192372243

EX NO:6	E-TICKETING
DATE:	

AIM:

To draw the diagrams[use case, activity, sequence, collaboration, class, state chart, component, deployment, package] for the E-ticketing 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

HARDWARE REQUIREMENTS:

Intel Pentium Processor I3/I5

SOFTWARE REQUIREMENTS:

Rational rose / Argo UML

PROBLEM ANALYSIS AND PROJECT PLANNING

In the E-Ticketing system the main process is a applicant have to login the database then the database verifies that particular username and password then the user must fill the details about their personal details then selecting the flight and the database books the ticket then send it to the applicant then searching the flight or else cancelling the process

PROJECT DESCRIPTION:

This software is designed for supporting the computerized e-ticketing. This is widely used by the passenger for reserving the tickets for their travel. This E-ticketing is organized by the central system. The information is provided from the railway reservation system.

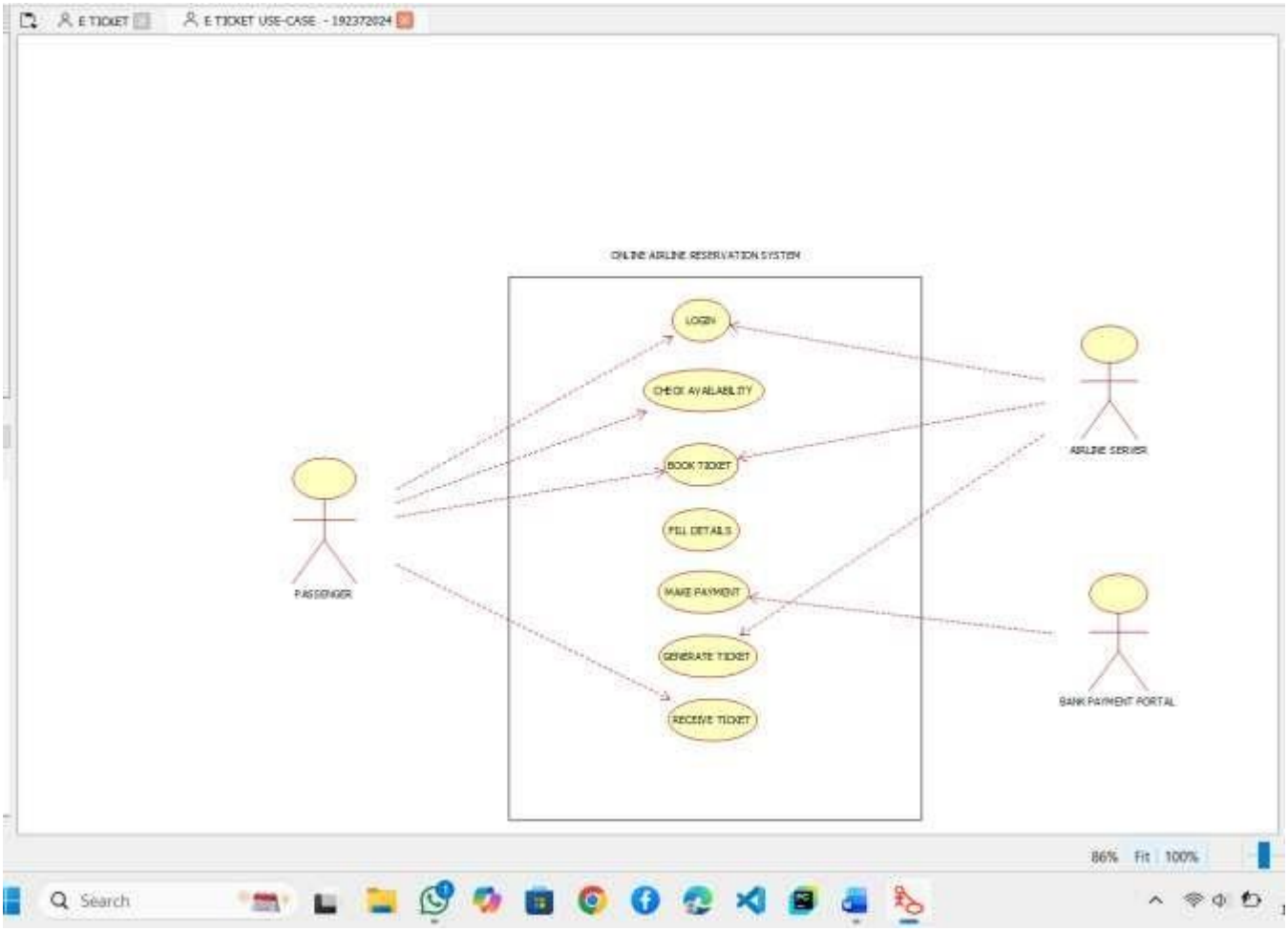
IEEE Software Requirement Specification format.

USE CASE DIAGRAM:

This diagram will contain the actors, use cases which are given below

Actors: Passenger, Railway reservation system..

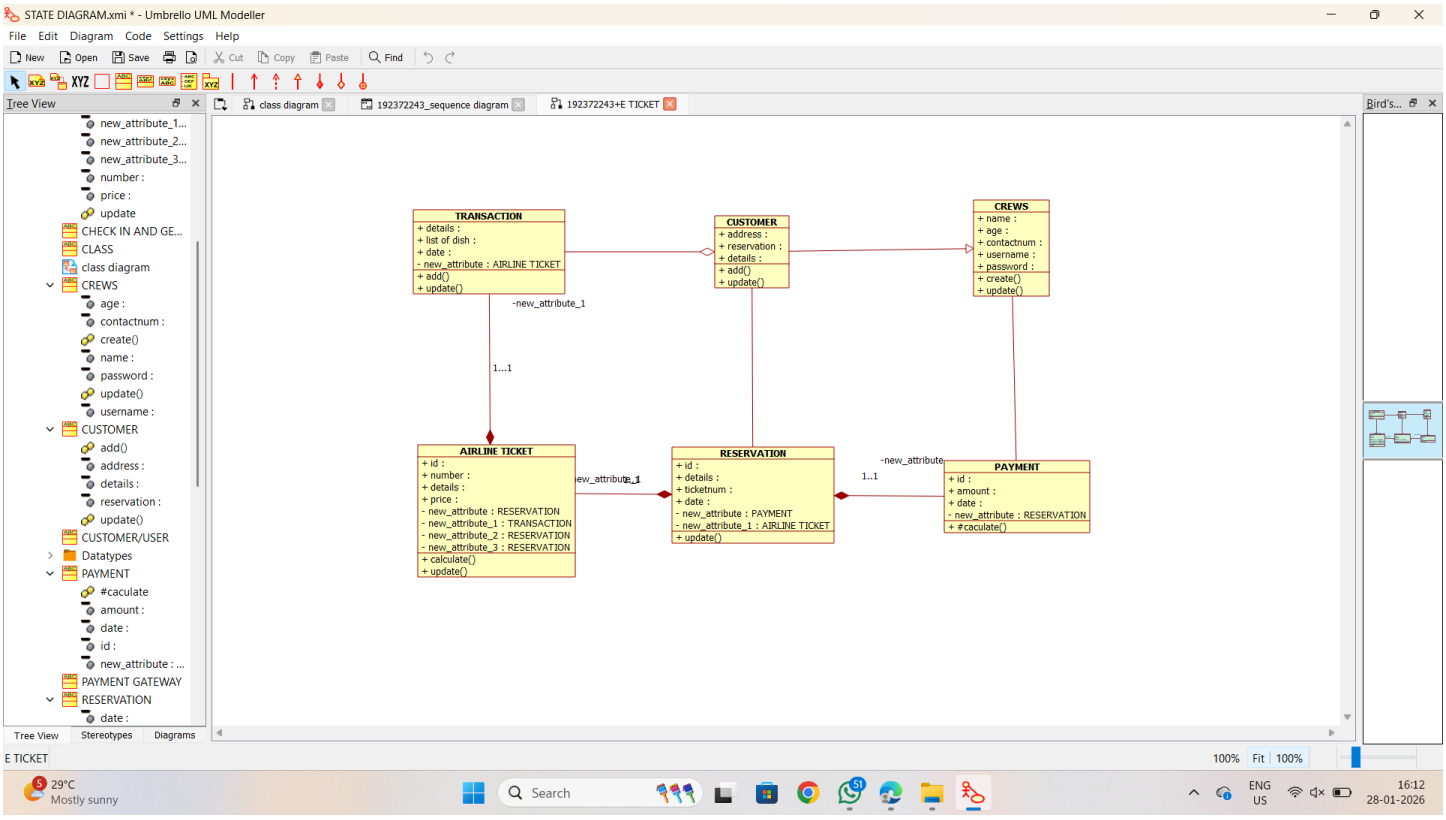
Use case: Status, reservation, cancellation, enter the train number, enter the number of seats, availability of seats, acceptance of ticket.



CLASS DIAGRAM:

This diagram consists of the following classes, attributes and their operations**.**

CLASSES	ATTRIBUTES	OPERATIONS
Central computer	Train name, Passenger name	Reservation(), login()
Passenger	Passenger age	Login()
Railway reservation system	Train number	Cancellation()

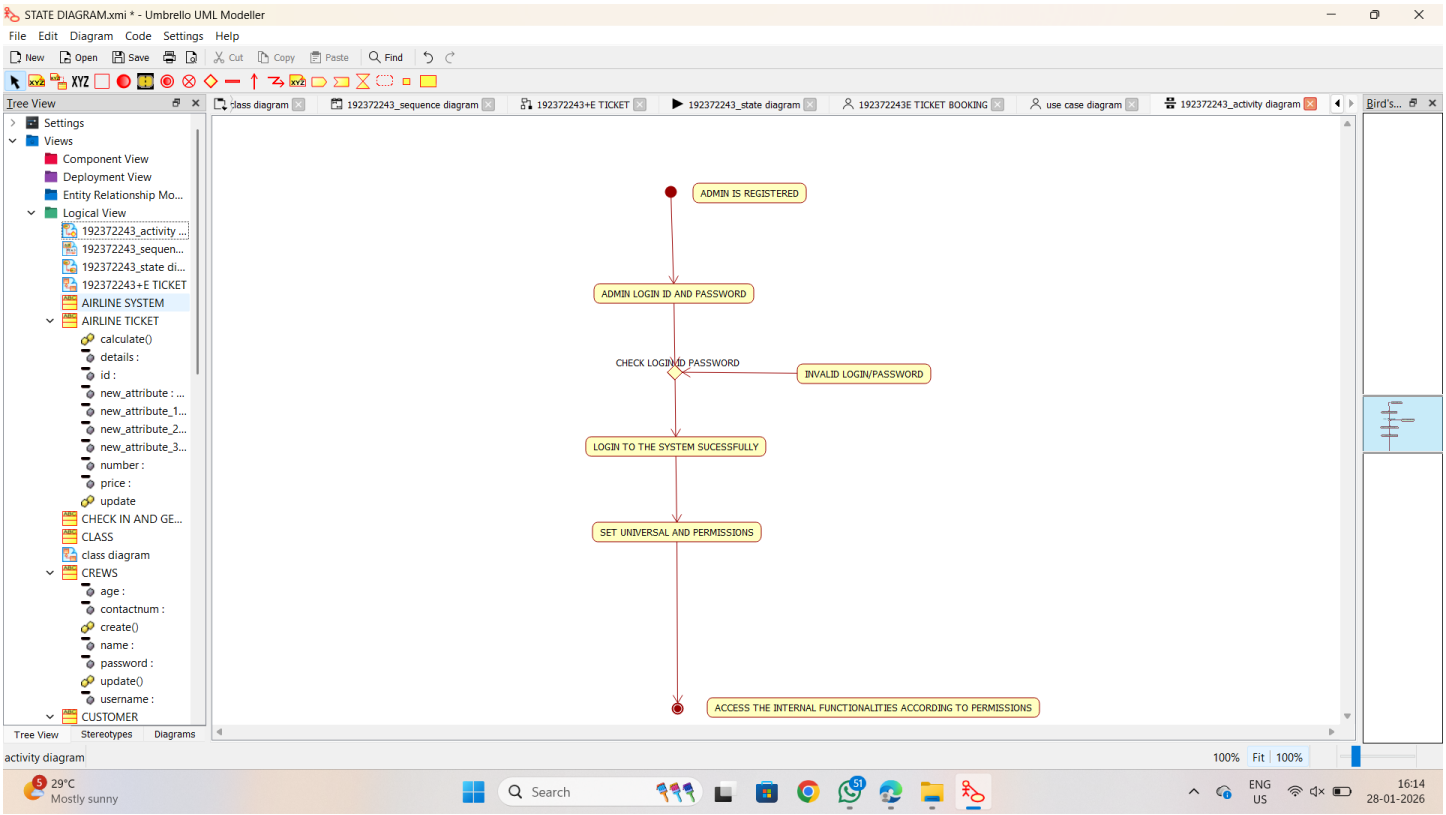


ACTIVITY DIAGRAM:

This diagram will have the activities as Start point, End point, Decision boxes as given below:

Activities: enter the train number, enter the number of seats, acceptance of ticket, accept seat.

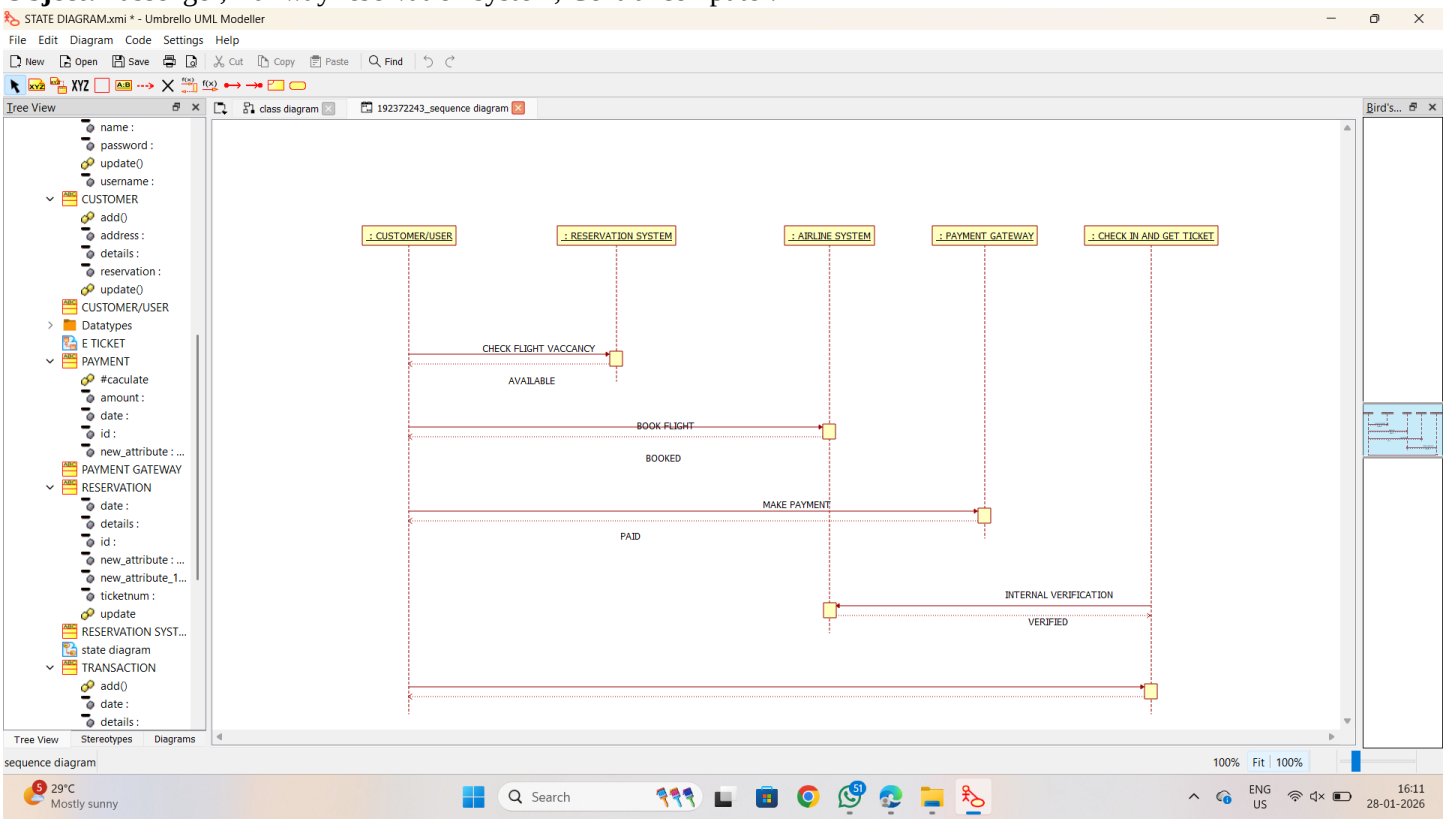
Decision box: Check availability of seats whether it is present or not.


STATE DIAGRAM.xmi * - Umbrello UML Modeller
File Edit Diagram Code Settings Help
New Open Save Cut Copy Paste Find
Tree View
Settings
Views
Component View
Deployment View
Entity Relationship Mo...
Logical View
192372243_activity ...
192372243_sequen...
192372243_state di...
192372243+E TICKET
AIRLINE SYSTEM
AIRLINE TICKET
calculate()
details :
id :
new_attribute : ...
new_attribute_1...
new_attribute_2...
new_attribute_3...
number :
price :
update
CHECK IN AND GE...
CLASS
class diagram
CREWS
age :
contactnum :
create()
name :
password :
update()
username :
CUSTOMER
Tree View Stereotypes Diagrams
class diagram
192372243_sequence diagram
192372243+E TICKET
192372243_state diagram
192372243E TICKET BOOKING
use case diagram
192372243_activity diagram
Bird's...
ADMIN IS REGISTERED
ADMIN LOGIN ID AND PASSWORD
CHECK LOGIN ID PASSWORD
INVALID LOGIN/PASSWORD
LOGIN TO THE SYSTEM SUCESSFULLY
SET UNIVERSAL AND PERMISSIONS
ACCESS THE INTERNAL FUNCTIONALITIES ACCORDING TO PERMISSIONS
activity diagram
100% Fit 100%
29°C
Mostly sunny
Search
ENG
US
16:14
28-01-2026

SEQUENCE DIAGRAM:

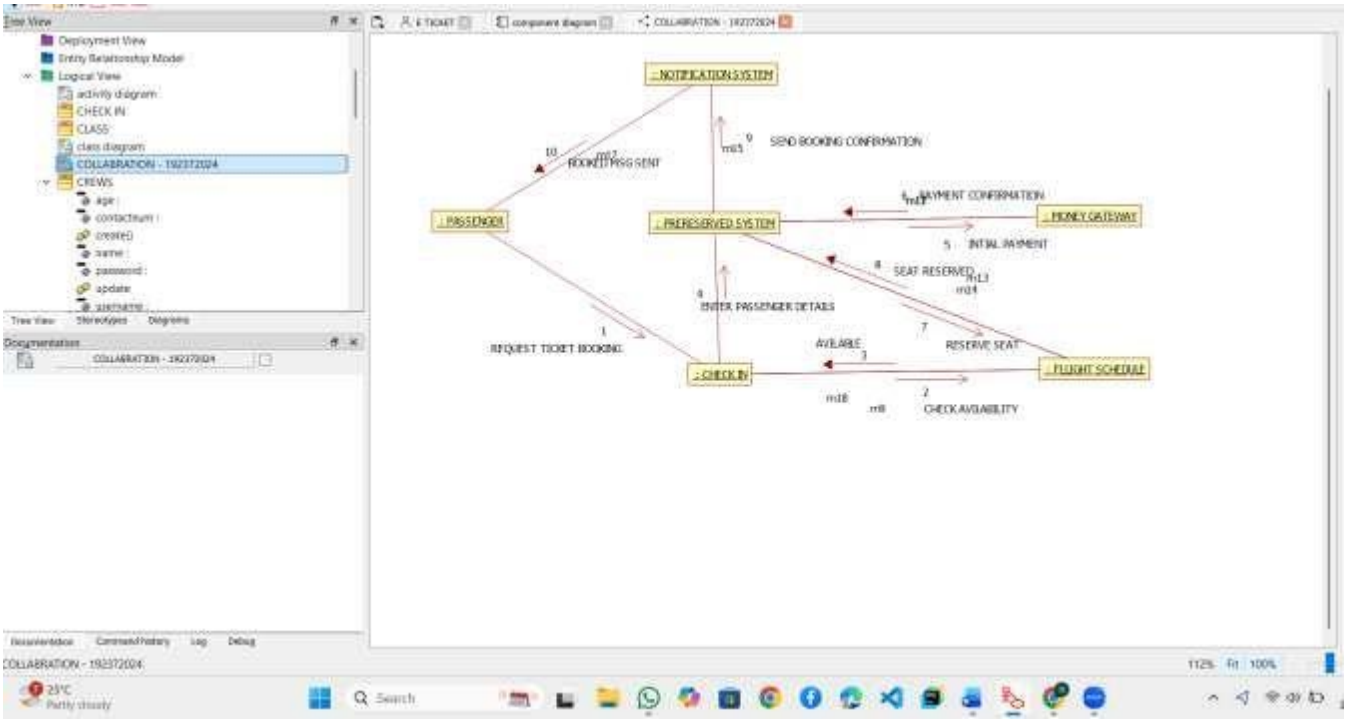
This diagram consists of the objects, messages and return messages.

Object: Passenger, Railway reservation system, Central computer.

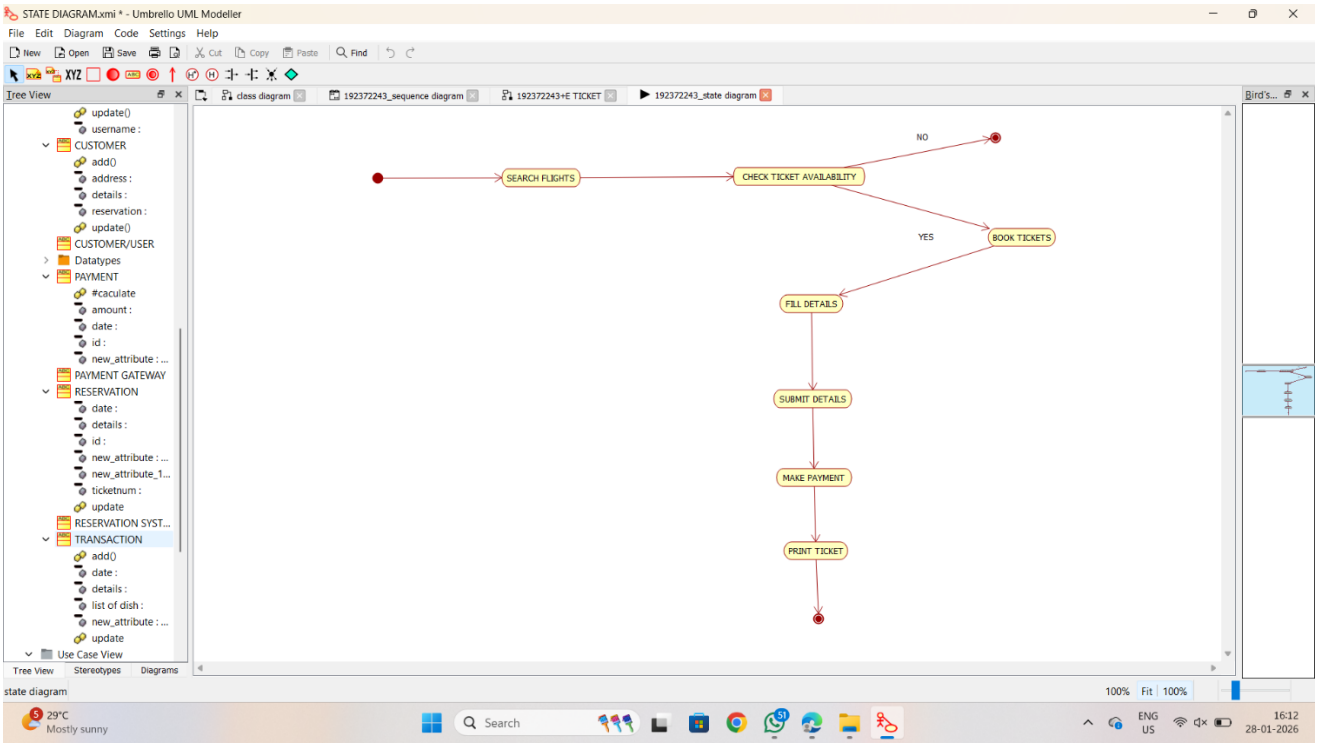


COLLABORATION DIAGRAM:

This diagram contains the objects and actors. This will be obtained by the completion of the sequence diagram and pressing the F5 key.



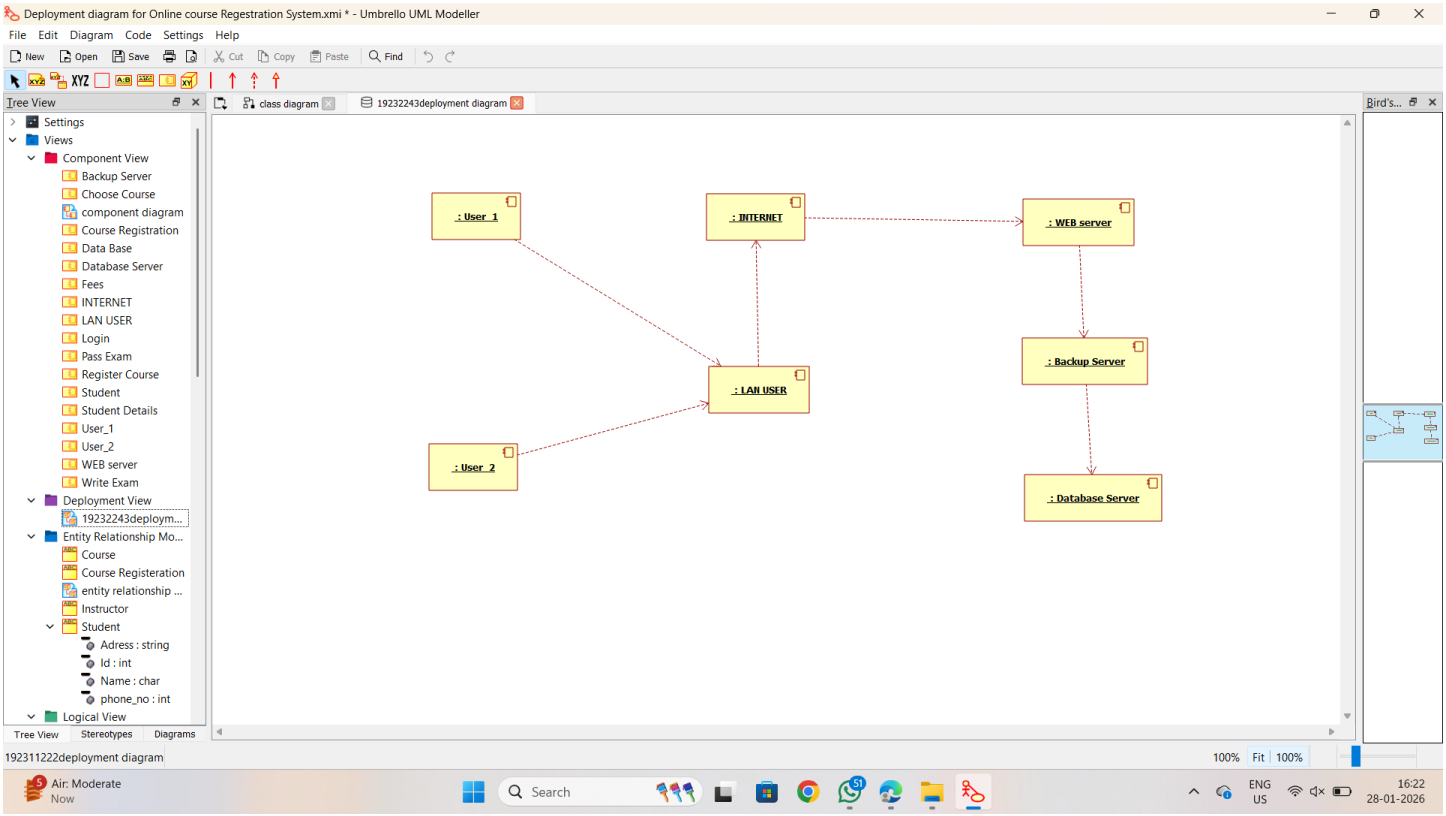
STATE CHART DIAGRAM:



It is a technique to describe the behavior of the system. It describes all the possible states that a particular object gets into the object oriented technique. State diagram are drawn for a single class to show to the lifetime behaviour of a single object

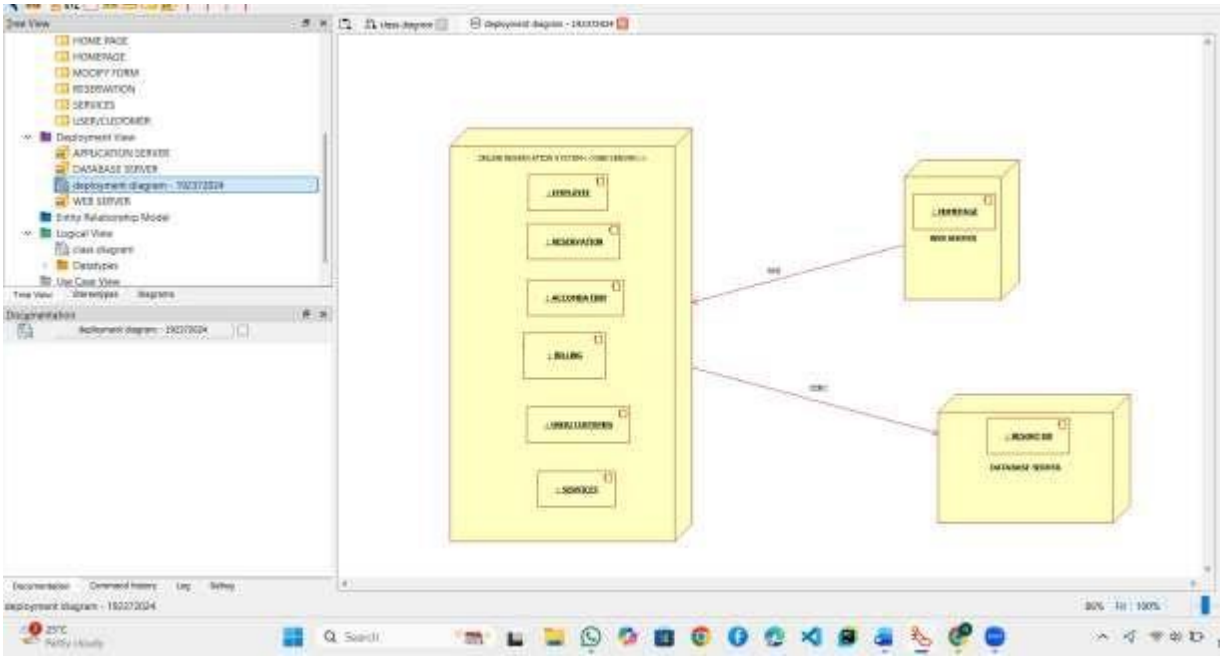
COMPONENT DIAGRAM:

The component diagram is represented by figure dependency and it is a graph of design of figure dependency. The component diagram's main purpose is to show the structural relationships between the components of a systems. It is represented by boxed figure. Dependencies are represented by communication association



DEPLOYMENT DIAGRAM:

A deployment diagram in the unified modeling language serves to model the physical deployment of artifacts on deployment targets. Deployment diagrams show "the allocation of artifacts to nodes according to the Deployments defined between them. It is represented by 3- dimentional box. Dependencies are represented by communication association.


Tree View
HOME PAGE
HOMEPAGE
MODIFY FORM
RESERVATION
SERVICES
USER/CUSTOMER
Deployment View
APPLICATION SERVER
DATABASE SERVER
deployment diagram - 192272024
WEB SERVER
Entity Relationship Model
Logical View
class diagram
Datatypes
Use Case View
Documentation
deployment diagram - 192272024
Search

PACKAGE DIAGRAM:

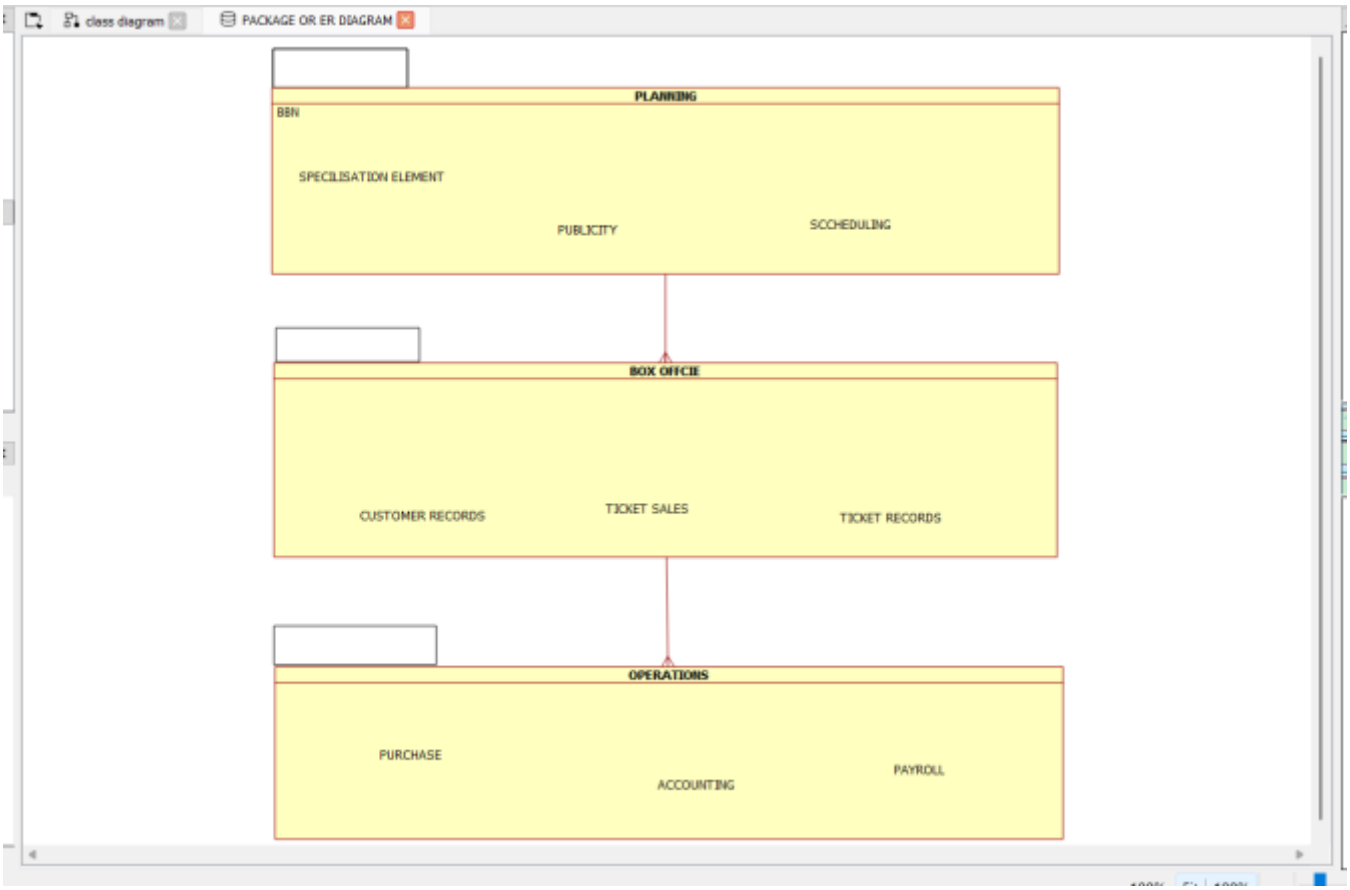
A package diagram in unified modeling language that depicts the dependencies between the packages that make up a model. A Package Diagram (PD) shows a grouping of elements in the OO model, and is a Cradle extension to UML. PDs can be used to show groups of classes in Class Diagrams (CDs), groups of components or processes in Component Diagrams (CPDs), or groups of processors in Deployment Diagrams (DPDs).

There are three types of layer. They are

User interface layer

Domain layer

Technical services layer



PROGRAM CODING:

PASSENGER:

Public class passenger

{

Public integer passenger passenger name; Public integer passenger passenger age; Public integer train no;

Public void passenger()

{

{

}

}

CENTRAL MANAGEMENT SYSTEM:

Public class central management

{

Public integer train name; Public integer passenger name; Public void reservation()

{

}

Public void cancellation()

{

}

Public void status()

{

}

Public void login()

{

}

Private void management()

{

}

}

RAILWAY RESERVATION SYSTEM:

Public class railway reservation system

{

Public integer trainno; Public integer train name;

Public integer passenger name; Public void status()

{

}

Public void reservation()

{

}

Public void cancellation()

{

}

Public void railway reservation system()

{

}

}

RESULT:

Thus the diagrams[use case, activity, sequence, collaboration, class, statechart, component, deployment, package] for the E-ticketing system has been designed, executed and output is verified.