LAB RECORD

23CSE111- Object Oriented Programming

Submitted by

CH.SC.U4CSE24128 – M. Charitha Varshini

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF COMPUTING

CHENNAI

March - 2025

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.U4CSE24128 – M. Charitha Varshini*** 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

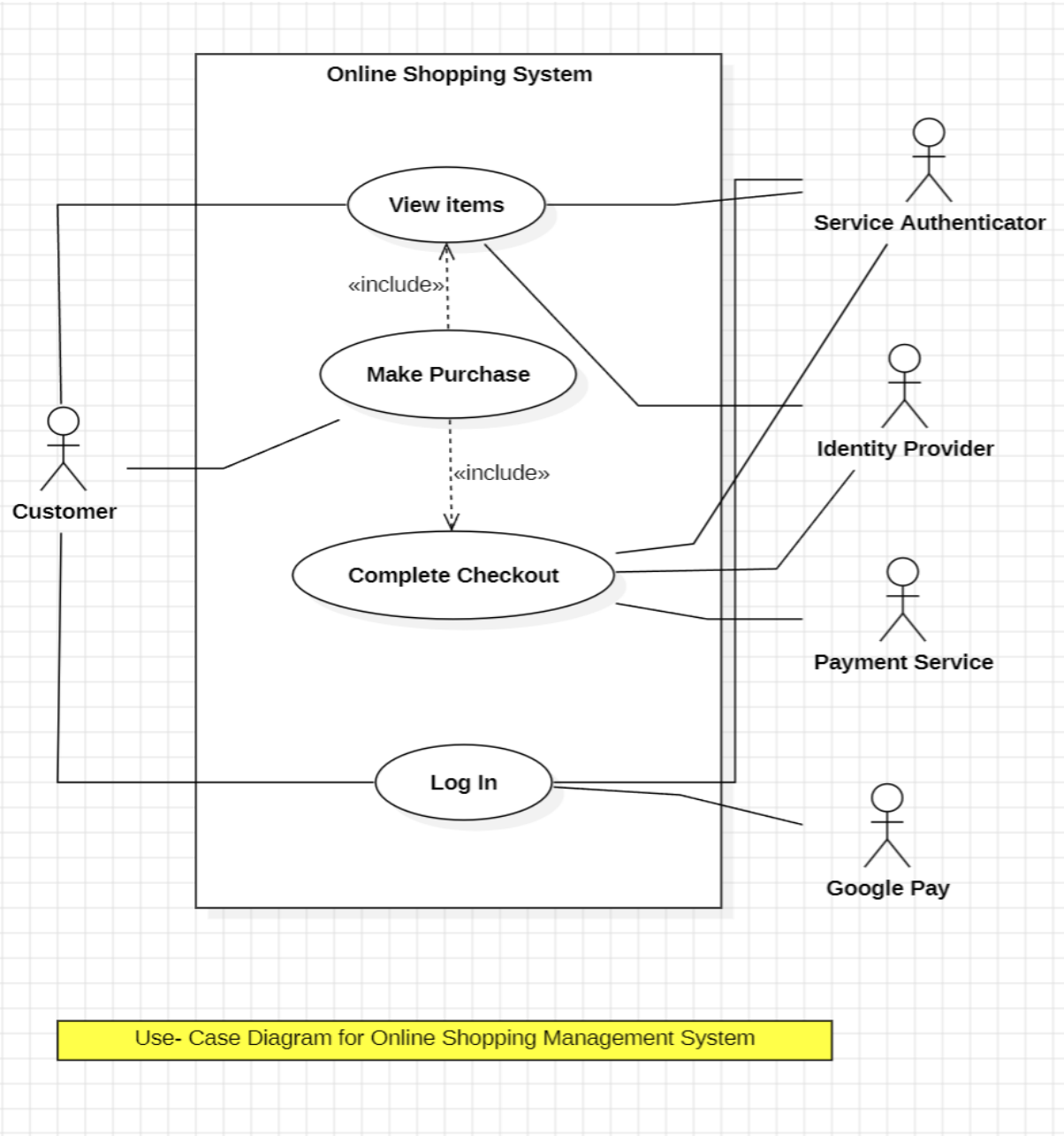
Internal Examiner 1 Internal Examiner 2

INDEX

UML DIAGRAMS

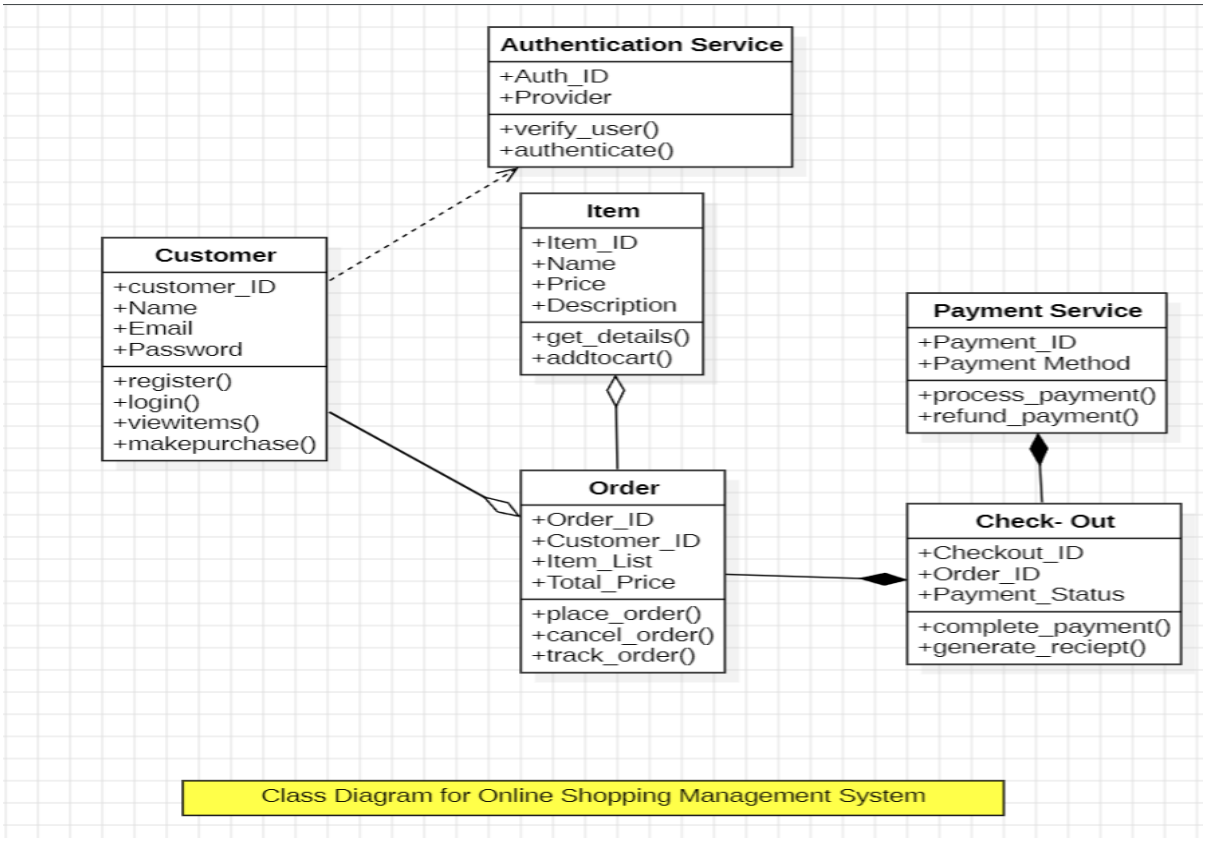
1. ONLINE SHOPPING MANAGEMENT SYSTEM

1. a) Use Case Diagram:



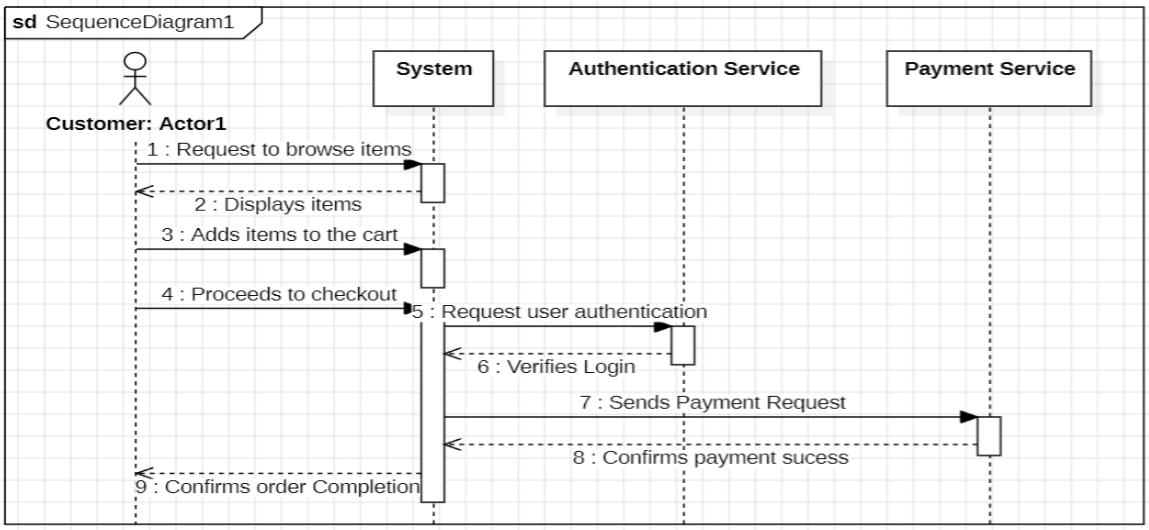
Use- Case Diagram for Online Shopping Management System

1. b) Class Diagram:



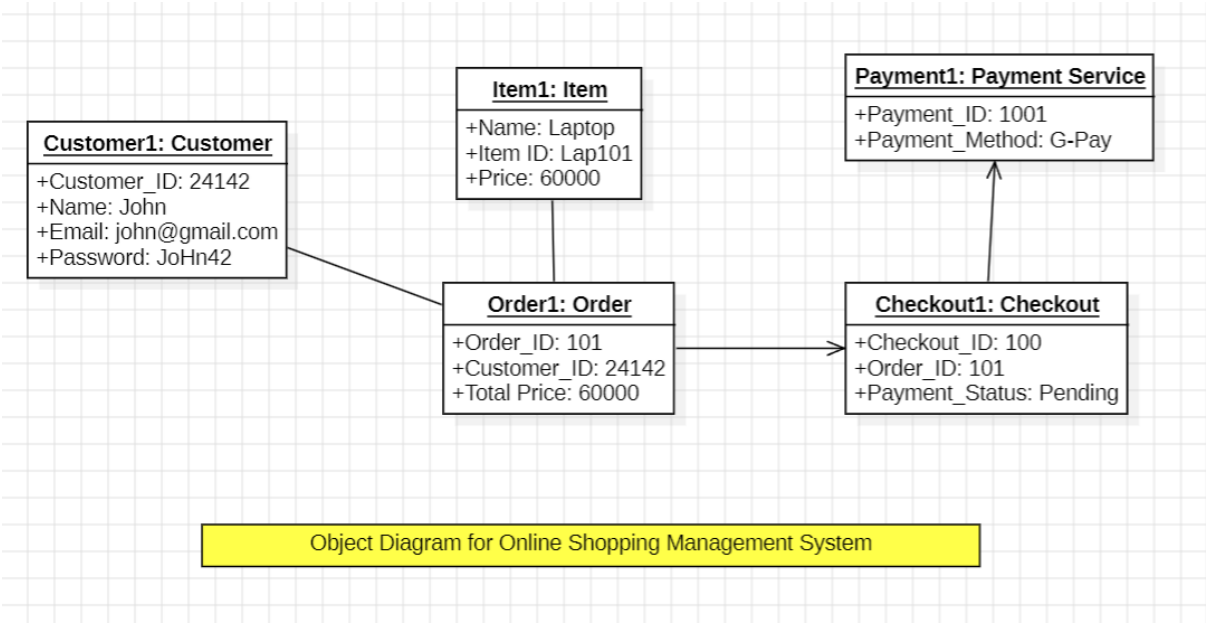
Class Diagram for Online Shopping Management System

1. c) Sequence Diagram:

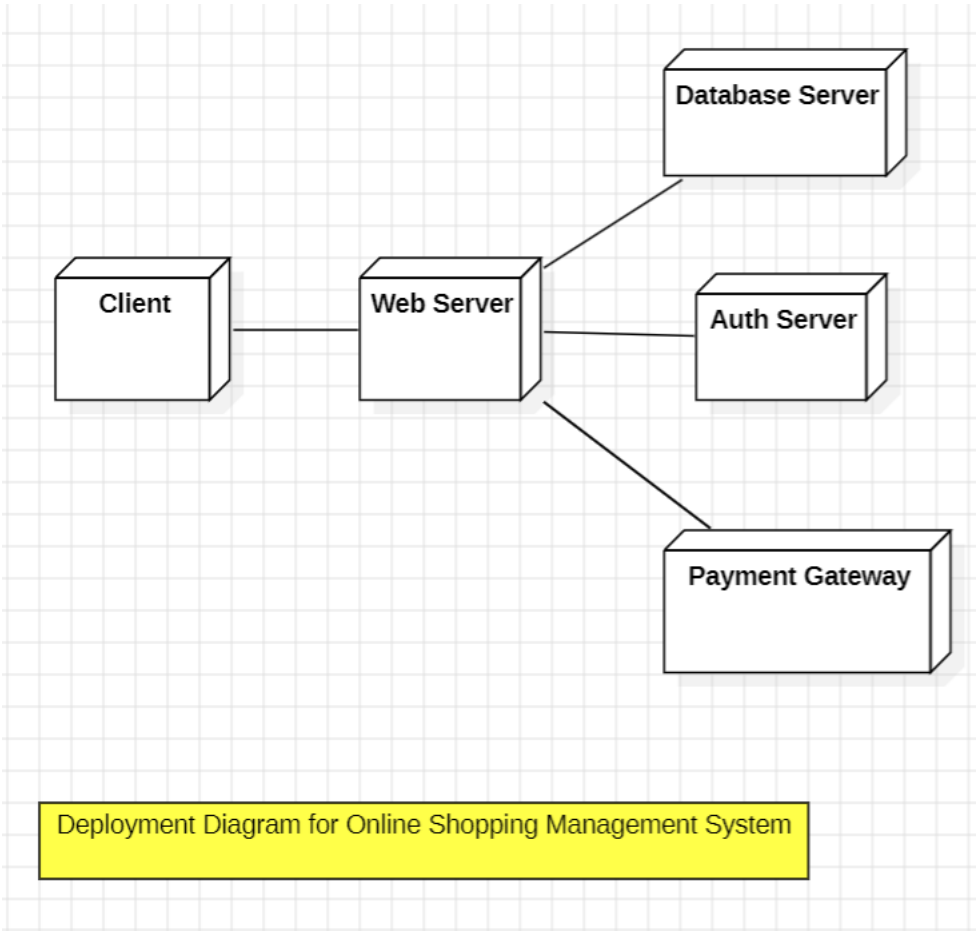


Sequence Diagram for Online Shopping Management System

1. d) Object Diagram:

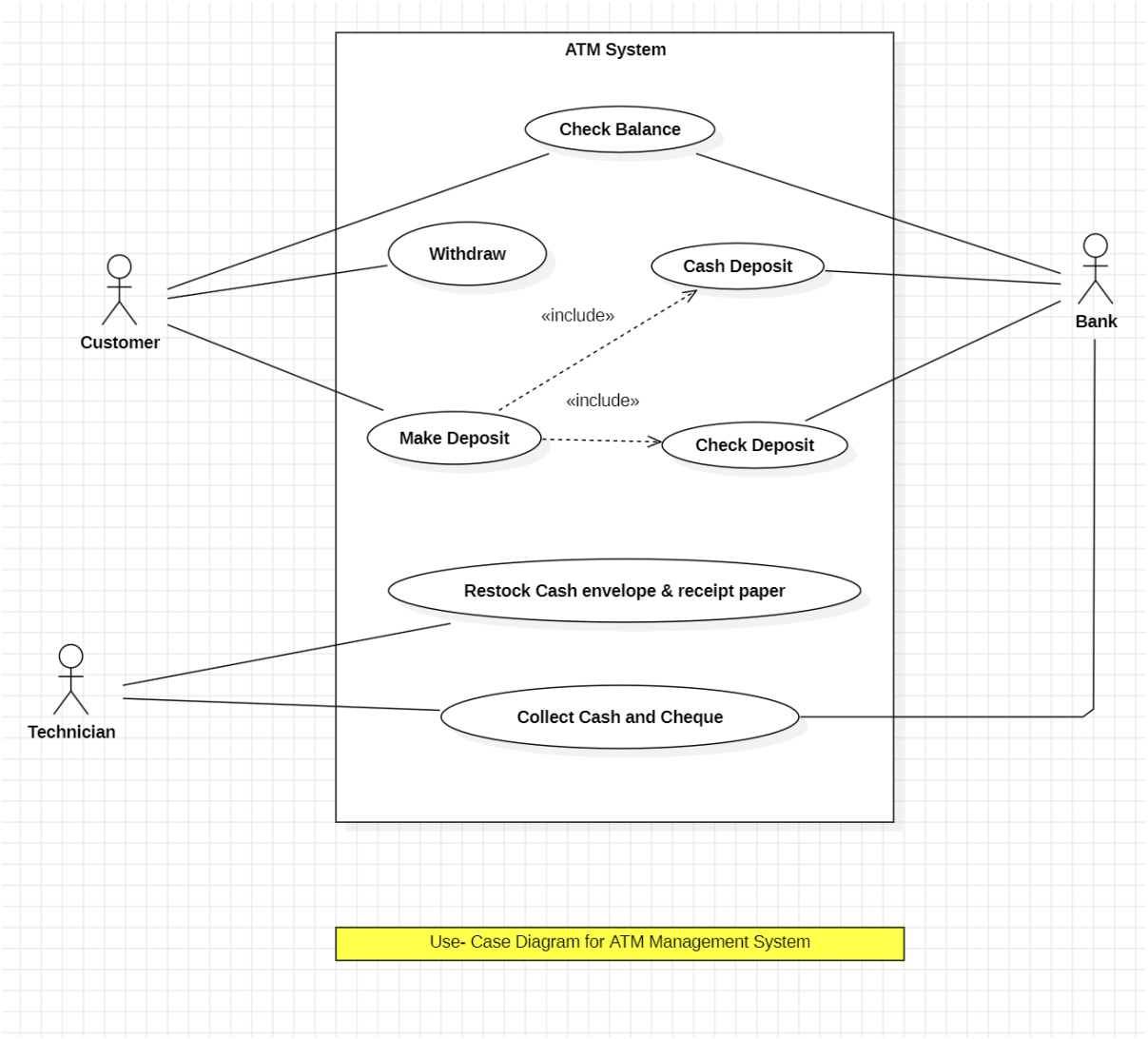


1. e) Deployment Diagram:



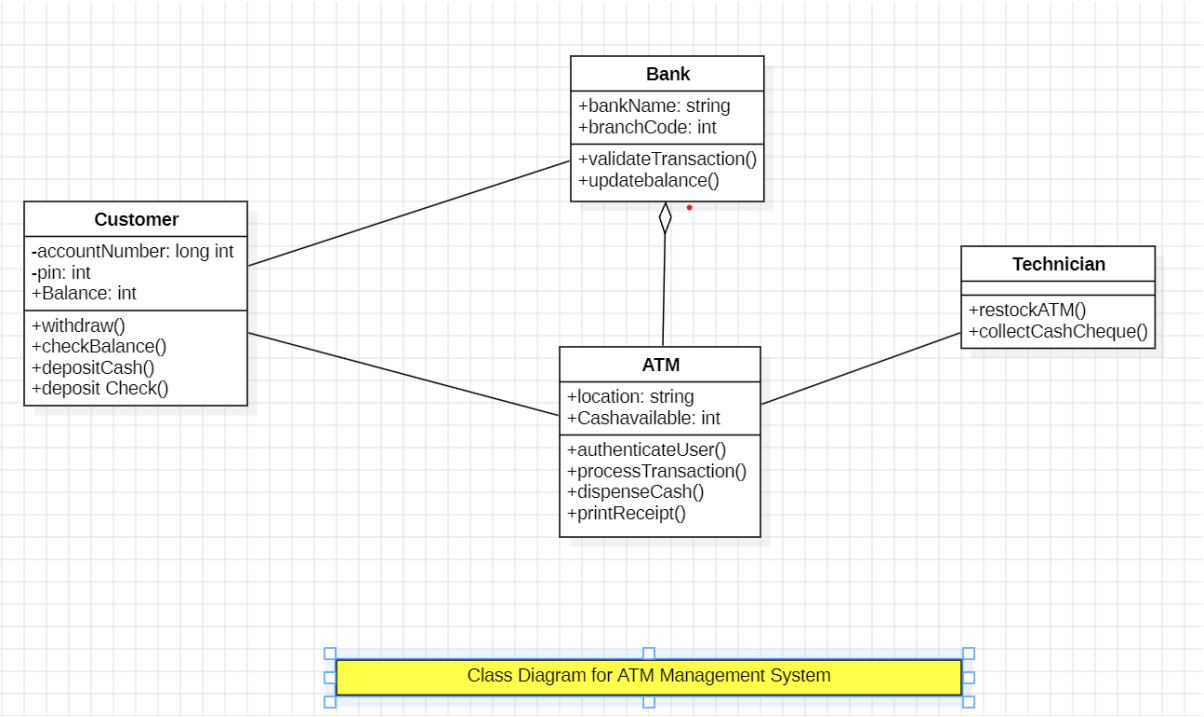
2. ATM MANAGEMENT SYSTEM

2.a) Use Case Diagram:

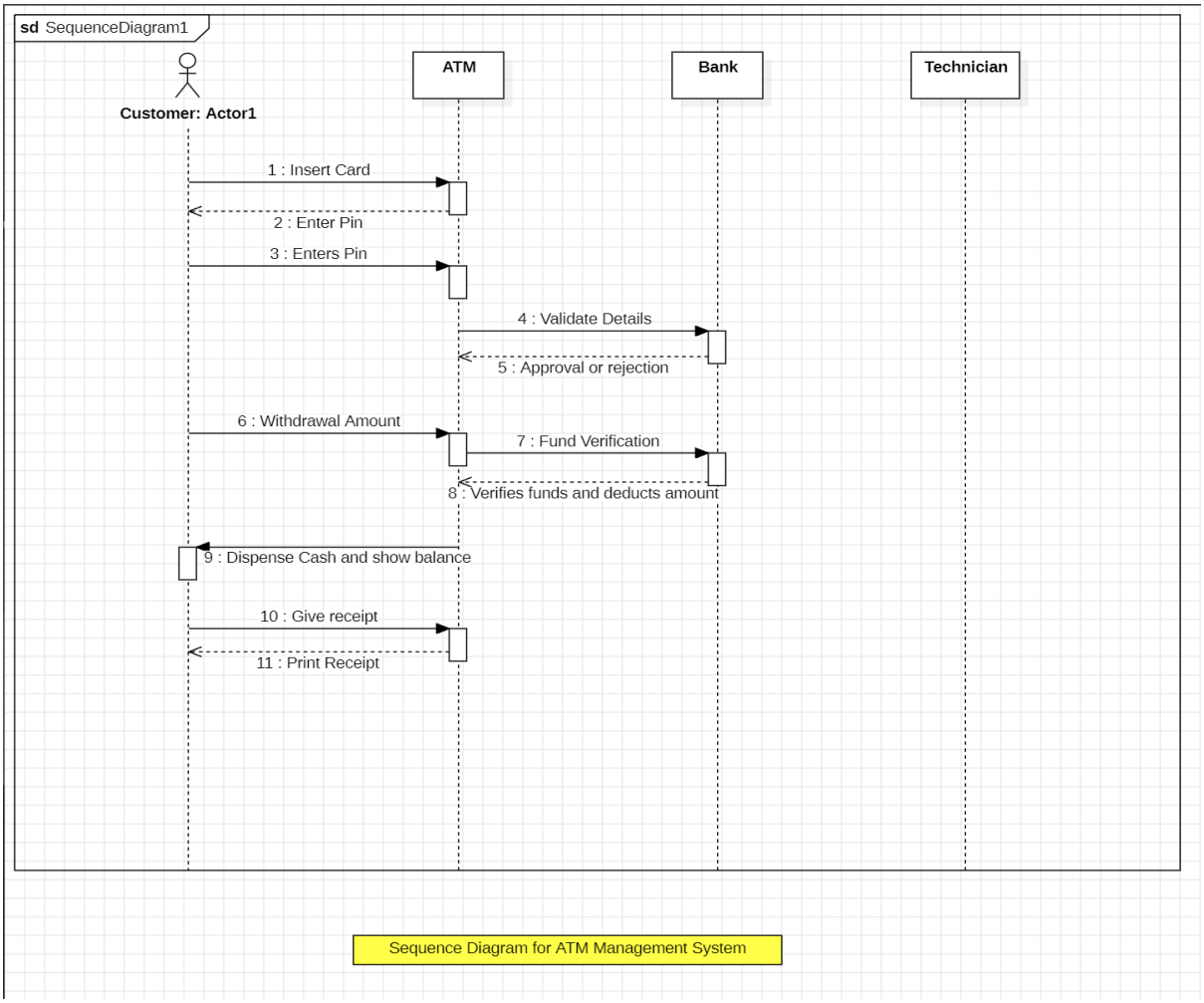


Use- Case Diagram for ATM Management System

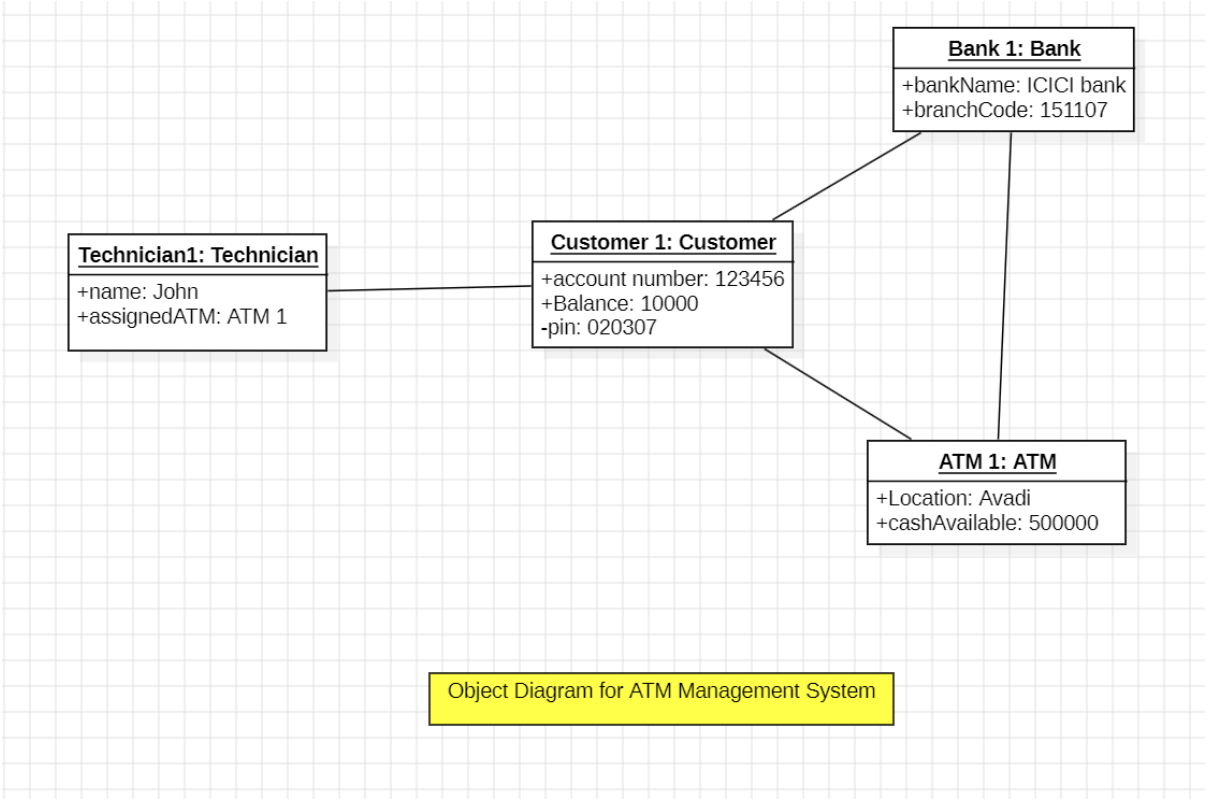
2.b) Class Diagram:



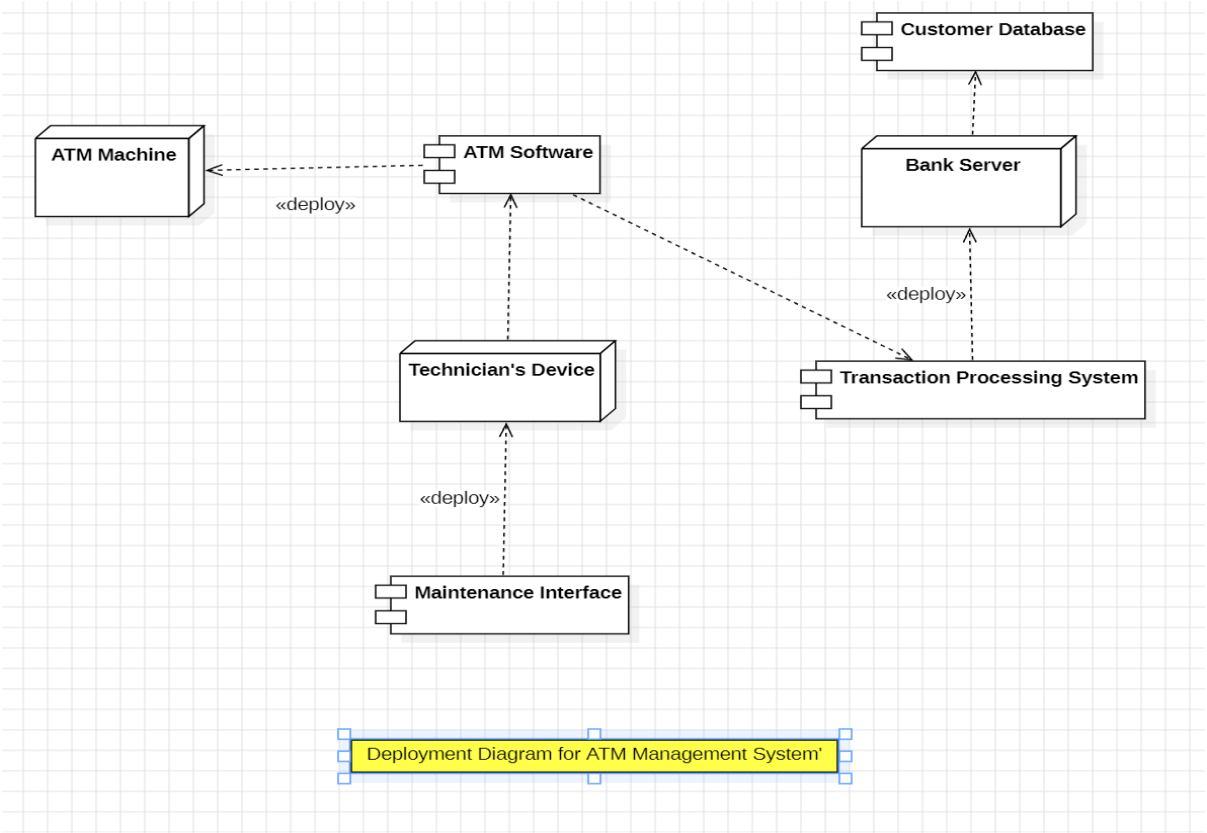
2. c) Sequence Diagram:



2.d) Object Diagram:



2.e) Deployment Diagram:



3.Basic Java Programs

3. a) Largest Number:

Code:

```java
public class LargestNumber {
    public static void main(String[] args) {
        int a = 10, b = 20, c = 15;

        if (a > b && a > c) {
            System.out.println("Largest: " + a);
        } else if (b > a && b > c) {
            System.out.println("Largest: " + b);
        } else {
            System.out.println("Largest: " + c);
        }
    }
}

```

Output:

```
PS D:\oops> javac LargestNumber.java
PS D:\oops> java LargestNumber
Largest: 20
PS D:\oops>
```

3.b) Number Check:

Code:

```java
import java.util.Scanner;

public class NumberCheck {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print(s:"Enter a number: ");
        int num = sc.nextInt();

        if (num > 0) {
            System.out.println(x:"Positive number");
        } else if (num < 0) {
            System.out.println(x:"Negative number");
        } else {
            System.out.println(x:"Zero");
        }
    }
}
```

Output:

```
PS D:\oops> javac NumberCheck.java
PS D:\oops> java NumberCheck
Enter a number: 7
Positive number
PS D:\oops>
```

3.c) Even or Odd:

Code:

```java
public class EvenOdd {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int num = 15;
        if (num % 2 == 0) {
            System.out.println(num + " is Even.");
        } else {
            System.out.println(num + " is Odd.");
        }
    }
}
```

Output:

```
PS D:\oops> javac EvenOdd.java
PS D:\oops> java EvenOdd
15 is Odd.
PS D:\oops>
```

3.d) Print Numbers:

Code:

```java
public class PrintNumbers {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        for (int i = 1; i <= 10; i++) {
            System.out.print(i + " ");
        }
    }
}
```

Output:

```
PS D:\oops> javac PrintNumbers.java
PS D:\oops> java PrintNumbers
1 2 3 4 5 6 7 8 9 10
PS D:\oops>
```

3.e) Factorial:

Code:

```java
public class Factorial {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int num = 5, fact = 1;
        for (int i = 1; i <= num; i++) {
            fact *= i;
        }
        System.out.println("Factorial: " + fact);
    }
}
```

}

Output:

```
PS D:\oops> javac Factorial.java
PS D:\oops> java Factorial
Factorial: 120
PS D:\oops>
```

3.f) While:

Code:

```
public class While {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int i = 1;
        while (i <= 10) {
            System.out.print(i + " ");
            i++;
        }
    }
}
```

Output:

```
PS D:\oops> javac While.java
PS D:\oops> java While
1 2 3 4 5 6 7 8 9 10
PS D:\oops>
```

3.g) Sum Natural Numbers:

Code:

```java
public class SumNaturalNumbers {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int n = 10, sum = 0, i = 1;
        while (i <= n) {
            sum += i;
            i++;
        }
        System.out.println("Sum: " + sum);
    }
}
```

Output:

```
PS D:\oops> javac SumNaturalNumbers.java
PS D:\oops> java SumNaturalNumbers
Sum: 55
PS D:\oops>
```

3.h) Reverse Numbers:

Code:

```
public class ReverseNumbers {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        for (int i = 10; i >= 1; i--) {
            System.out.print(i + " ");
        }
    }
}
```

Output:

```
PS D:\oops> javac ReverseNumbers.java
PS D:\oops> java ReverseNumbers
10 9 8 7 6 5 4 3 2 1
PS D:\oops>
```

3.i) Sum Of Digits:

Code:

```java
public class SumOfDigits {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int num = 1234, sum = 0;
        while (num > 0) {
            sum += num % 10;
            num /= 10;
        }
        System.out.println("Sum of digits: " + sum);
    }
}
```

Output:

```
PS D:\oops> javac SumOfDigits.java
PS D:\oops> java SumOfDigits
Sum of digits: 10
PS D:\oops>
```

3.j) Even Numbers:

Code:

```java
public class EvenNumbers {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int i = 2;
        while (i <= 20) {
            System.out.print(i + " ");
            i += 2;
        }
    }
}
```

Output:

```
PS D:\oops> javac EvenNumbers.java
PS D:\oops> java EvenNumbers
2 4 6 8 10 12 14 16 18 20
PS D:\oops>
```