

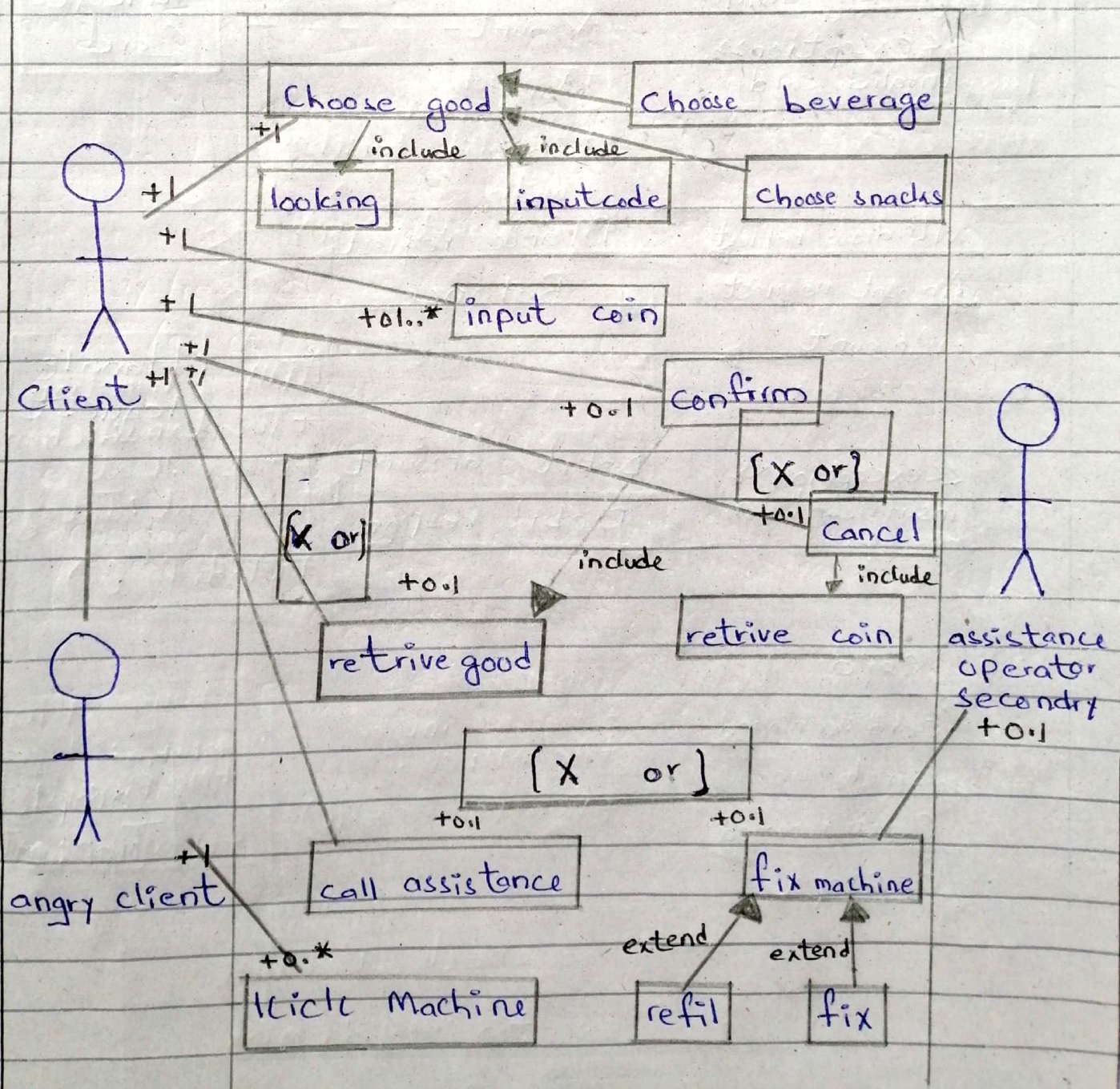
Name : Amad Afridi

ID : 13119

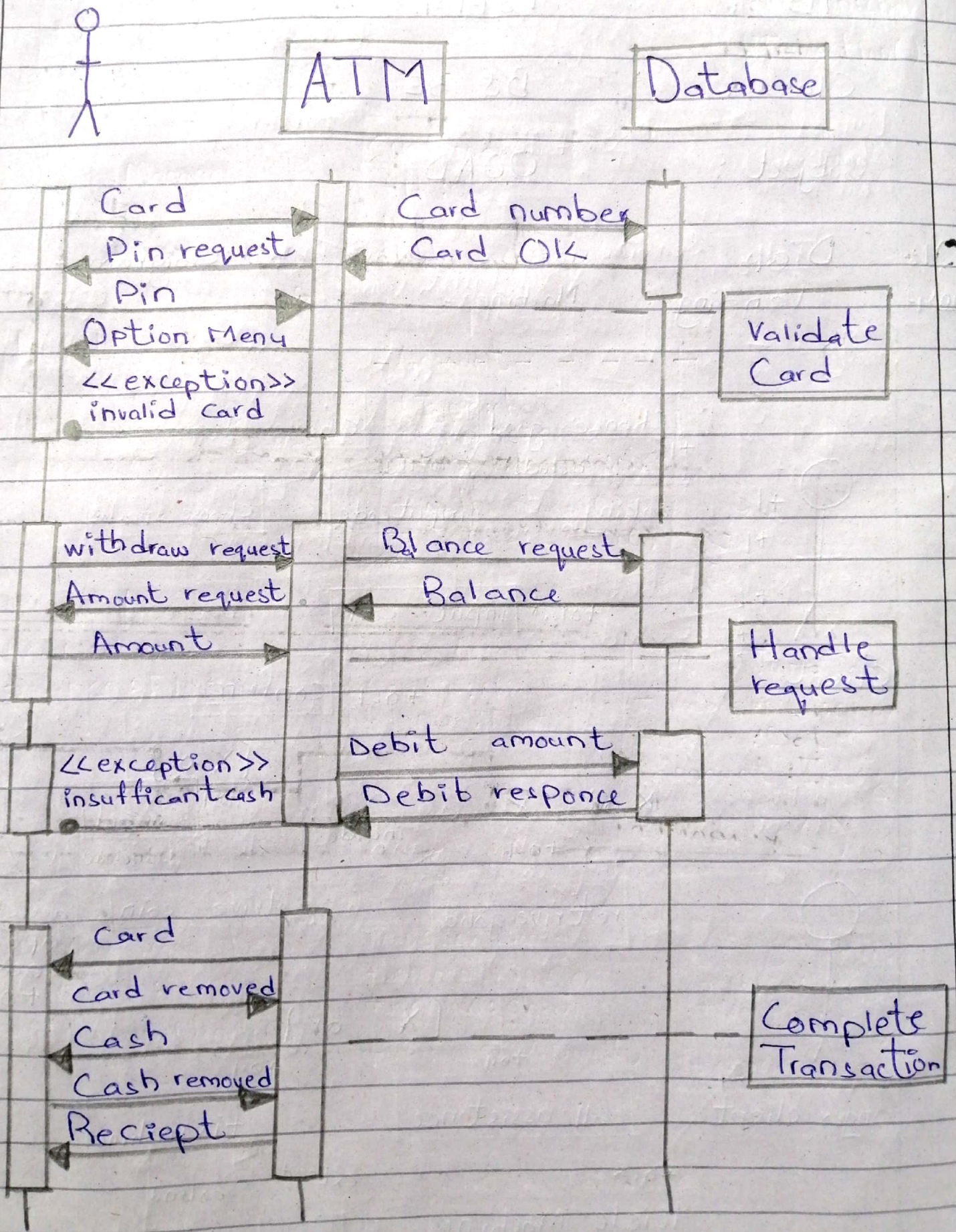
Class : BS SE B

Subject : OOAD

Q1:- Draw use case diagrams:-
Ans:- Vending Machine:-

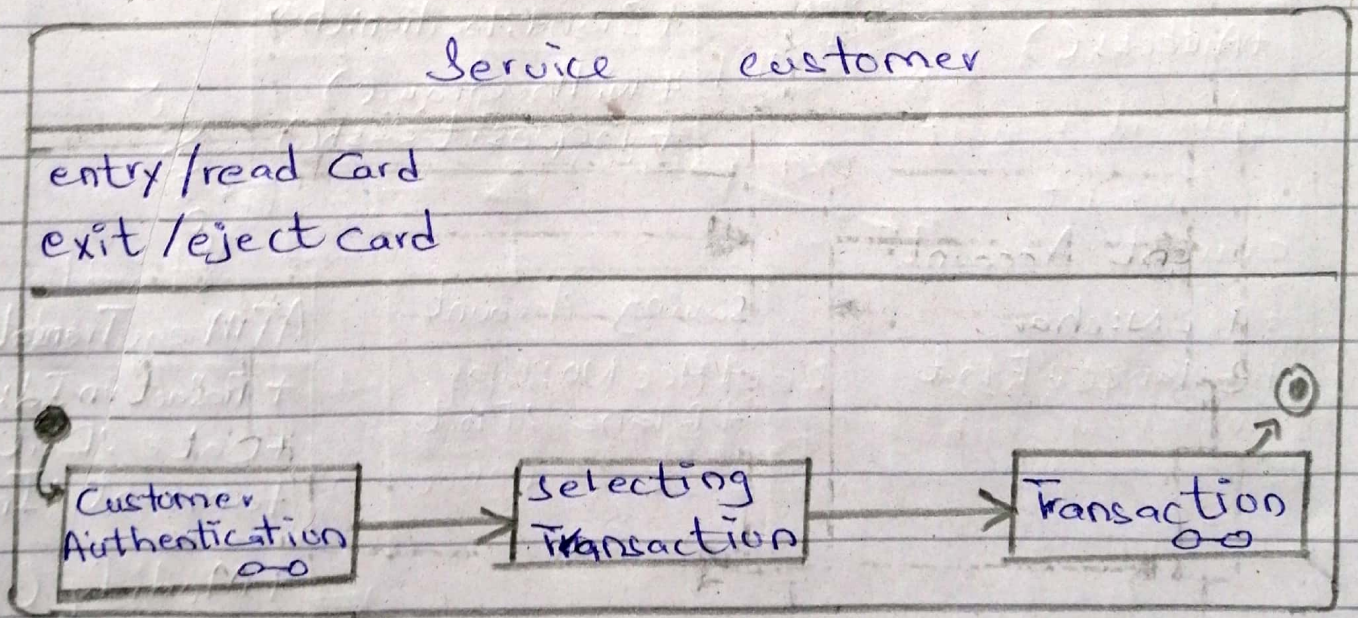
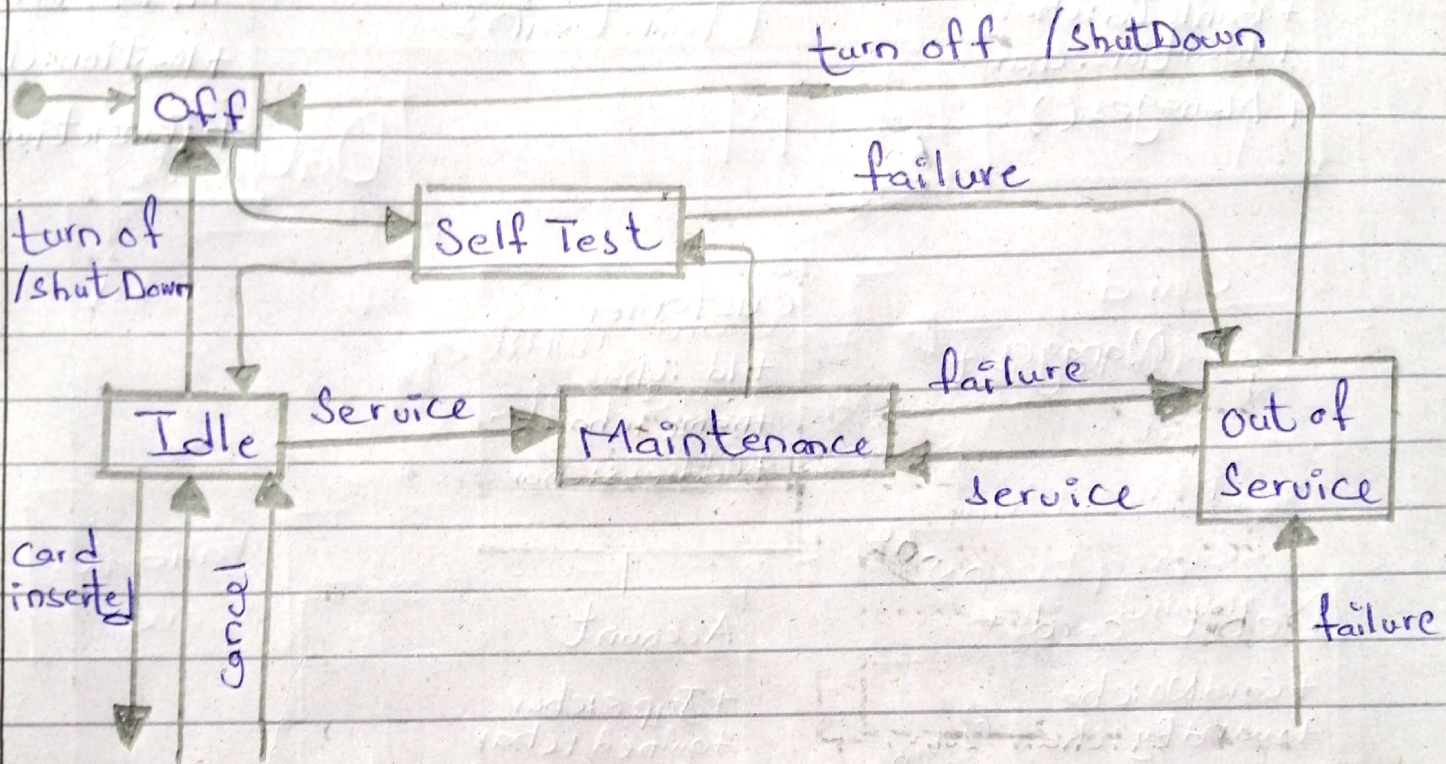


Q2 - Draw Sequence Diagram:-
Ans:- Sequence Diagram for ATM withdrawal:-



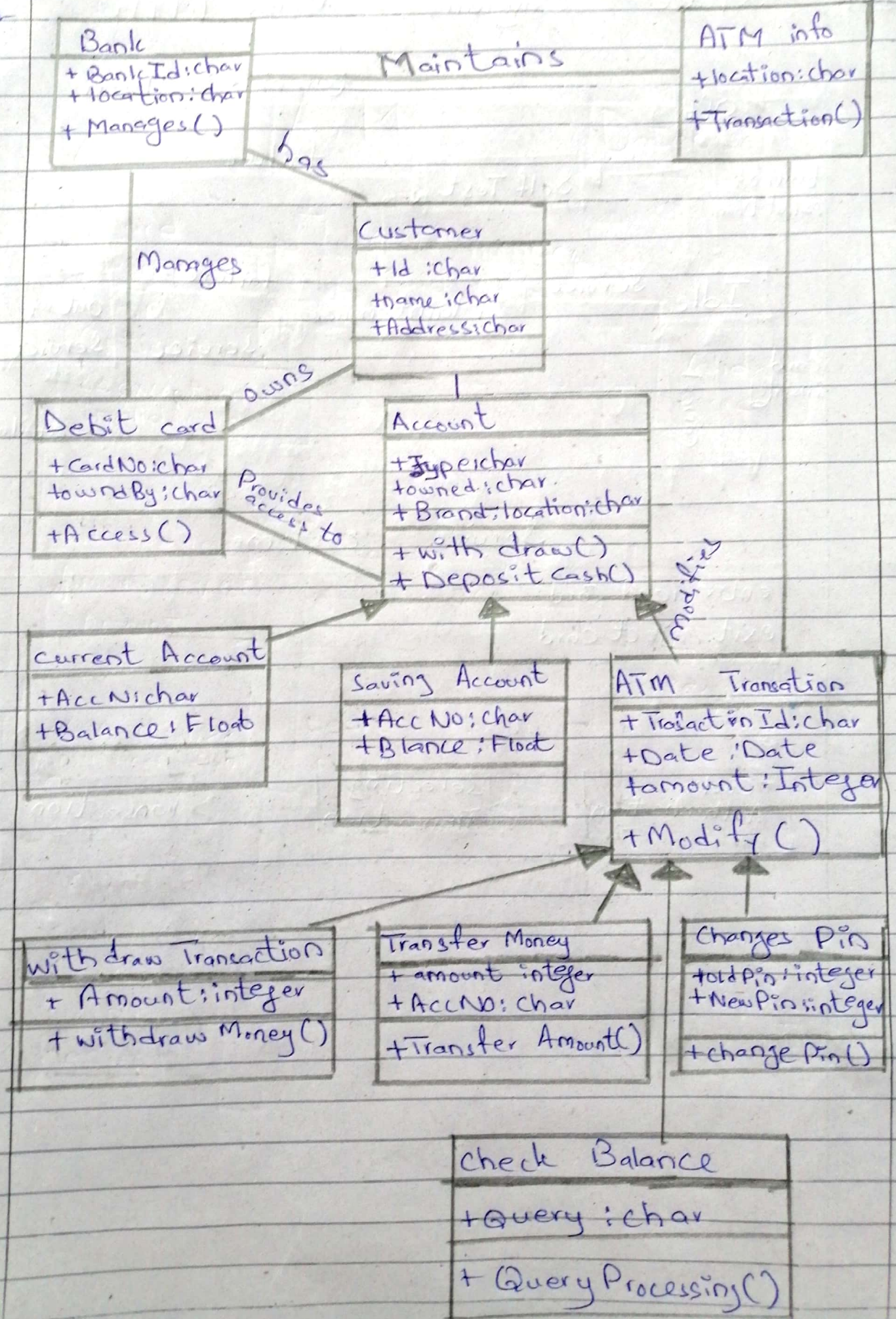
Q3
Ans

Draw state Daigram Bank ATM
State Daigram



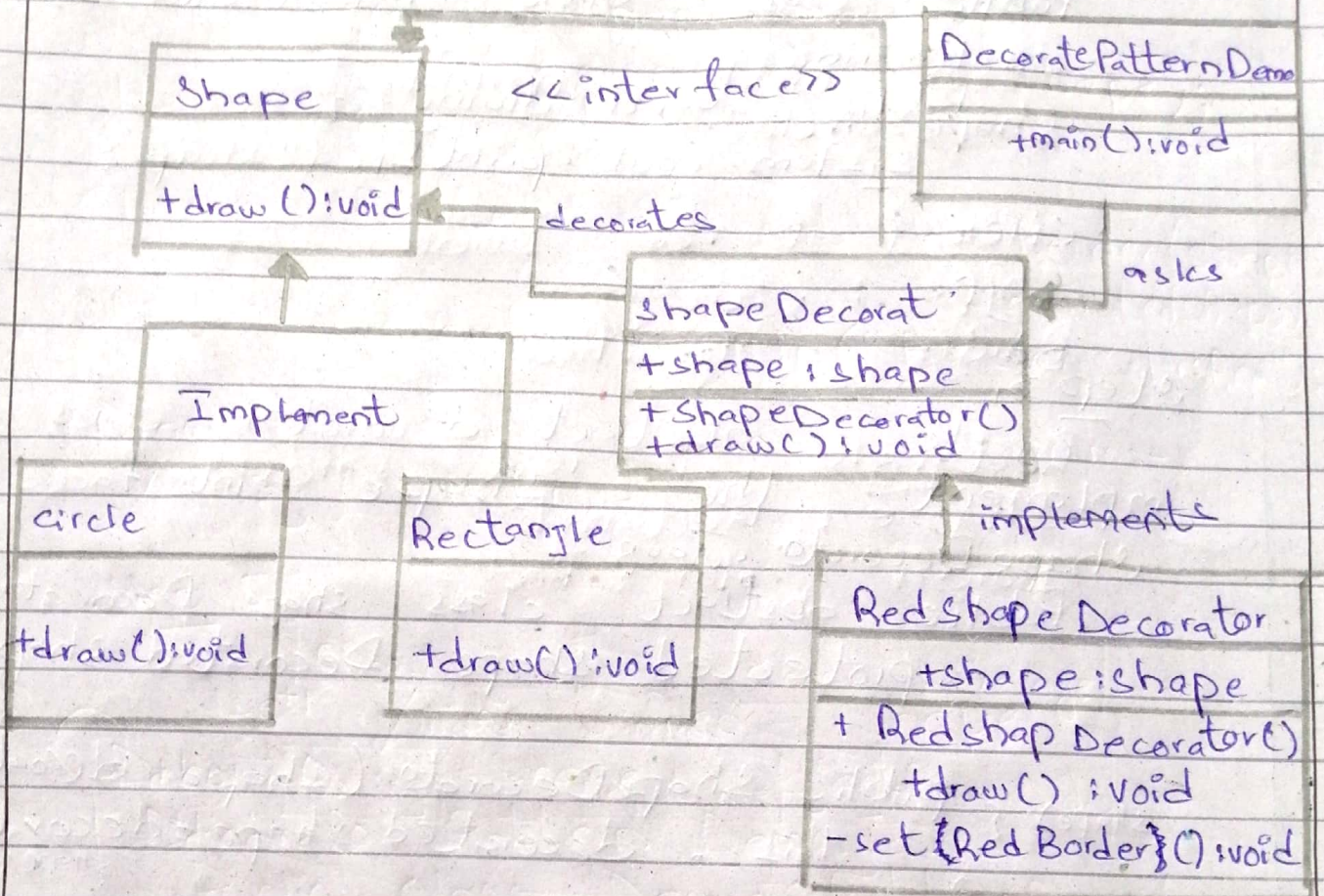
Q4r
Ansl

Draw class Daigram.



Q5
Ans

Design Pattern:



- Step 1:-

Create interface
Shape.java

```

public interface shape {
    void draw();
}
    
```

- Step 2:-

Create concrete classes implementing
the same interface
Rectangle.java

```

public class Rectangle implements
@override
public void draw() {
    System.out.println("shape:
}
    
```


circle, Java

```

Public class circle implements sh
@ override
Public void draw() {
System.out.println ("shape:");
}
}

```

step 3.

create abstract decorator class implement the shape interface.

Shape Decorator, Java

```

Public abstract class shape Decorator implem
protected shape Decorated shape;
shape) { Public shapeDecorator (shape decorated
this . decorated shape = decorated
shape;
}
Public void draw() {
decorated shape.draw();
}
}
}

```

step 4.

create concrete decorator class extending the shape Decorator class

Red Shape Decorator, Java

```

shape decorator { Public class Redshape Decorator extends
Public Redshape Decorator (shape
super (decorator shape); decorated
shape);
}
}
@ override
Public void draw();

```



```

decorated shape.draw();
set red border(decorated shape);
}

```

```

private void set RedBorder(shape
decoratedshape){
System.out.println("Border Color:Red");
}
}

```

Step 5

Use the Red Shape
Decorator Shape Objects -

Decorator Pattern Demo.java

```

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

```

```

Shape circle = new circle();

```

```

Shape red circle = new RedShape
Decorator(new circle());

```

```

Shape red rectangle = new RedShape
Decorator(new
rectangle());

```

```

System.out.println("circle with normal
border");

```

```

circle.draw();

```

```

System.out.println("In circle of red
border");

```

```

red circle.draw();

```

```

System.out.println("In rectangle of red
border");

```

```

} redRectangle.draw();
}

```

Step 6

verify the output

circle with normal border
Shape: circle

circle of red border
Shape: circle
Border color: Red

Rectangle of red border
Shape: Rectangle
Border color: Red -