

## Exp 9

### E-BOOK MANAGEMENT SYSTEM

#### AIM:

To draw the diagrams [usecase, activity, sequence, collaboration, class, statechart, component, deployment, package ] for E-book management system.

### SOFTWARE REQUIREMENTS SPECIFICATION

#### SL.NO SOFTWARE REQUIREMENTS SPECIFICATION

##### 1.0 Hardware Requirements

##### 1.1 Software Requirements

##### 1.2 Problem Analysis and Project Plan

##### 1.3 Project description

##### 1.4 Reference

#### 1.0 HARDWARE REQUIREMENTS:

Intel Pentium Processor I3/I5

#### 1.1 SOFTWARE REQUIREMENTS:

## Rational rose / Argo UML 1.2 PROBLEM ANALYSIS AND PROJECT PLANNING

E-book Management System gives an idea about how books are maintained in the

particular websites. The books that are to be purchased, the books that are to be sold are maintained

here. . Further some additional details of the current books that is available in the store are also

given. E book Management System in this project is done in an authorized way. The password and

user id has been set here.

### 1.3 PROJECT DESCRIPTION:

This software is designed to manage the books that were read through the internet. This

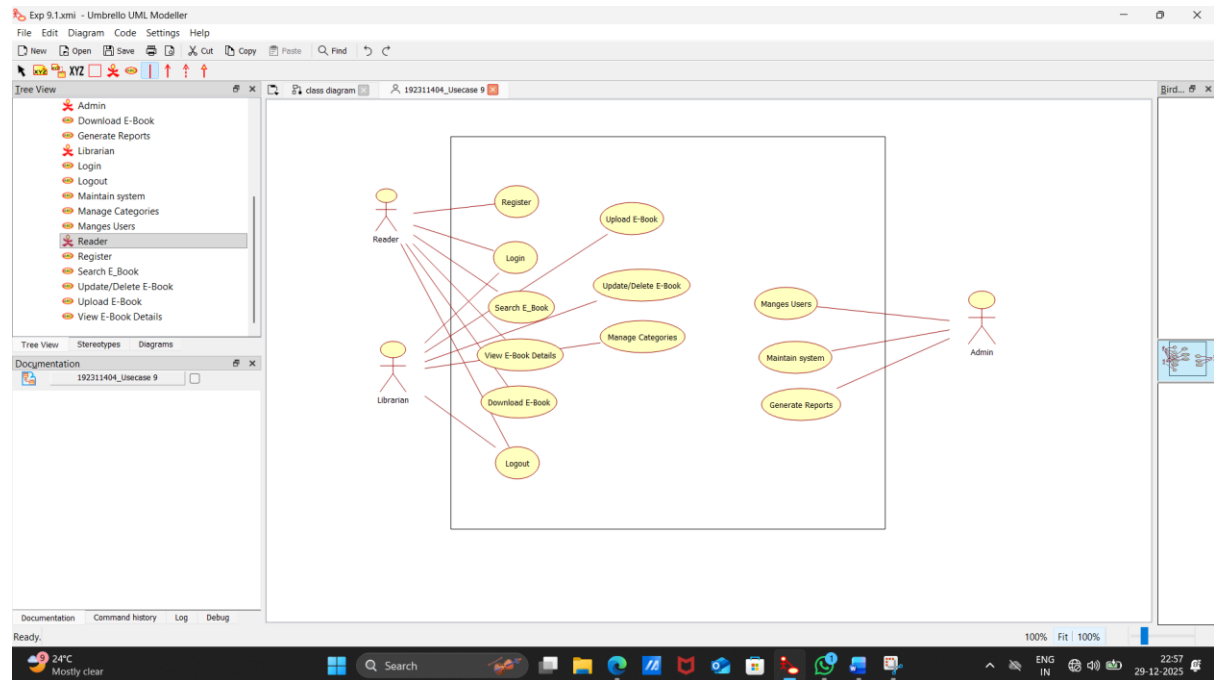
consists of the details of the e-book that were read by the user online. It will be controlled by the

central system. This system act as a backup of all details together.

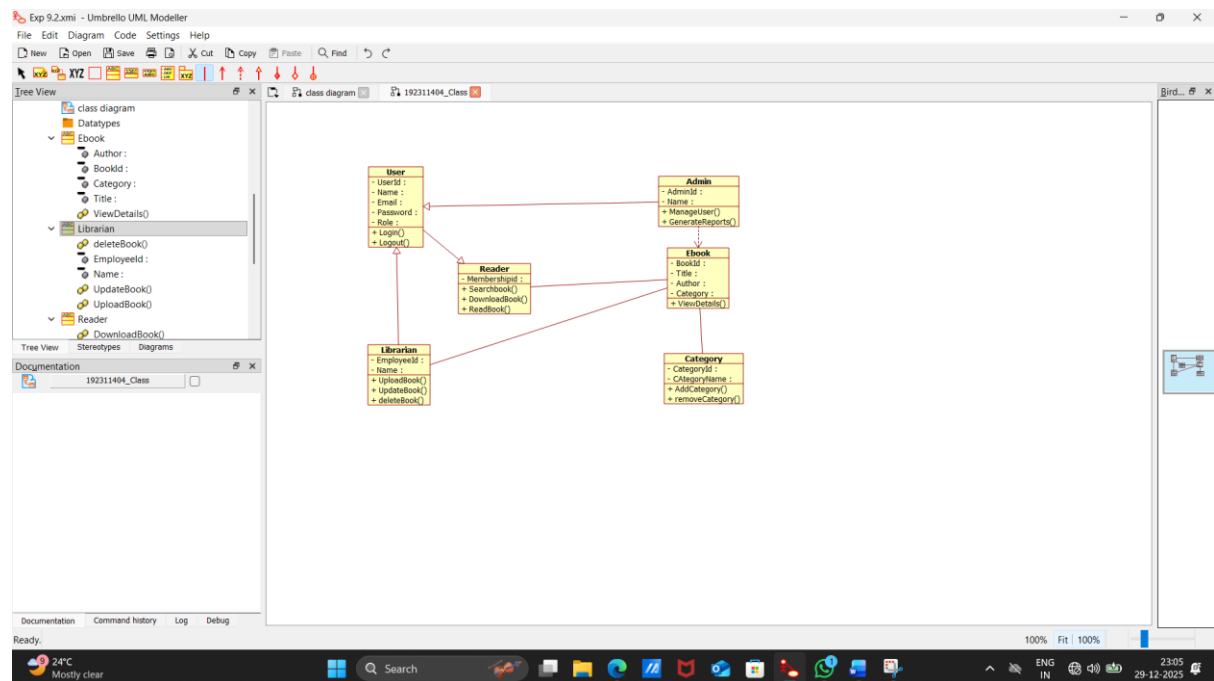
### 1.4 REFERENCES:

IEEE Software Requirement Specification format.

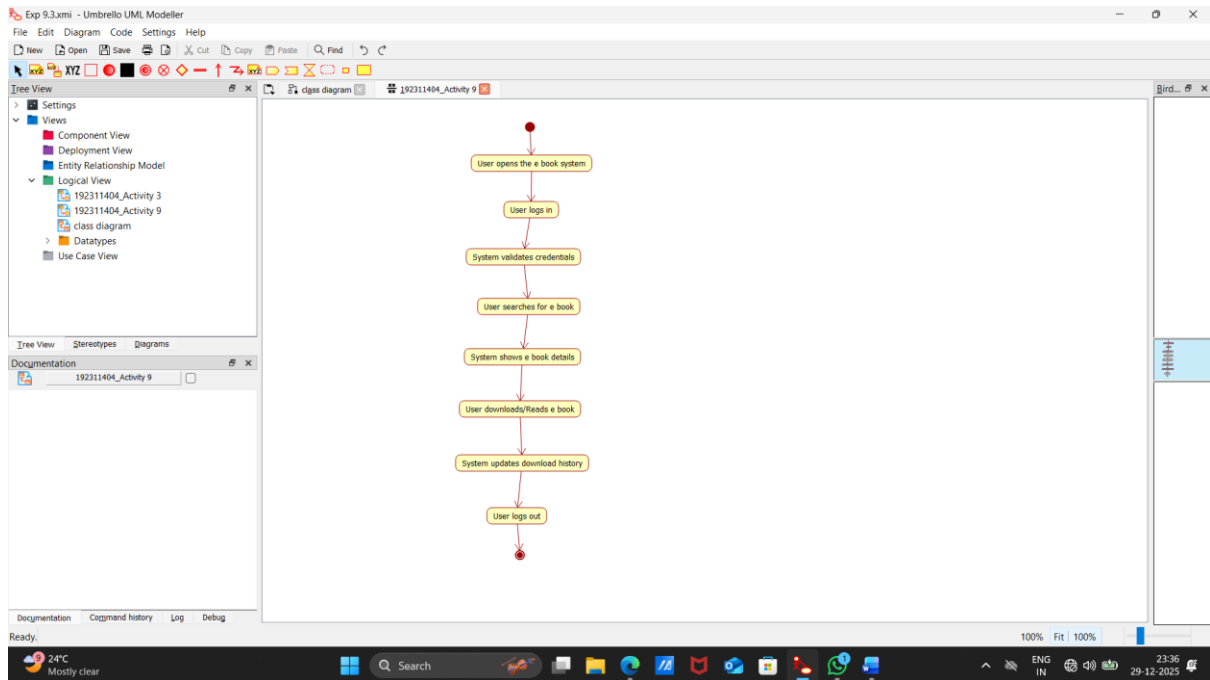
# Usecase Diagram:



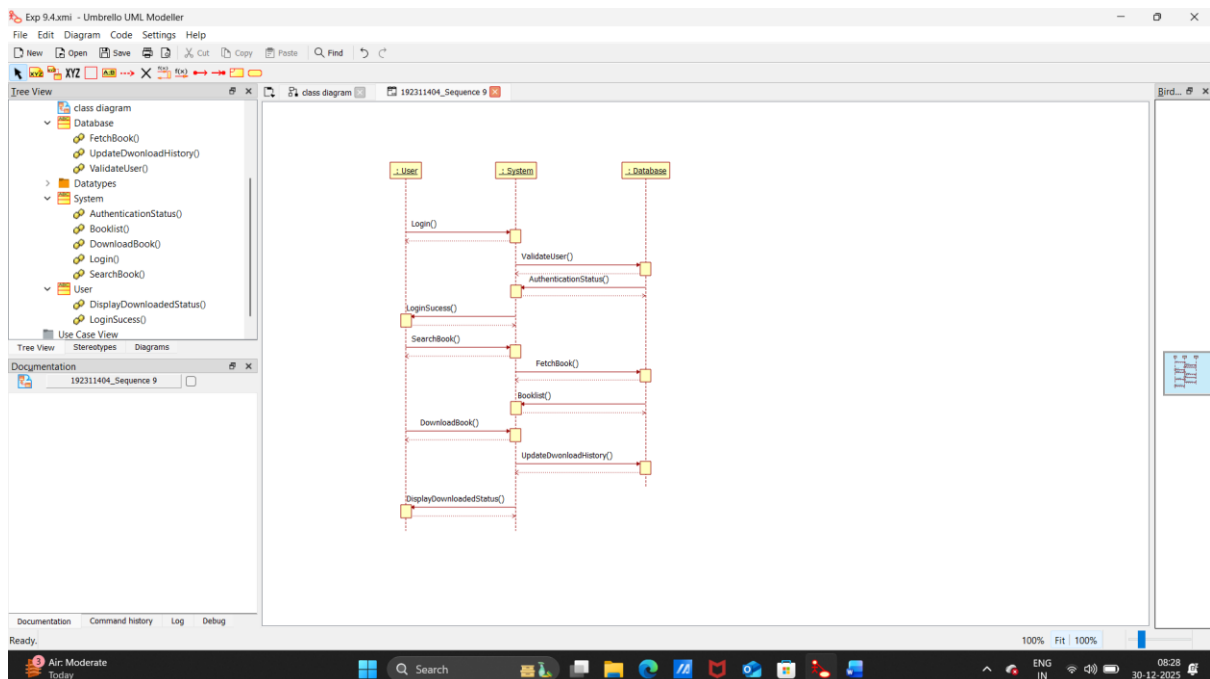
# Class Diagram:



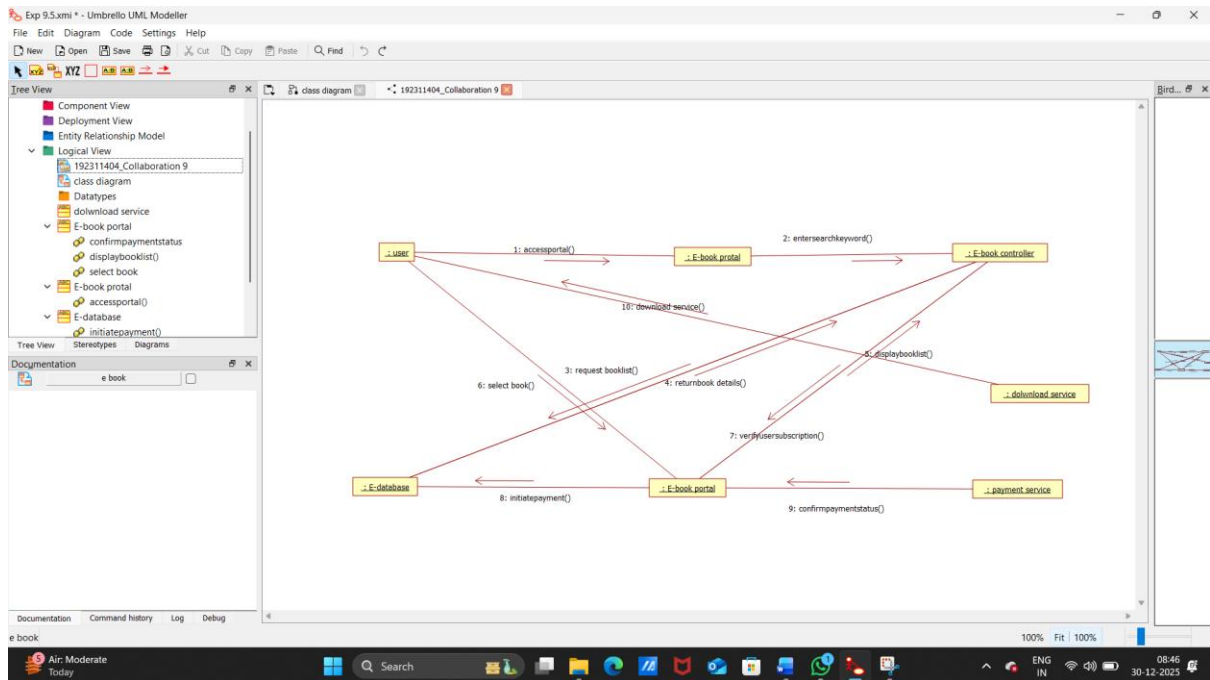
## Activity Diagram:



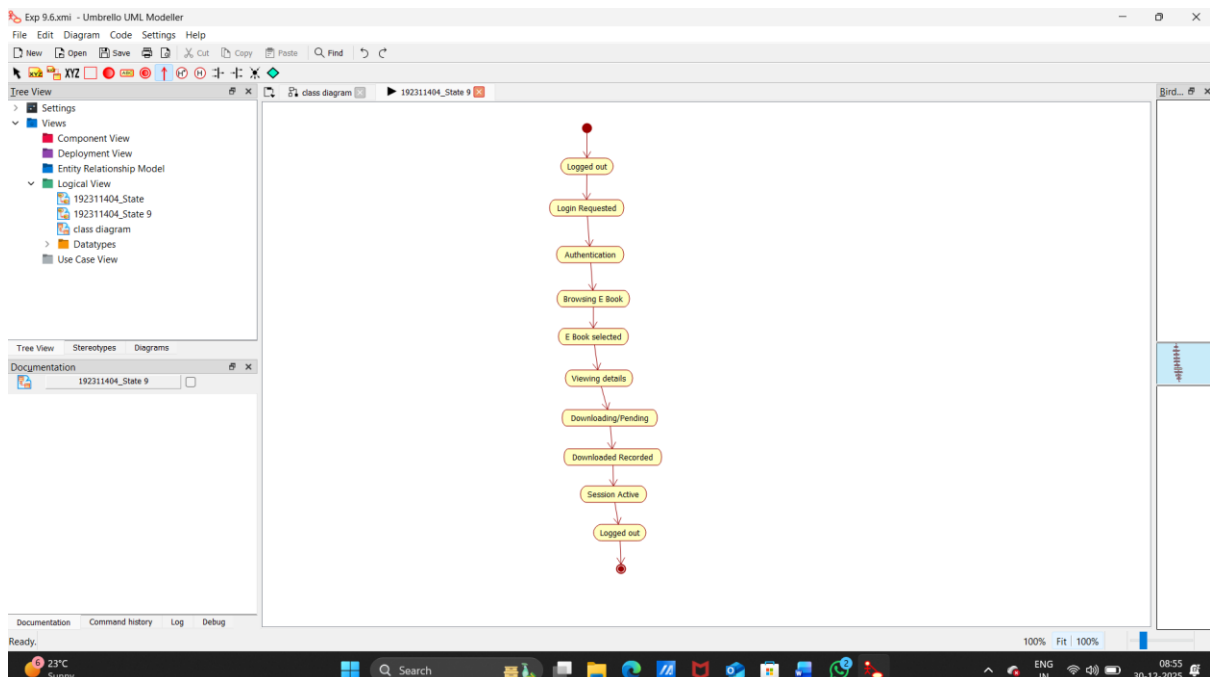
## Sequence Diagram:



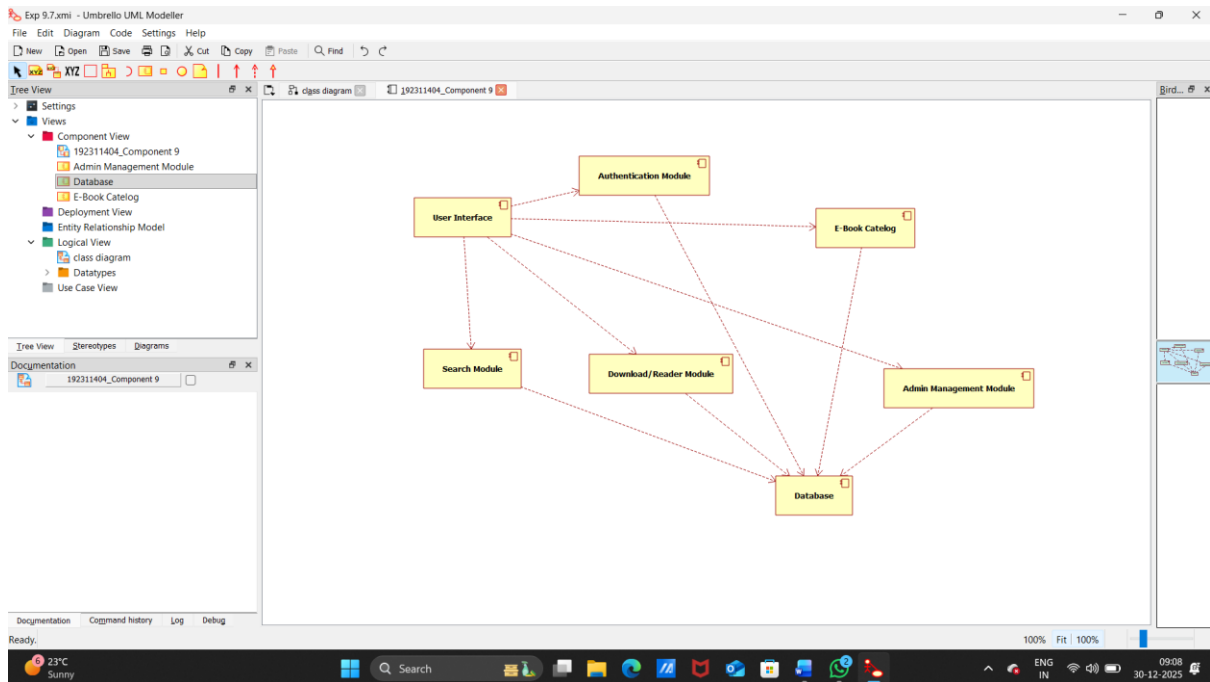
## Collaboration Diagram:



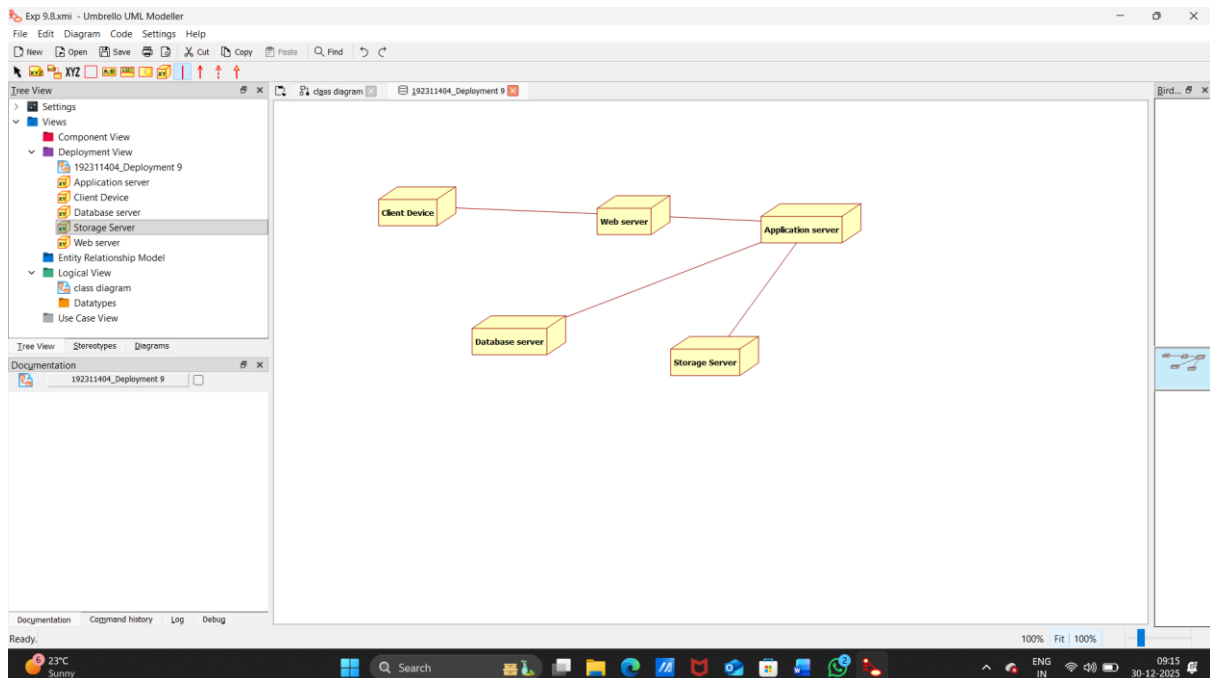
## State Diagram:



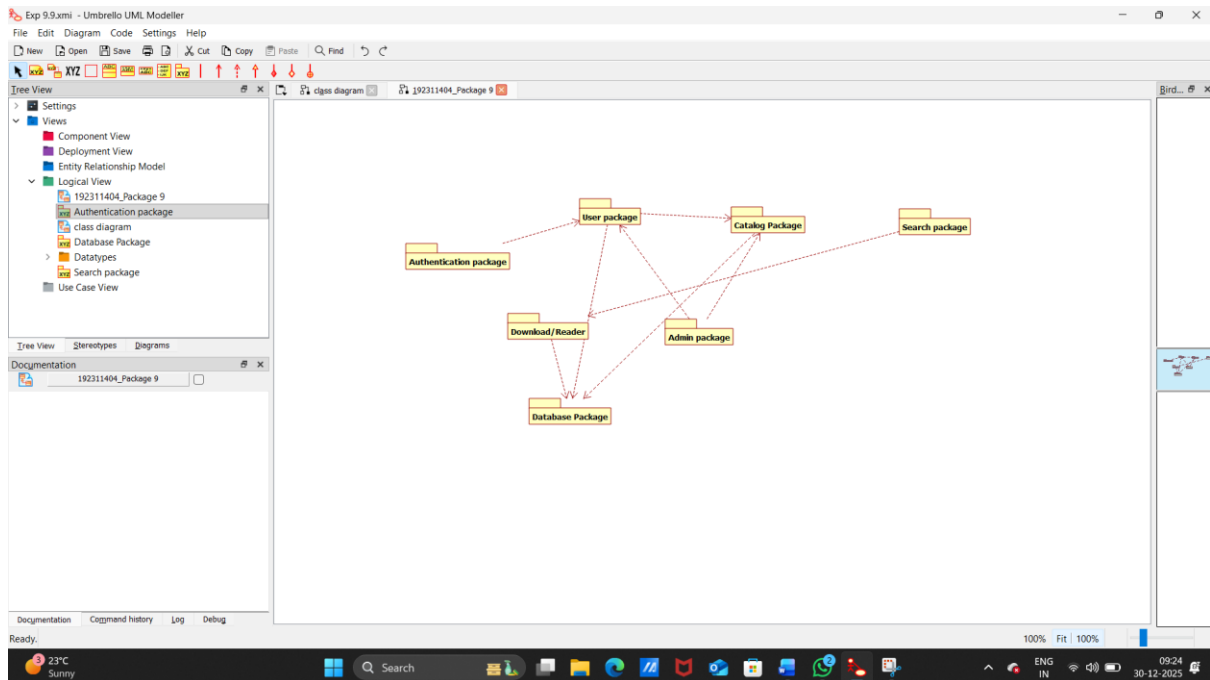
## Component Diagram:



## Deployment Diagram:



## Package Diagram:



## PROGRAM CODING:

### E-BOOK MANAGEMENT:

public class e-book management extends internet

{

public Integer enterid;

public Integer verifyuser;

public void checkavailability()

{

}

```
}
```

USER:

```
public class user1 extends internet
```

```
{
```

```
public Integer login;
```

```
public Integer logout;
```

```
public void surfbooks()
```

```
{
```

```
}
```

```
}
```

CENTRAL SYSTEM:

```
public class central system
```

```
{
```



```
public Integer enterid;
```

```
public Integer download;
```

```
public Integer login;
```

```
public Integer logout;
```

```
    public void verify()  
{  
    }  
}
```

```
    public void status()  
{  
    }  
}
```

```
}
```

RESULT: Thus the draw the diagrams [usecase, activity, sequence, collaboration, class, statechart, component, deployment, package ] for E-book management system.