EX NO: 3 DATE:	EXAM REGISTRATION SYSTEM

AIM:

To draw the diagrams [use case, activity, sequence, collaboration, class, statechart, component, deployment, package] for the Exam registration 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

SEQUENCE DIAGRAM:

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

Object: student, educational officer, central education 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. A final state is represented as circle surrounding a small dot.

STATE CHART DIAGRAM:

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

REGISTER NO:

PROGRAM CODING:

CENETRAL EDUATIONAL SYSTEM:

```
Public class central educational system
{
        Public integer student details;
        Public void valid proof()
        {
        }
}
```

EDUCATIONAL OFFICER:

```
Public class educational officer
{
        Public integer id no;
        Public string name;
        Public void verification of proof()
        {
        }
        Public void issue hall ticket()
        {
        }
}
```

STUDENT:

```
Public class student
{
        Public integer student details;
```

Public void payment of fees()

{

Public void receive hall ticket()

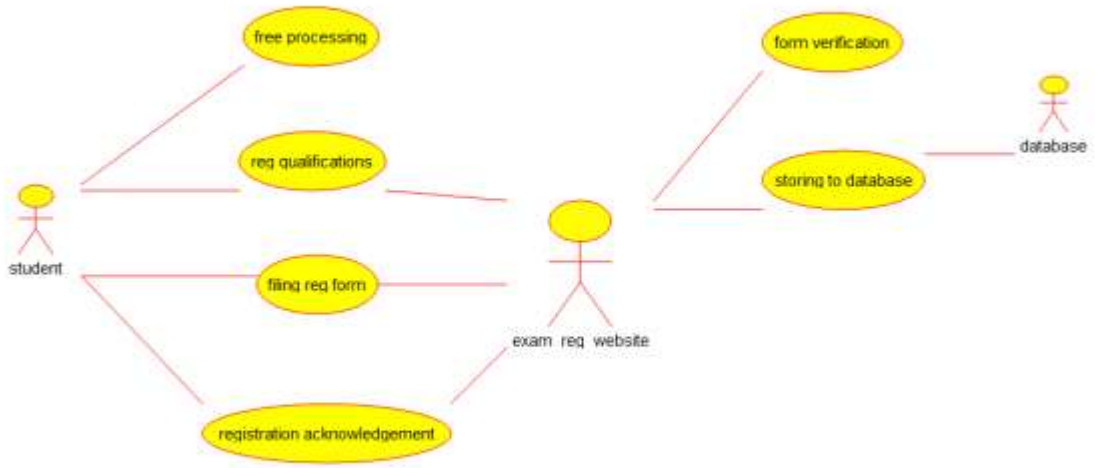
{

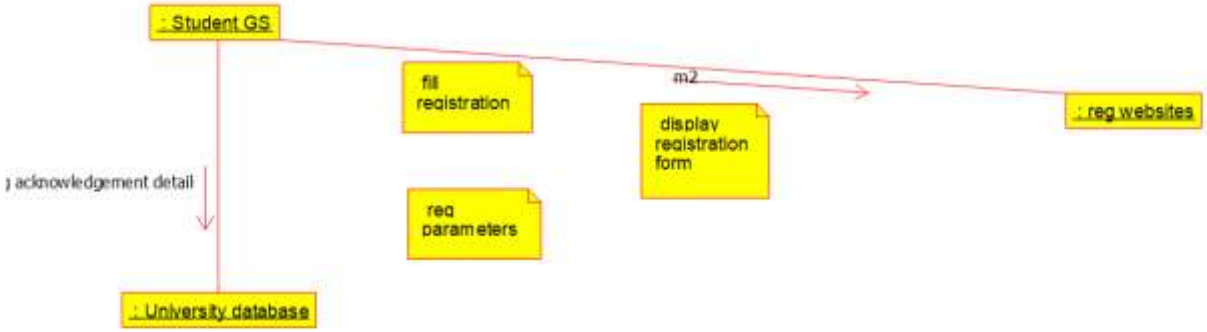
}

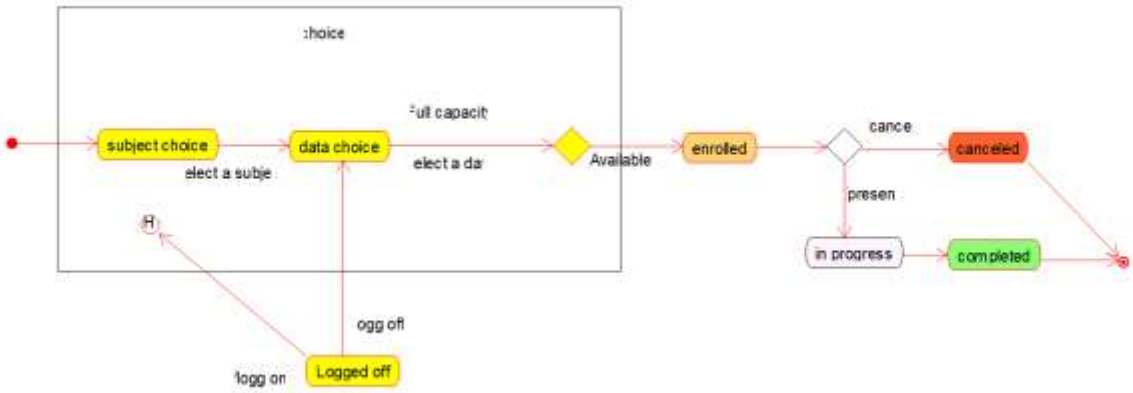
}

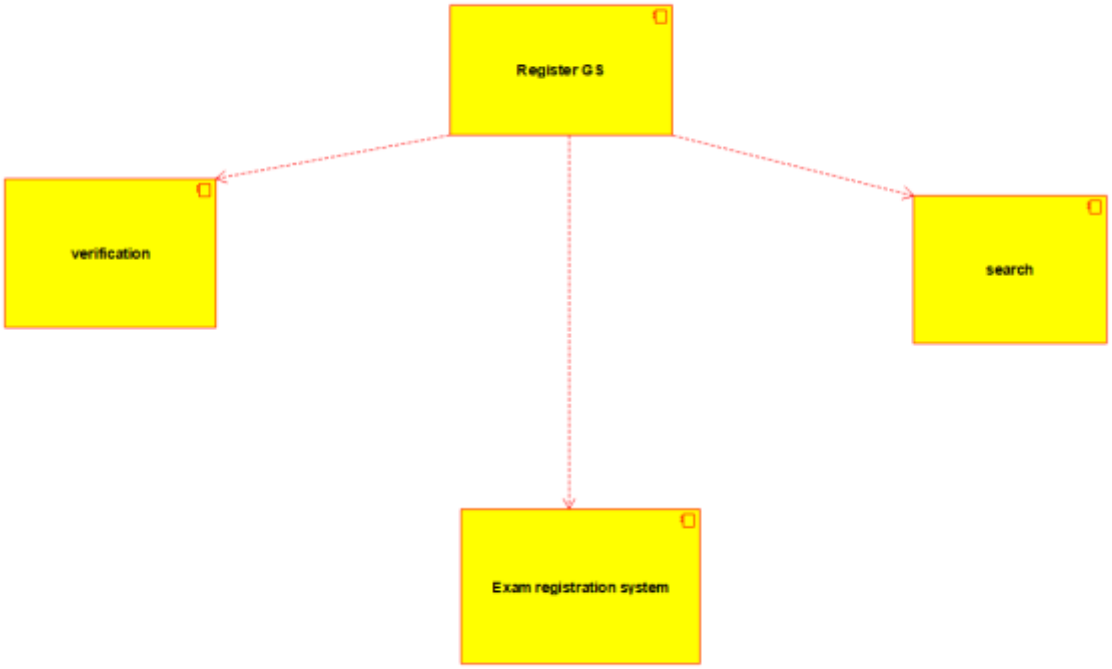
RESULT:

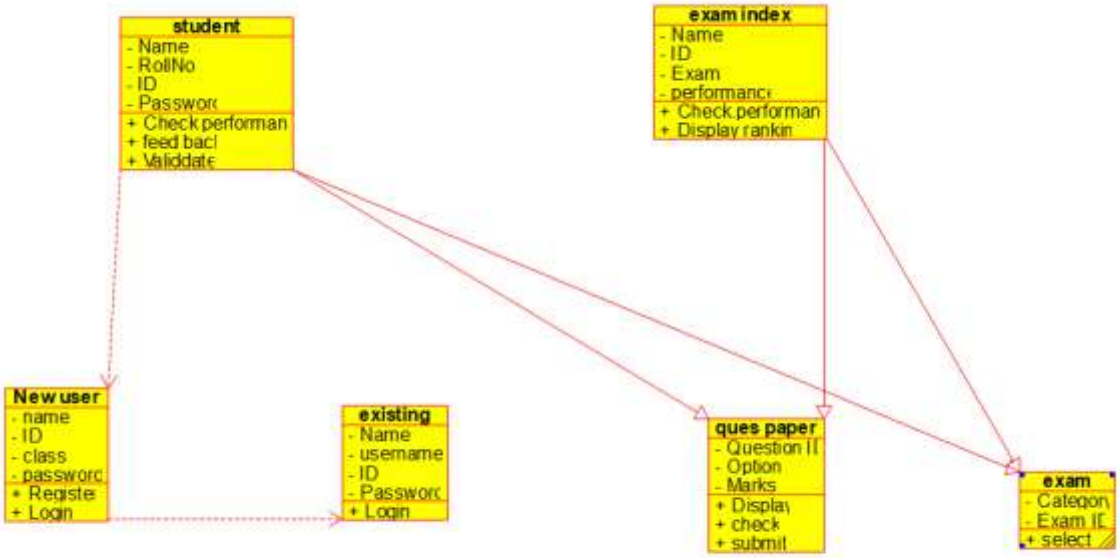
Thus the diagrams [use case, activity, sequence, collaboration, class, statechart, component, deployment, package] for the Exam registration system.




: Student GS
fill registration
m2
display registration form
: reg.websites
req parameters
: University database


:hoice
subject choice
data choice
Available
enrolled
canceled
in progress
completed
Logged off


Register GS
verification
search
Exam registration system


student
- Name
- RollNo
- ID
- Passwor
+ Check performan
+ feed bac
+ Validdate
exam index
- Name
- ID
- Exam
- performanc
+ Check performan
+ Display rankin
New user
- name
- ID
- class
- passwor
+ Registe
+ Login
existing
- Name
- username
- ID
- Passwor
+ Login
ques paper
- Question II
- Option
- Marks
+ Display
+ check
+ submit
exam
- Categor
- Exam IE
+ select