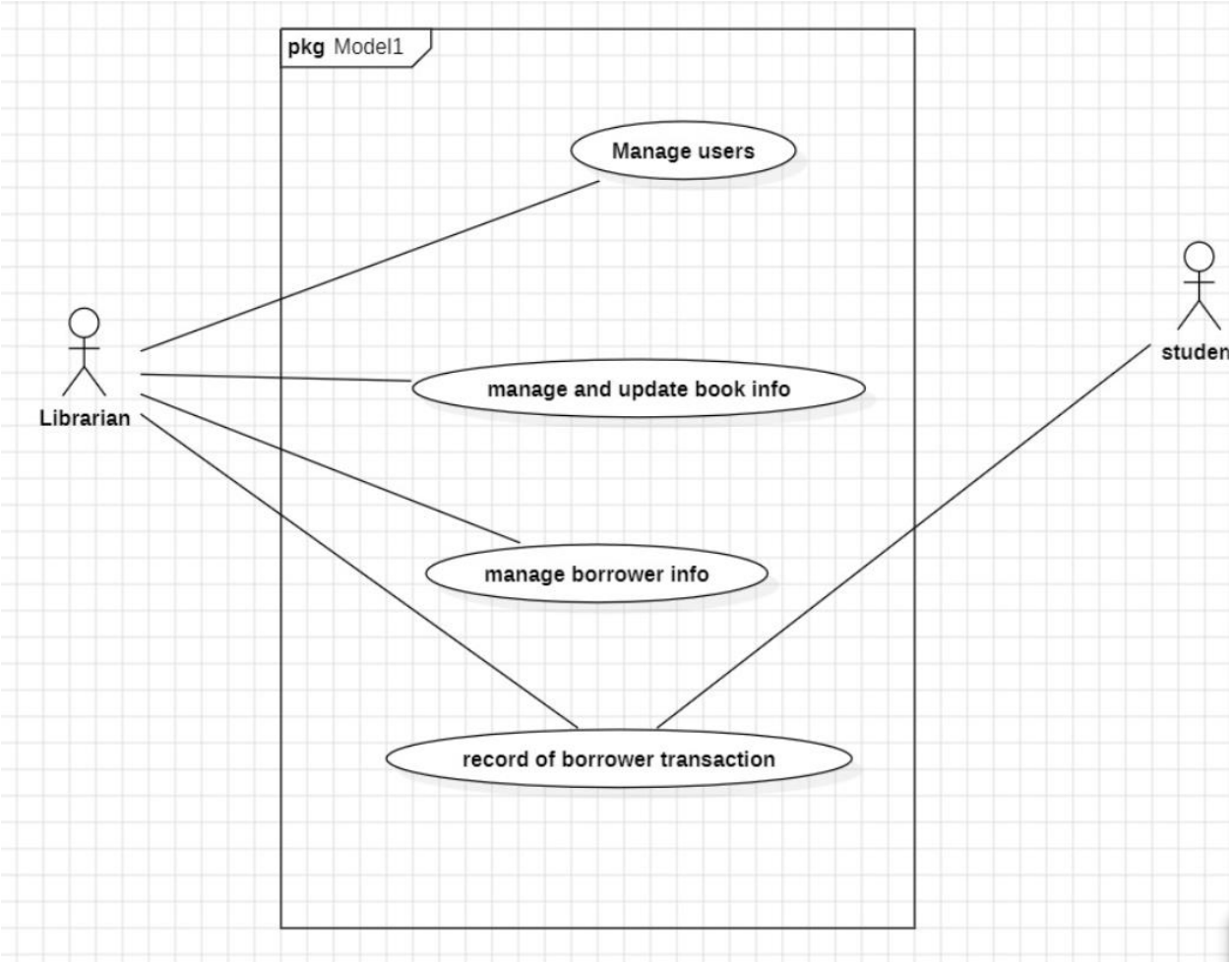
Internal Examiner 1 Internal Examiner 2

INDEX

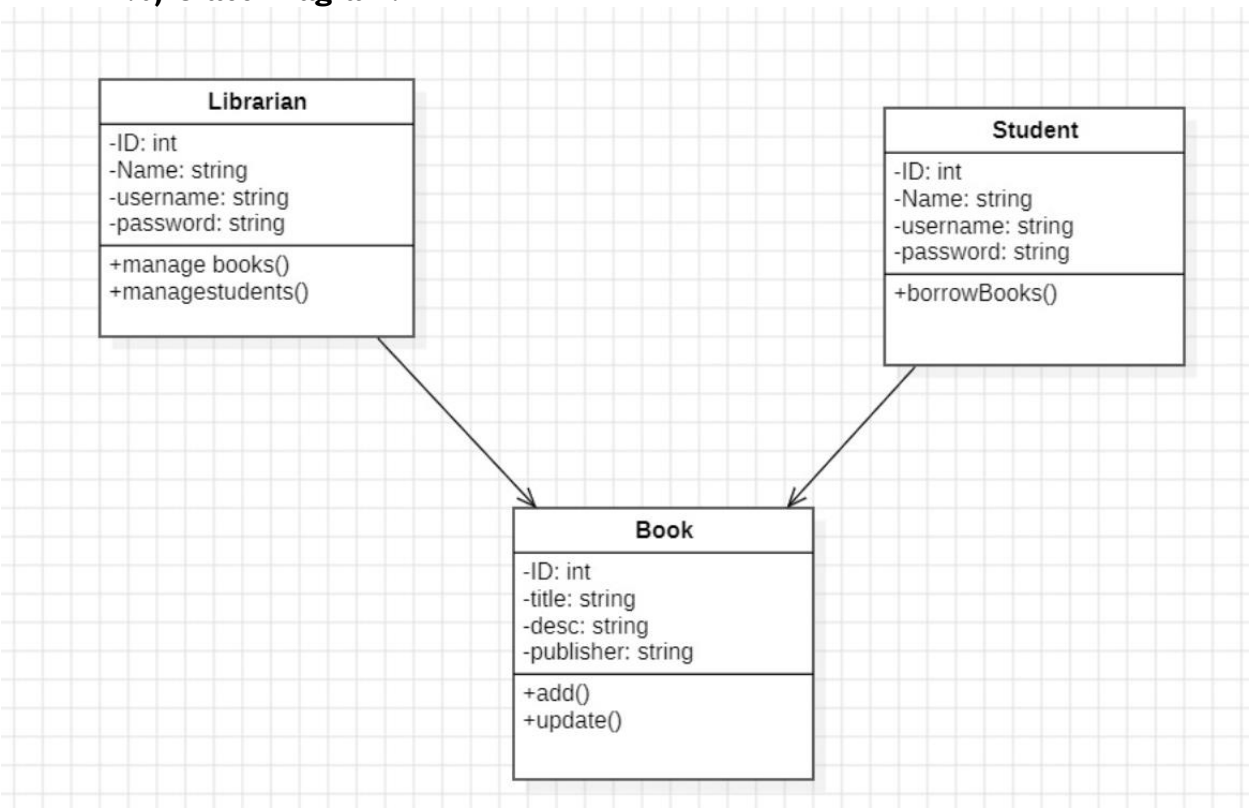
UML DIAGRAMS

1. REFERENCE CENTER

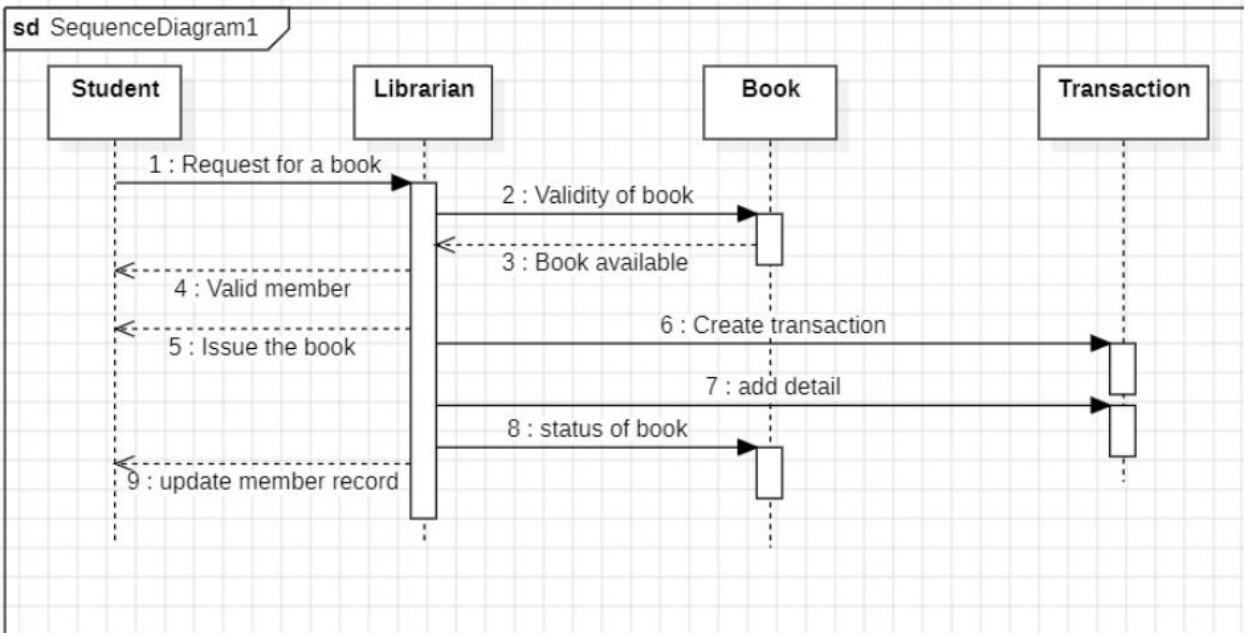
1.a) Use Case Diagram:



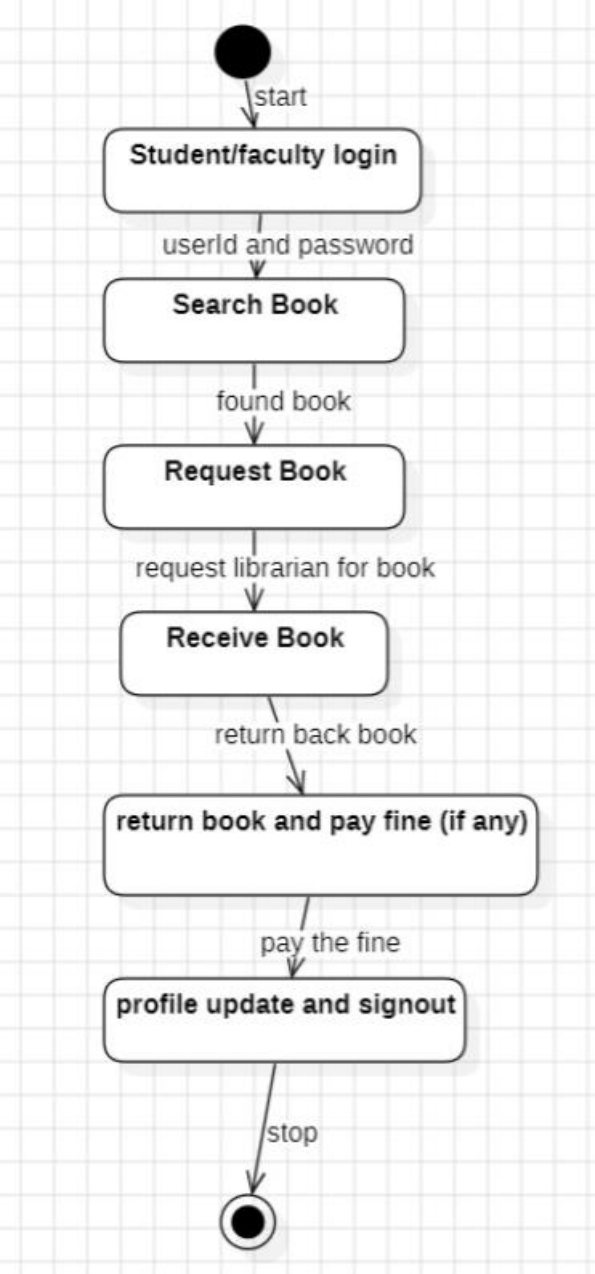
1.b) Class Diagram:



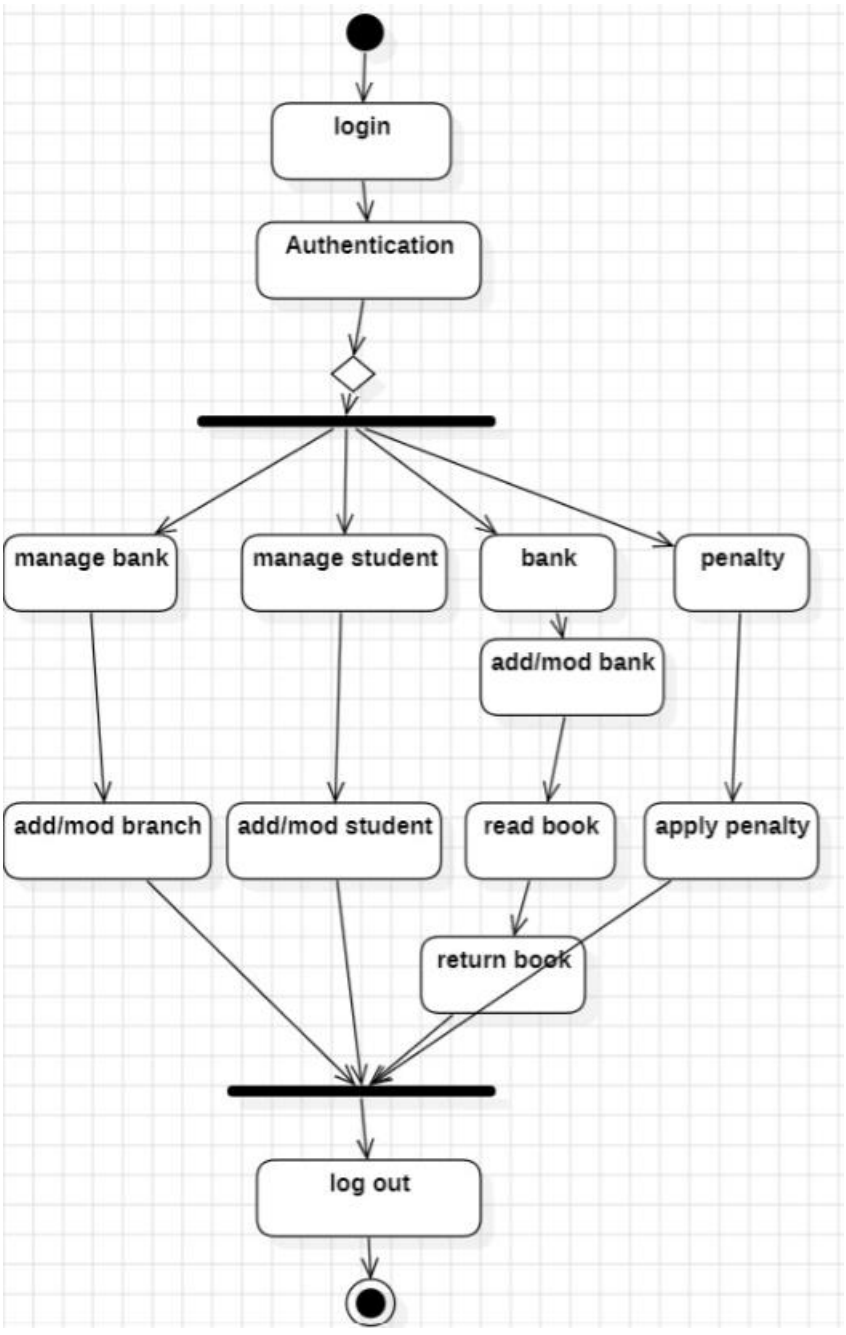
1.c) Sequence Diagram:



1.d) Activity Diagram:

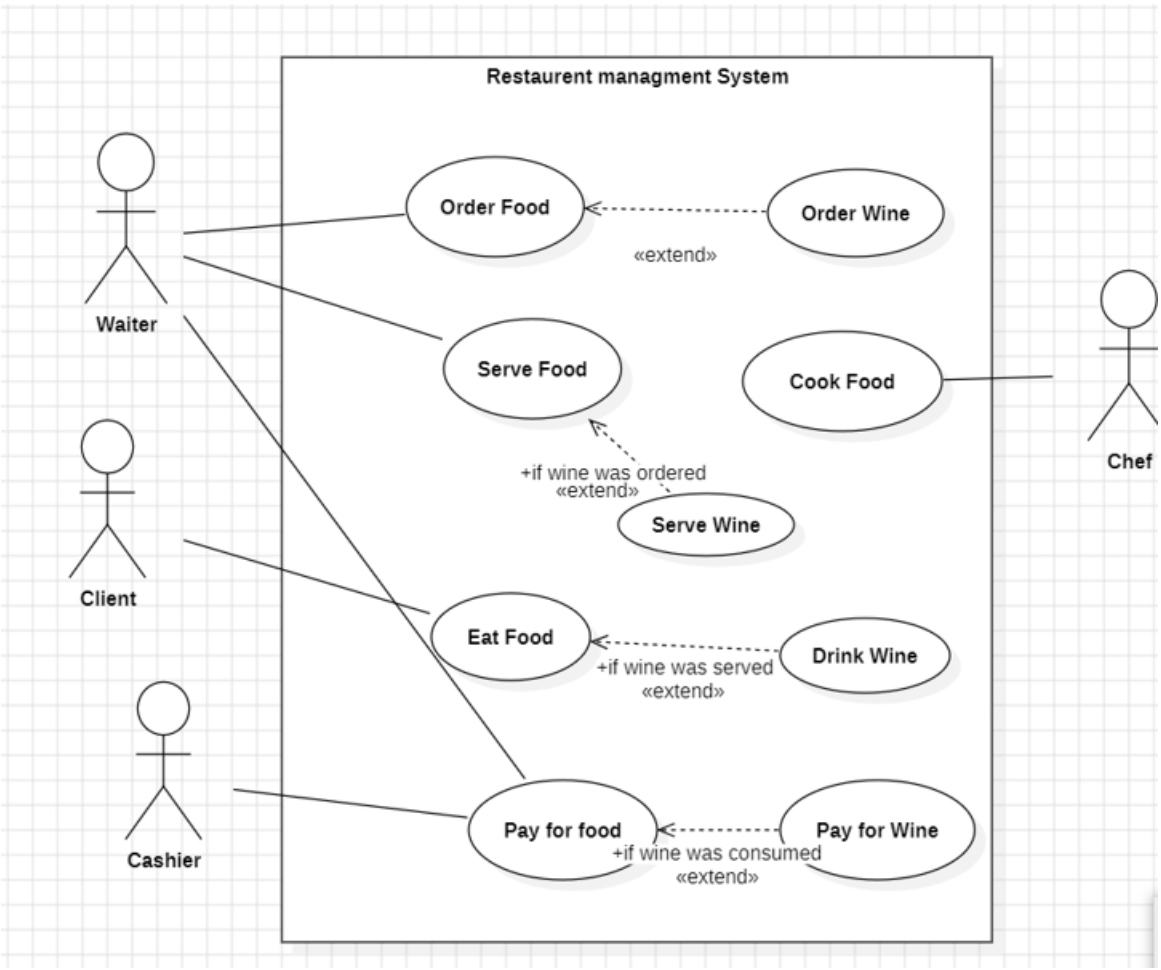


1.e) State-Activity Diagram:

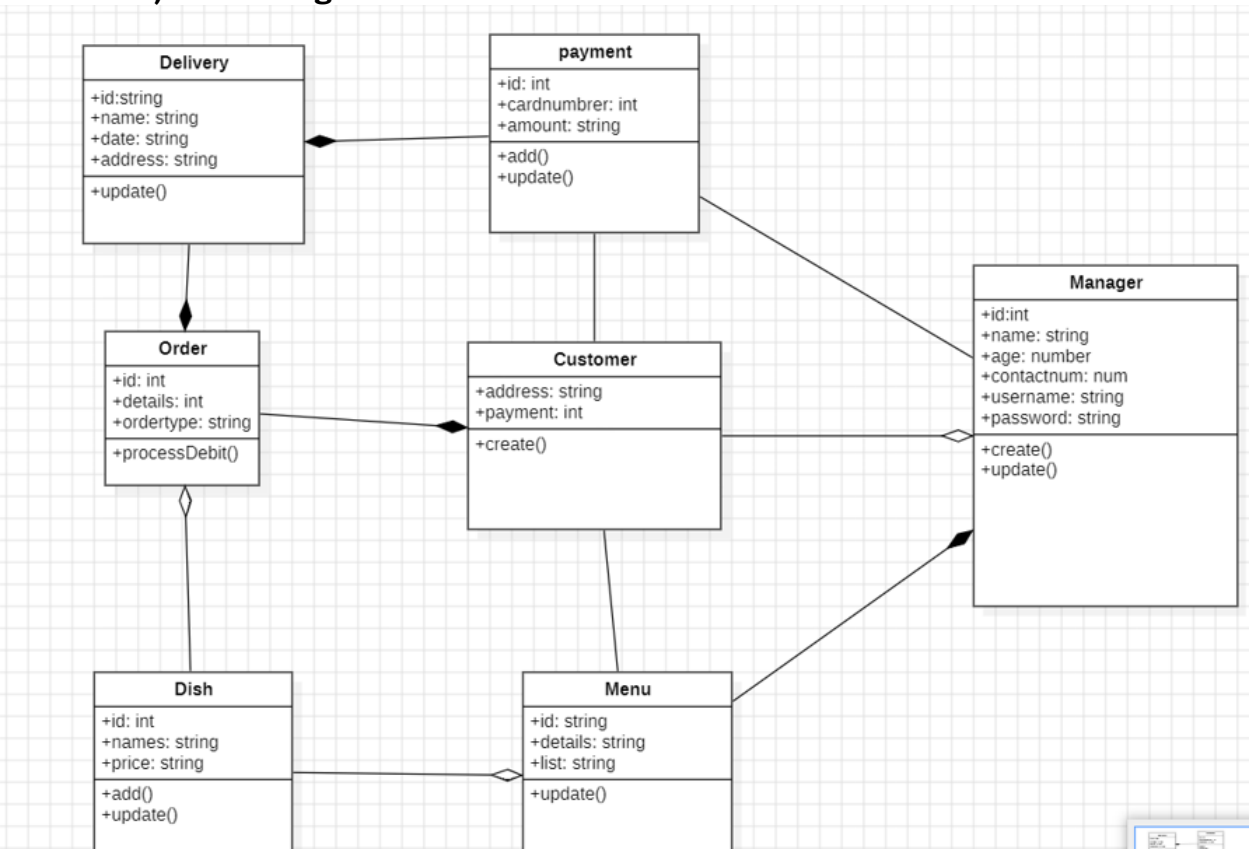


2. CAFE

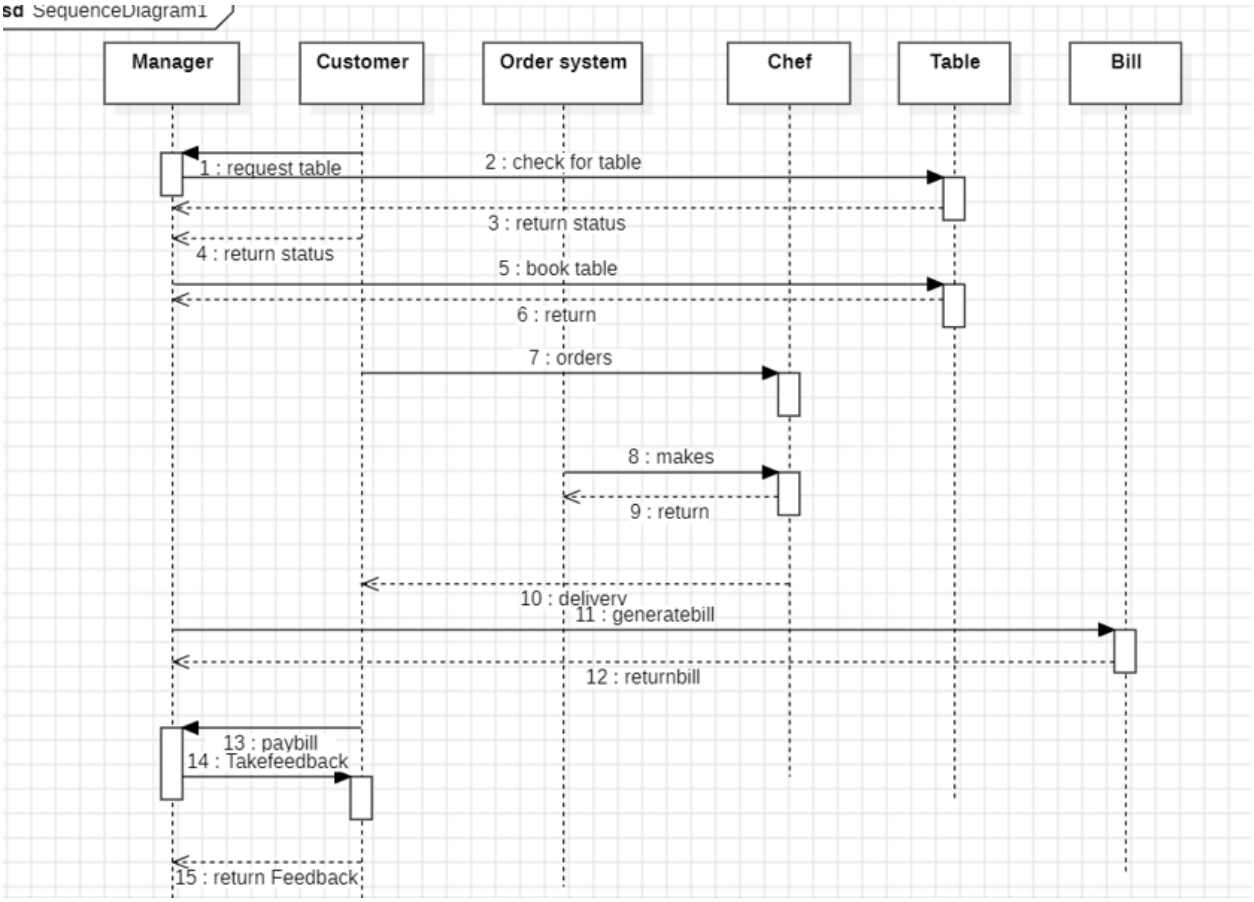
2.a) Use Case Diagram:



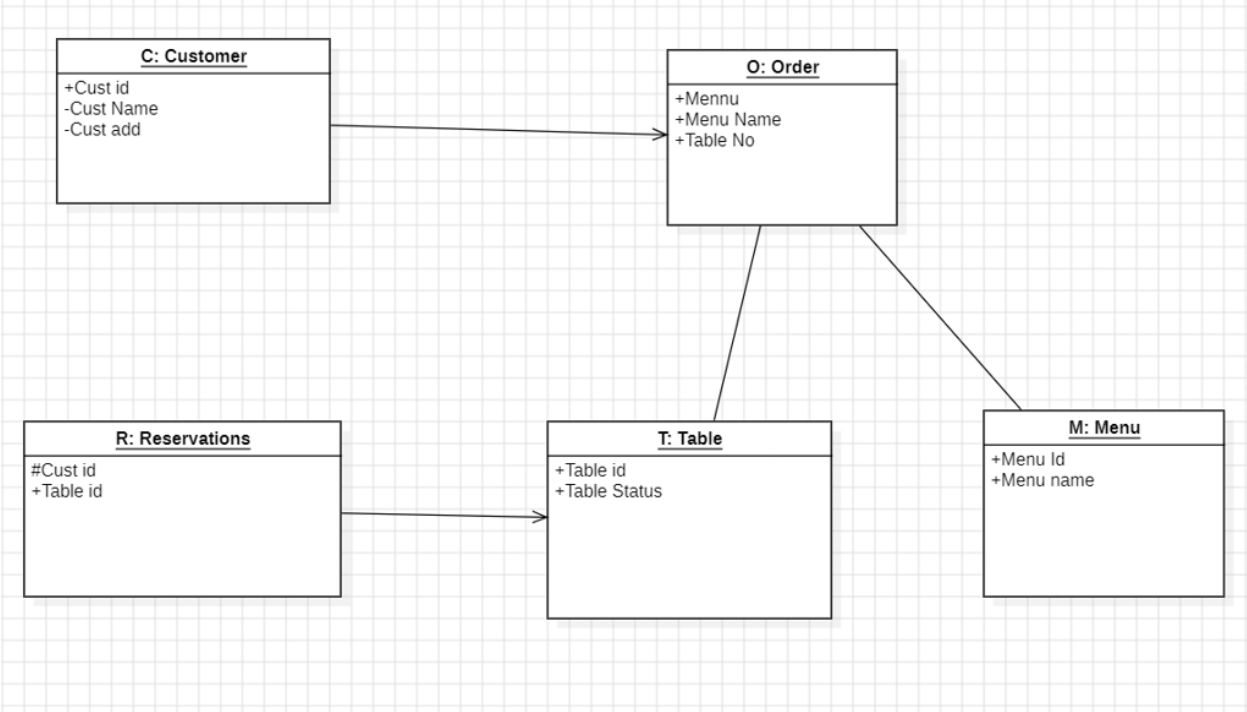
2.b) Class Diagram:



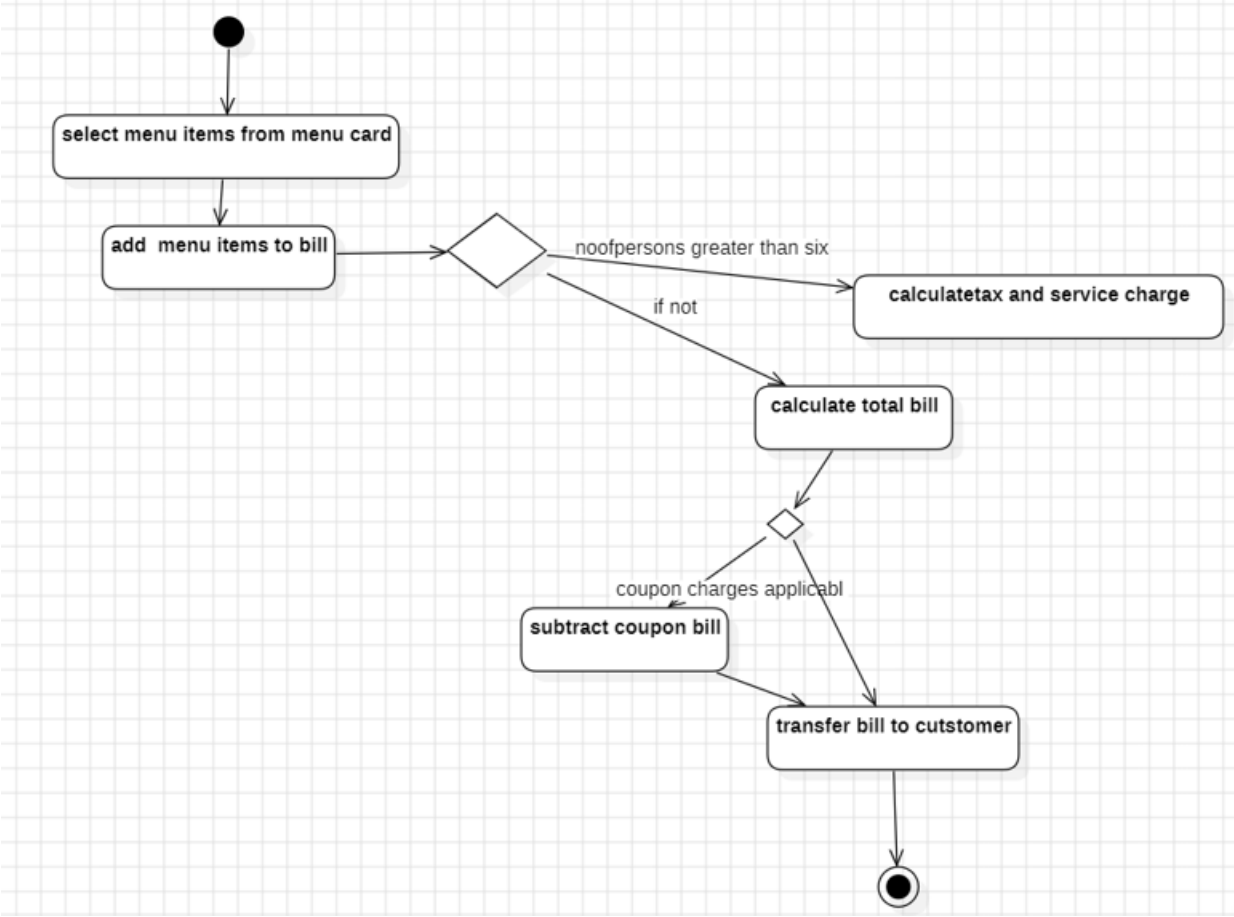
2.c) Sequence Diagram:



2.d) Object Diagram:



2.e) State-Activity Diagram:



3. Basic Java Programs

3.a) Amount:

Code:

```
class Account {
    double balance;

    public Account(double balance) {
        this.balance = balance;
    }

    public void displayBalance() {
        System.out.println("Balance: " + balance);
    }
}
class SavingsAccount extends Account {
    public SavingsAccount(double balance) {
        super(balance);
    }


    public void displayBalance() {
        System.out.println("Savings Account Balance: " + balance);
    }
}

public class Amount {
    public static void main(String[] args) {
        SavingsAccount account = new SavingsAccount(5000);
        account.displayBalance();
    }
}
```

Output:

```
C:\Users\HP\Documents\java programs>javac Amount.java

C:\Users\HP\Documents\java programs>java Amount.java
Savings Account Balance: 5000.0
```

3.b) Transport:

Code:

```
class Vehicle {
    public void type() {
        System.out.println("This is a vehicle");
    }
}
class Car extends Vehicle {
    public void type() {
        System.out.println("This is a car");
    }
}

public class Transport {
    public static void main(String[] args) {
        Car car = new Car();
        car.type();
    }
}
```

Output:

```
C:\Users\HP\Documents\java programs>javac Transport.java

C:\Users\HP\Documents\java programs>java Transport.java
This is a car
```

3.c) Carnivorous:

Code:.

```
class Animal {
        private String name;
        public Animal(String name) {
           this.name = name;
    }
        public void makeSound() {
           System.out.println(name + " makes a sound.");
    }
        public String getName() {
           return name;
    }
}
class Dog extends Animal {
        public Dog(String name) {
           super(name);
    }
        public void makeSound() {
           System.out.println(getName() + " barks.");
    }
}

public class Carnivorous {
        public static void main(String[] args) {
           Dog dog = new Dog("Buddy");
           dog.makeSound();
           Animal animal = new Animal("Generic Animal");
           animal.makeSound();
    }
}
```

Output:

```
C:\Users\HP\Documents\java programs>javac Carnivorous.java

C:\Users\HP\Documents\java programs>java Carnivorous.java
Buddy barks.
Generic Animal makes a sound.
```

3.d) Work:

Code:

```
class Employee {
    String name;

    public Employee(String name) {
        this.name = name;
    }

    public void work() {
        System.out.println(name + " is working");
    }
}

class Manager extends Employee {
    public Manager(String name) {
        super(name);
    }

       public void work() {
        System.out.println(name + " is managing the team");
    }
}

public class Work {
    public static void main(String[] args) {
        Manager manager = new Manager("Alice");
        manager.work();
    }
}
```

Output;

```
C:\Users\HP\Documents\java programs>javac Work.java

C:\Users\HP\Documents\java programs>java Work.java
Alice is managing the team
```

3.e) Study:

Code:

```
class Book {
    String title;

    public Book(String title) {
        this.title = title;
    }

    public void display() {
        System.out.println("Book Title: " + title);
    }
}


class EBook extends Book {
    public EBook(String title) {
        super(title);
    }

       public void display() {
        System.out.println("E-Book Title: " + title);
    }
}

public class Study {
    public static void main(String[] args) {
        EBook ebook = new EBook("Java Programming");
        ebook.display();
    }
}
```

Output:

```
C:\Users\HP\Documents\java programs>javac Study.java

C:\Users\HP\Documents\java programs>java Study.java
E-Book Title: Java Programming
```

3.f) Account:

Code:

```
class BankAccount {
    double balance;

    public BankAccount(double balance) {
        this.balance = balance;
    }

    public void deposit(double amount) {
        balance += amount;
    }

    public void displayBalance() {
        System.out.println("Bank Account Balance: " + balance);
    }
}


class CheckingAccount extends BankAccount {
    public CheckingAccount(double balance) {
        super(balance);
    }

    public void displayBalance() {
        System.out.println("Checking Account Balance: " + balance);
    }
}

public class Account {
    public static void main(String[] args) {
        CheckingAccount account = new CheckingAccount(1000);
        account.deposit(500);
        account.displayBalance();
    }
}
```

Output:

```
C:\Users\HP\Documents\java programs>javac Account.java

C:\Users\HP\Documents\java programs>java Account.java
Checking Account Balance: 1500.0
```

3.g) Measurements:

Code:

```
class Room {

    double length, width;


    public Room(double length, double width) {
        this.length = length;
        this.width = width;
    }


    public double calculateArea() {
        return length * width;
    }
}


class RoomWithHeight extends Room {

    double height;


    public RoomWithHeight(double length, double width, double height) {
        super(length, width); // Call the parent class constructor
        this.height = height;
    }


    public double calculateVolume() {
        return calculateArea() * height;
    }
}

public class Measurements {
    public static void main(String[] args) {

        RoomWithHeight room = new RoomWithHeight(5, 4, 3);


        double area = room.calculateArea();
        System.out.println("Room Area: " + area + " square meters");


        double volume = room.calculateVolume();
        System.out.println("Room Volume: " + volume + " cubic meters");
    }
}
```

Output:

```
C:\Users\HP\Documents\java programs>javac Measurements.java

C:\Users\HP\Documents\java programs>java Measurements.java
Room Area: 20.0 square meters
Room Volume: 60.0 cubic meters
```

.

3.h)ArmstrongNumber

```
import java.util.Scanner;

public class ArmstrongNumber {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter a number: ");
        int num = scanner.nextInt();

        int originalNum = num;
        int result = 0;
        int digits = String.valueOf(num).length();

        while (num != 0) {
            int remainder = num % 10;
            result += Math.pow(remainder, digits);
            num /= 10;
        }

        if (result == originalNum) {
            System.out.println(originalNum + " is an Armstrong number.");
        } else {
            System.out.println(originalNum + " is not an Armstrong number.");
        }

        scanner.close();
    }
}
```

Output:

```
C:\Users\HP\Documents\java programs>javac ArmstrongNumber.java

C:\Users\HP\Documents\java programs>java ArmstrongNumber.java
Enter a number: 20
20 is not an Armstrong number.
```

3.i)Art:

Code:

```
class Shape {
    public void draw() {
        System.out.println("Drawing a shape");
    }
}


class Circle extends Shape {

    public void draw() {
        System.out.println("Drawing a circle");
    }
}

public class Art {
    public static void main(String[] args) {
        Circle circle = new Circle();
        circle.draw();
    }
}
```

Output:

```
C:\Users\HP\Documents\java programs>javac Art.java

C:\Users\HP\Documents\java programs>java Art.java
Drawing a circle
```

3.j):Sounds:

Code:

```
class Animal {
  public void animalSound() {
    System.out.println("The animal makes a sound");
  }
}

class Pig extends Animal {
  public void animalSound() {
    System.out.println("The pig says: wee wee");
  }
}

class Dog extends Animal {
  public void animalSound() {
    System.out.println("The dog says: bow wow");
  }
}

class Sounds {
  public static void main(String[] args) {
    Animal myAnimal = new Animal();
    Animal myPig = new Pig();
    Animal myDog = new Dog();
    myAnimal.animalSound();
    myPig.animalSound();
    myDog.animalSound();
  }
}
```

Output:

```
C:\Users\HP\Documents\java programs>java Sounds.java
The animal makes a sound
The pig says: wee wee
The dog says: bow wow
```