Ex No:1	PASSPORT AUTOMATION SYSTEM
Date:	

AIM:

To draw the diagrams [usecase, activity, sequence, collaboration, class, statechart, collaboration, component, deployment, package] for the Passport Automation System.

SOFTWARE REQUIREMENTS SPECIFICATION

	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 PLAN

To simplify the process of applying passport, software has been created by designing through rational rose tool. Initially the applicant login the passport automation system and submits his details. These details are stored in the database and verification process done by the passport administrator, regional administrator and police the passport is issued to the applicant.

1.3 PROJECT DESCRIPTION:

This software is designed for the verification of the passport details of the applicant by the central computer. The details regarding the passport will be provided to the central computer and the computer will verify the details of applicant and provide approval to the office. Then the passport will issue from the office to the applicant.

1.4 REFERENCES:

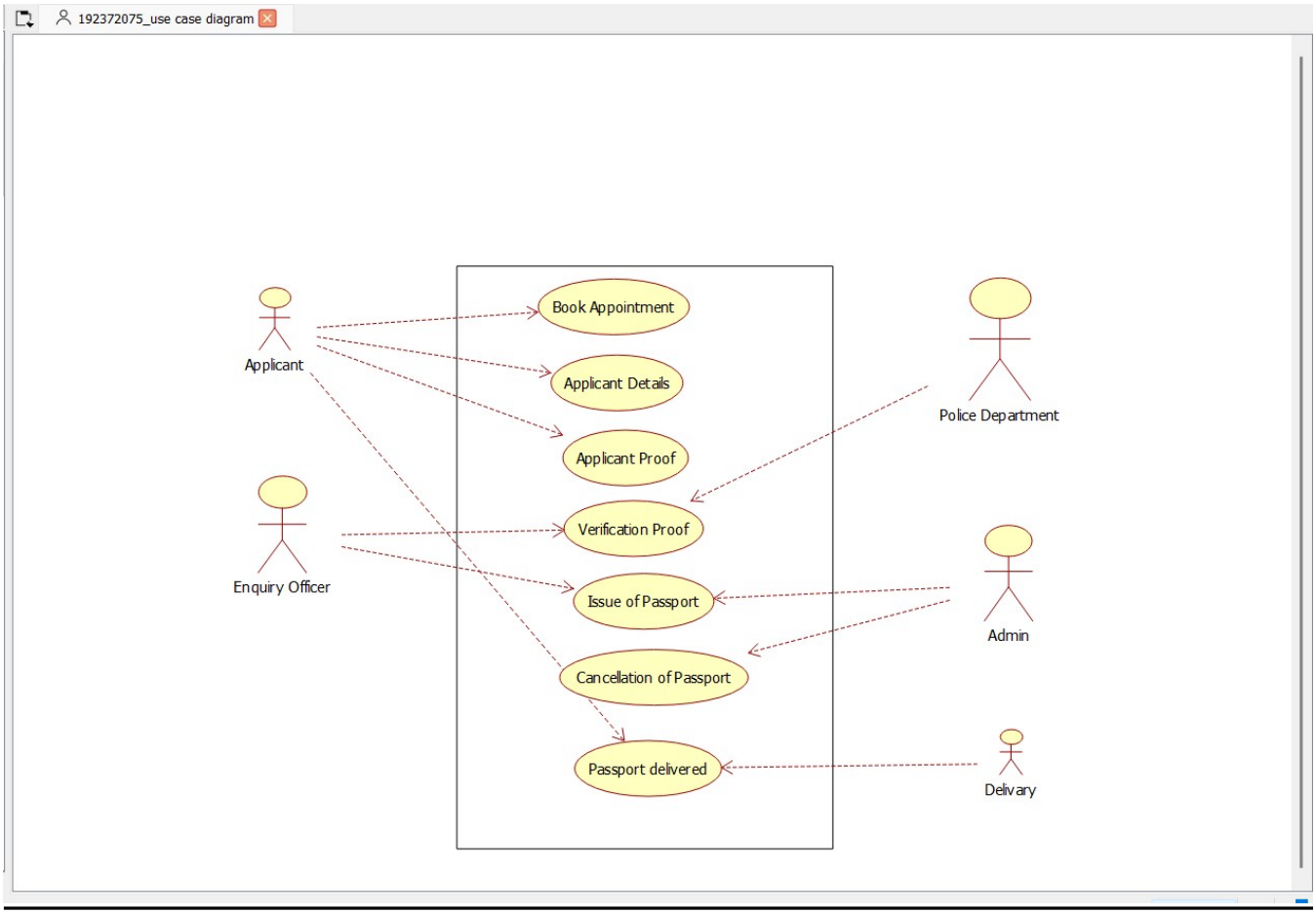
IEEE Software Requirement Specification format.

USE CASE DIAGRAM:

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

Actors: Applicant, Enquiry Officer.

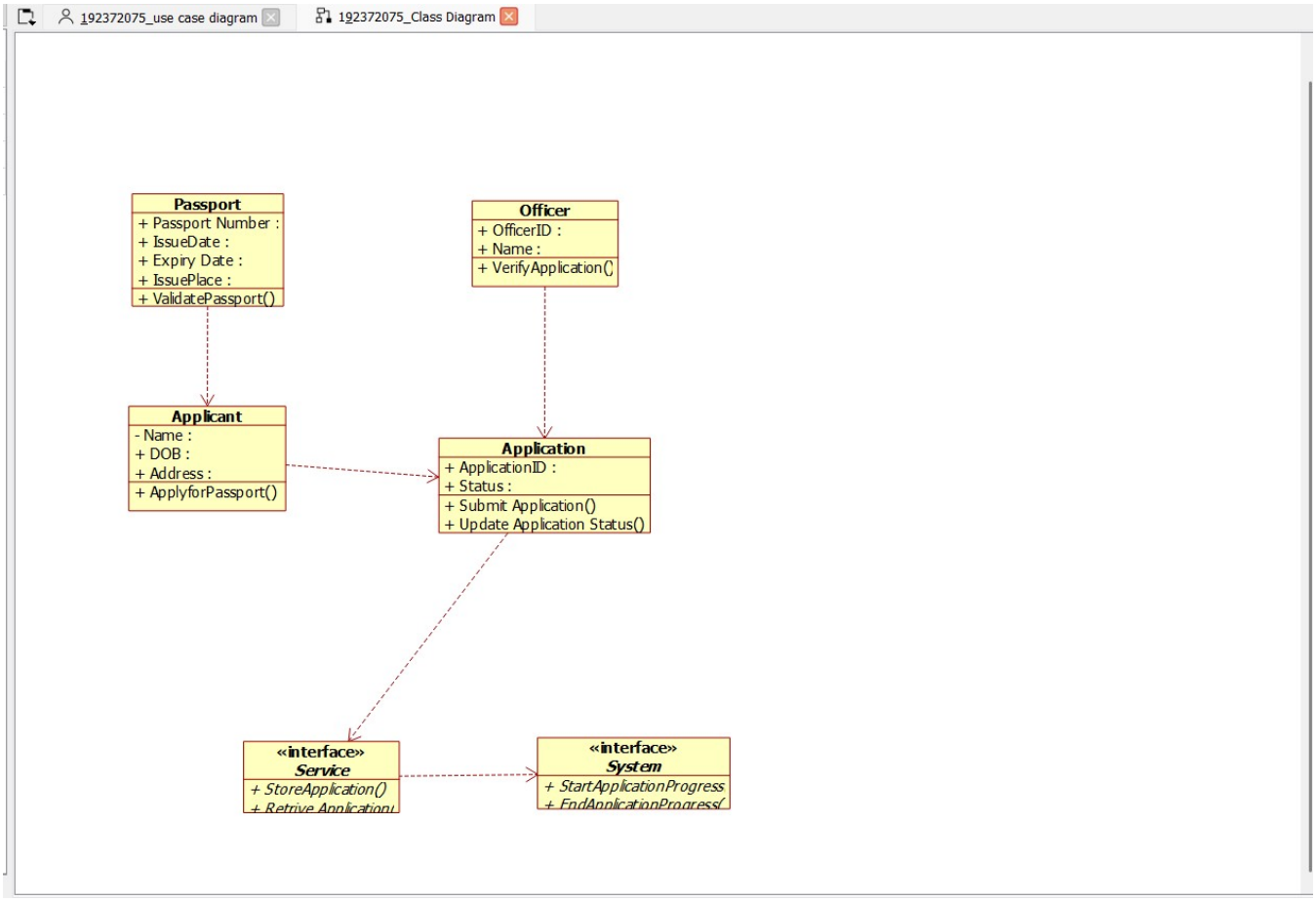
Use case: Applicant details, Applicant proof, Verification of proof, Issue of passport, Cancellation of the passport.



CLASS DIAGRAM:

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

CLASSES	ATTRIBTES	OPERATIONS
Passport management system	Verify details, Store proof	Verification of proof()
Enquiry officer	Applicant details	Issue of passport()
Applicant	Name, Details	Apply passport()

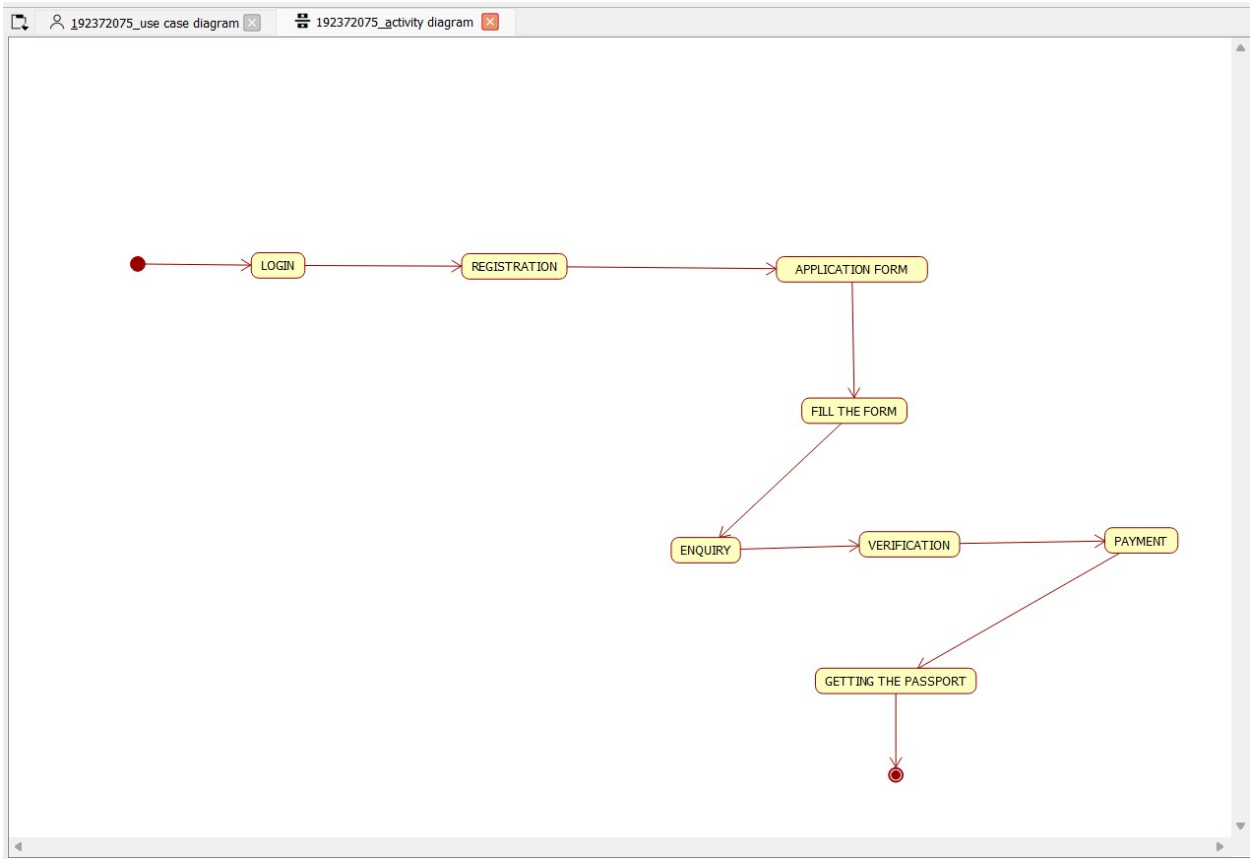


ACTIVITY DIAGRAM:

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

Activities: Enter applicant details, Submission of proof, Verification of details, Issue of passport.

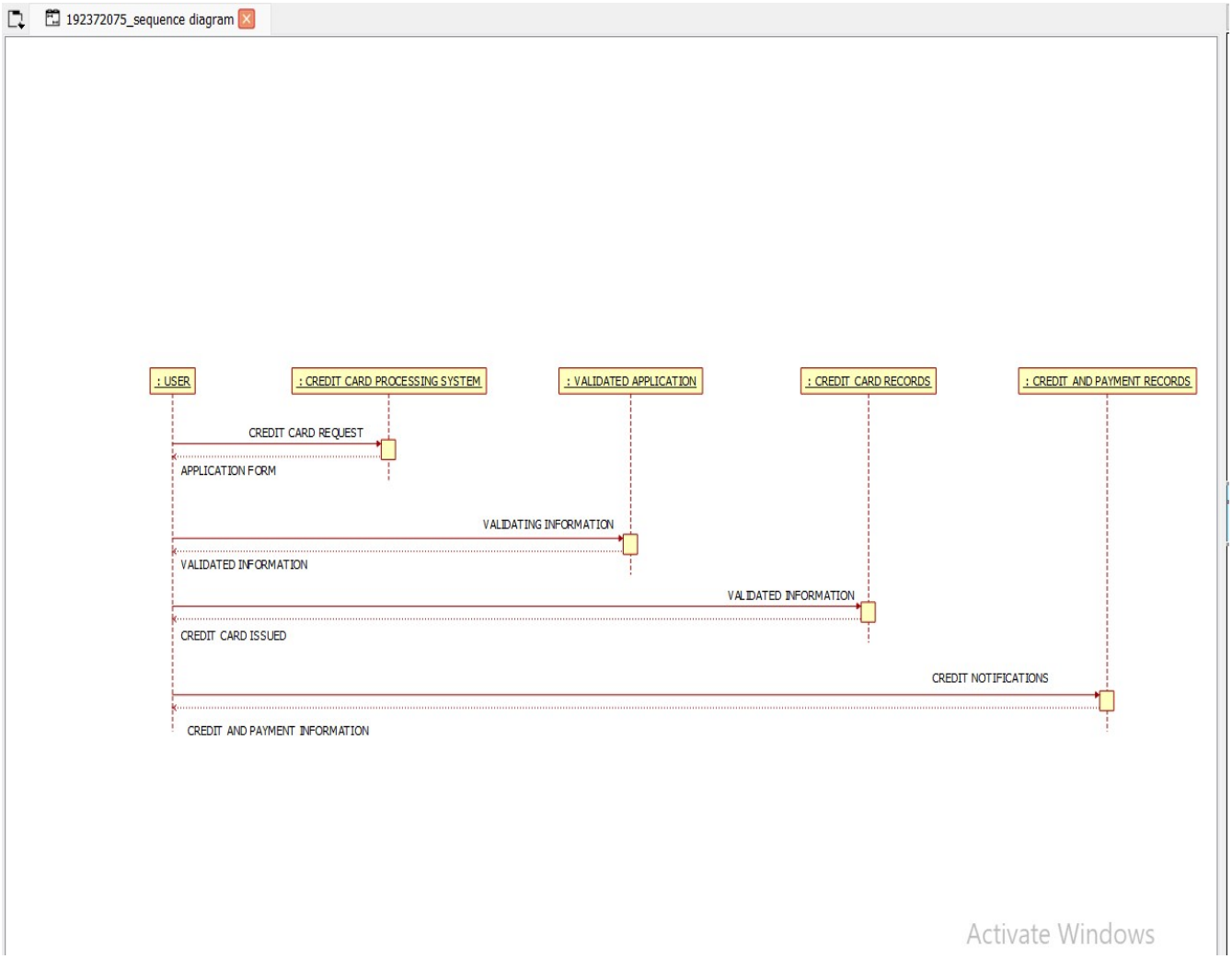
Decision box: Check details whether it is correct or not



SEQUENCE DIAGRAM:

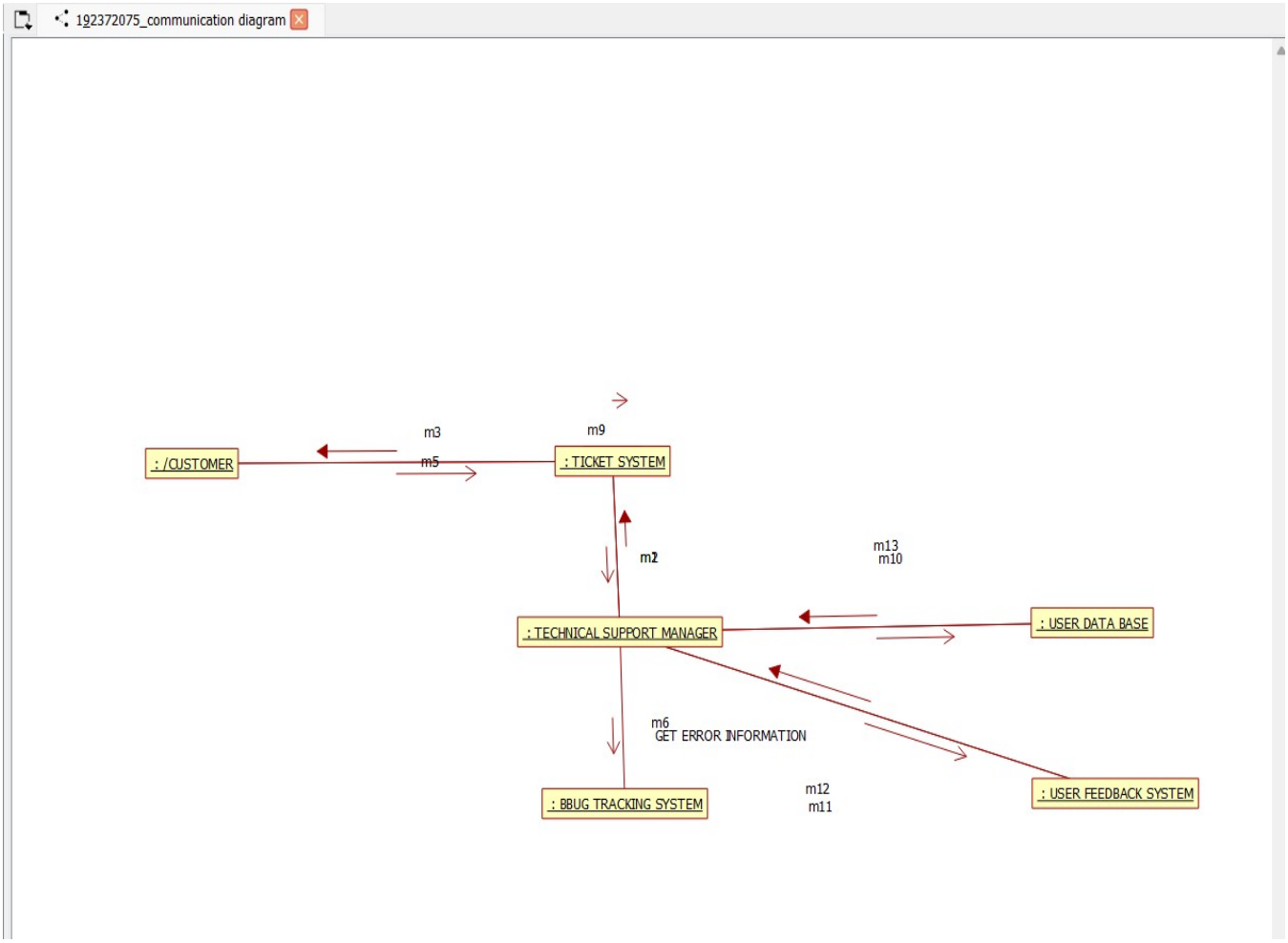
This diagram consists of the objects, messages and return messages.

Object: Applicant, Enquiry officer, Passport management system.



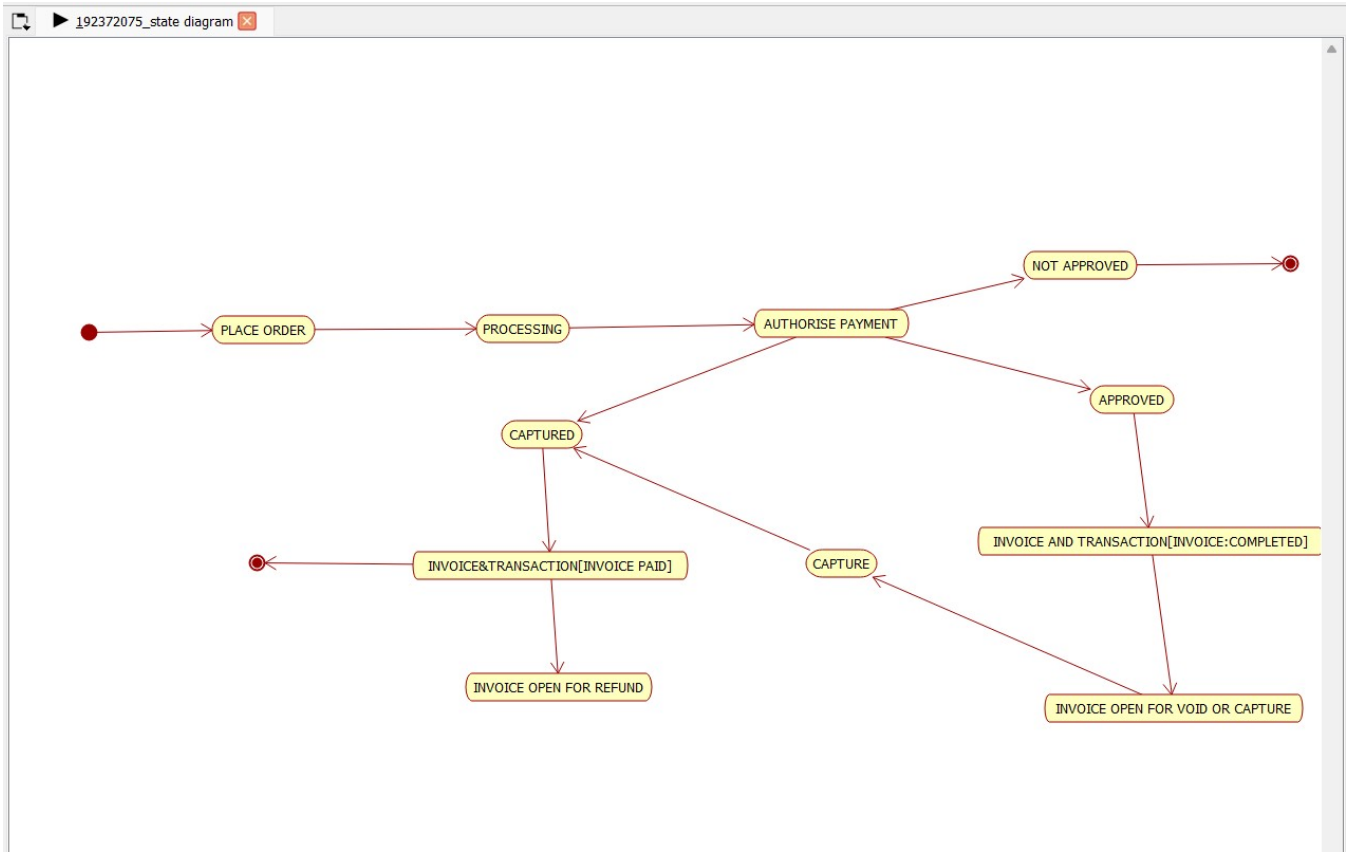
COLLABORATION DIAGRAM:

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



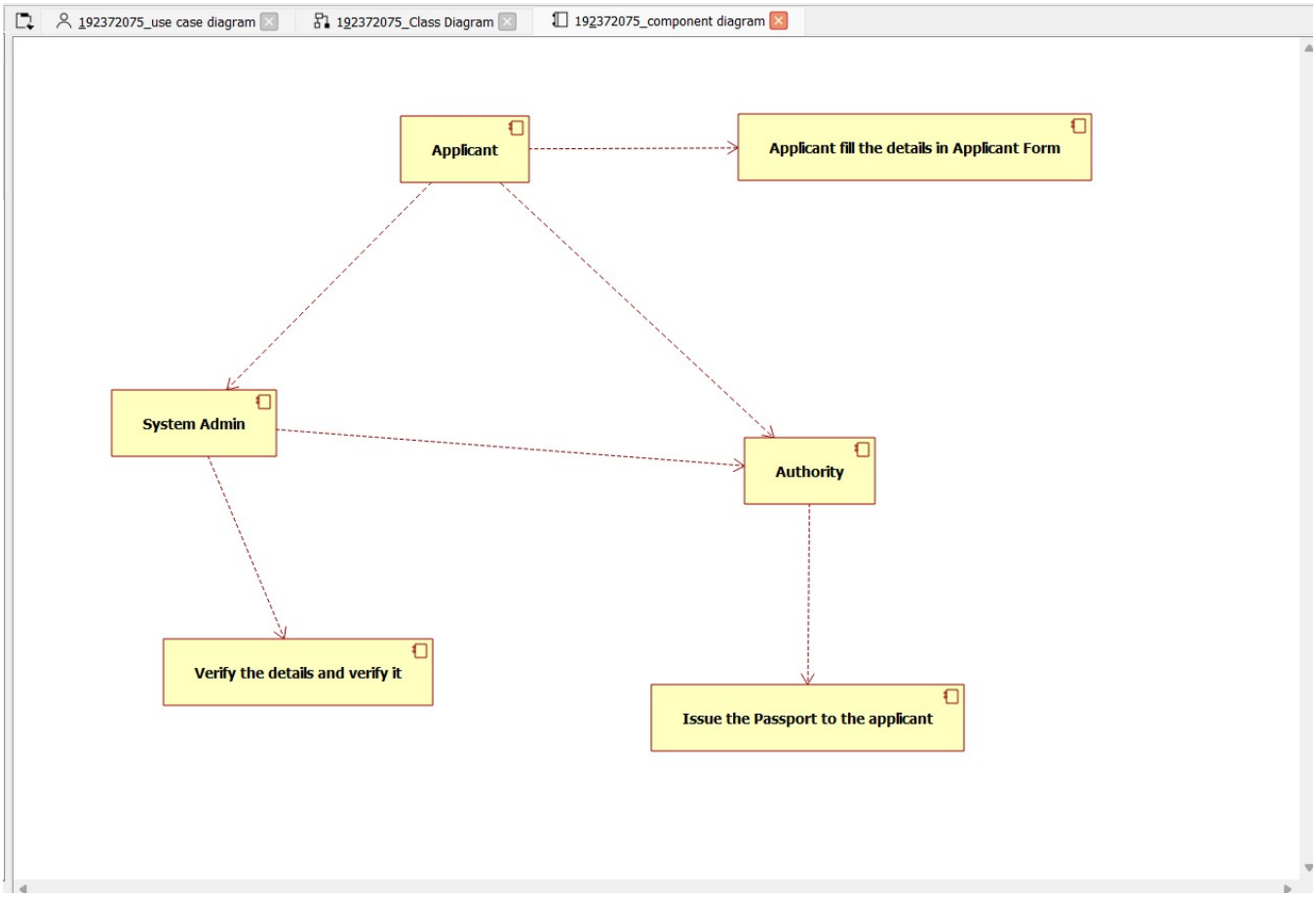
STATE CHART DIAGRAM :

The purpose of state chart diagram is to understand the algorithm involved in performing a method. It is also called as state diagram. A state is represented as a round box, which may contain one or more compartments. An initial state is represented as small dot. An final state is represented as circle surrounding a small dot.



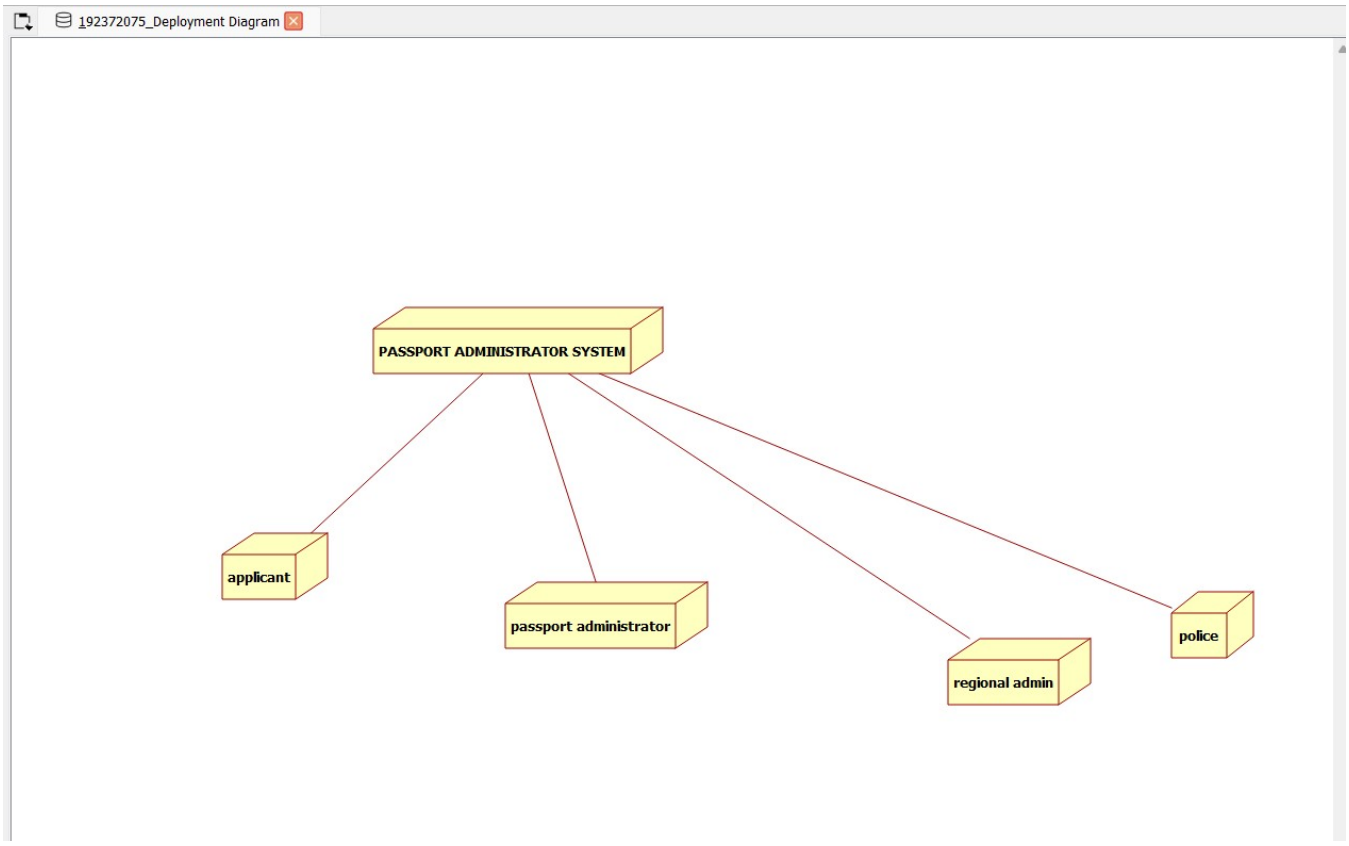
COMPONENT DIAGRAM

The component diagram's main purpose is to show the structural relationships between the components of a system. 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.

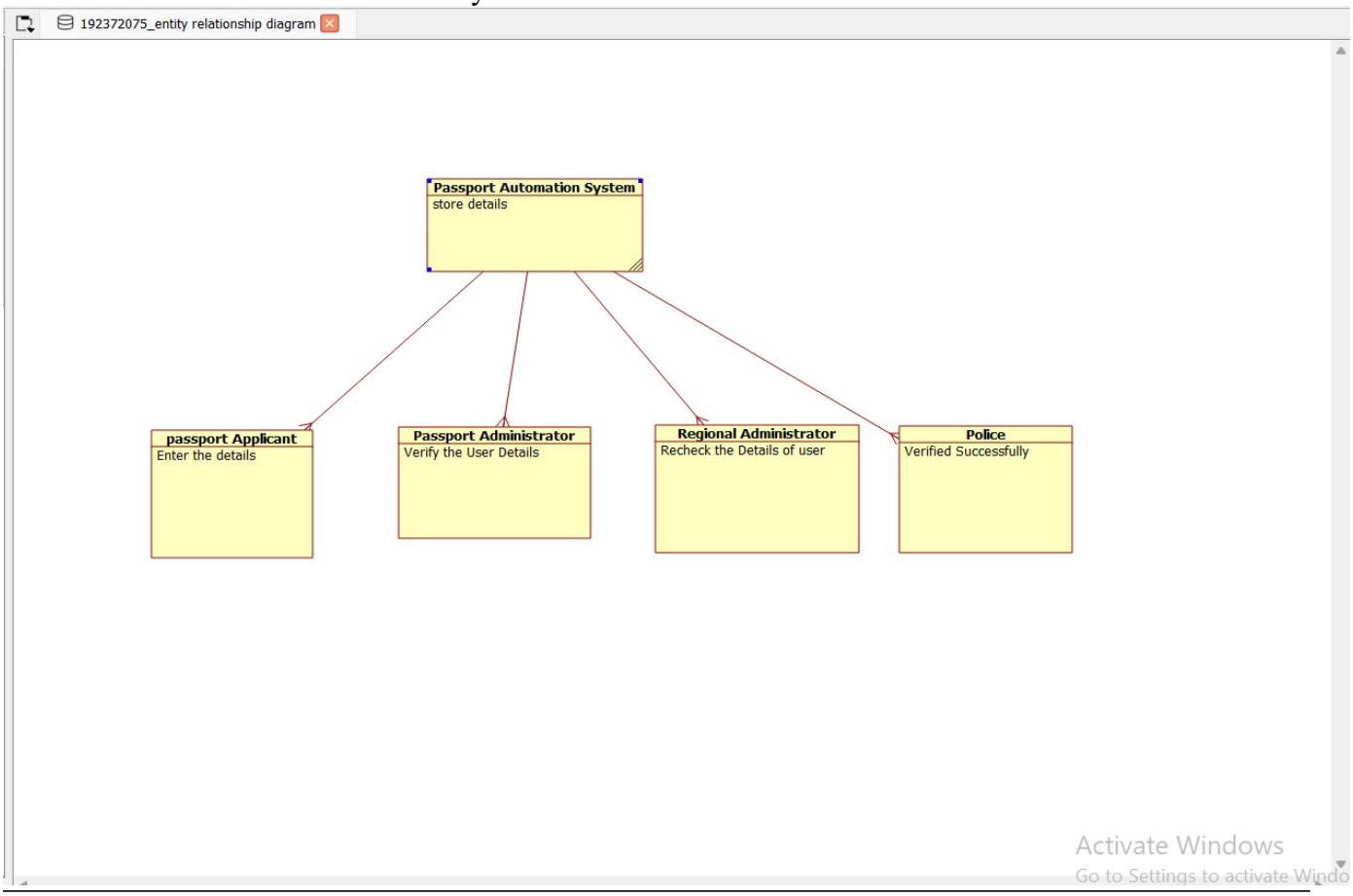


PACKAGE DIAGRAM:

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

- o User interface layer
- o Domain layer
- o Technical services layer



PROGRAM CODING:

APPLICANT:

```
Public class Applicant
{
    Public Integer firstname;
    Public Integer lastname;
    Public void passport()
    {
```

```
        }
}
```

PASSPORT APPLICATION SYSTEM:

```
Public class passport application system
{
        Public Integer details;
        Public Integer proof;
        Public class Applicant
        {
        Public Integer firstname;
        Public Integer lastname;
```



```
        Public void passport()
        {
        }
        Public void verification()
        {
        }
        Public void issue()
        {
        }
        Public void cancel()
        {
        }
}
```

OFFICER:

```
Public class officer
{
```

```
        Public Integer form;
        Public Integer responsible;
        Public void Database()
        {
        }
}
```

RESULT:

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