EX NO:

DATE

STUDENT INFORMATION SYSTEM

AIM

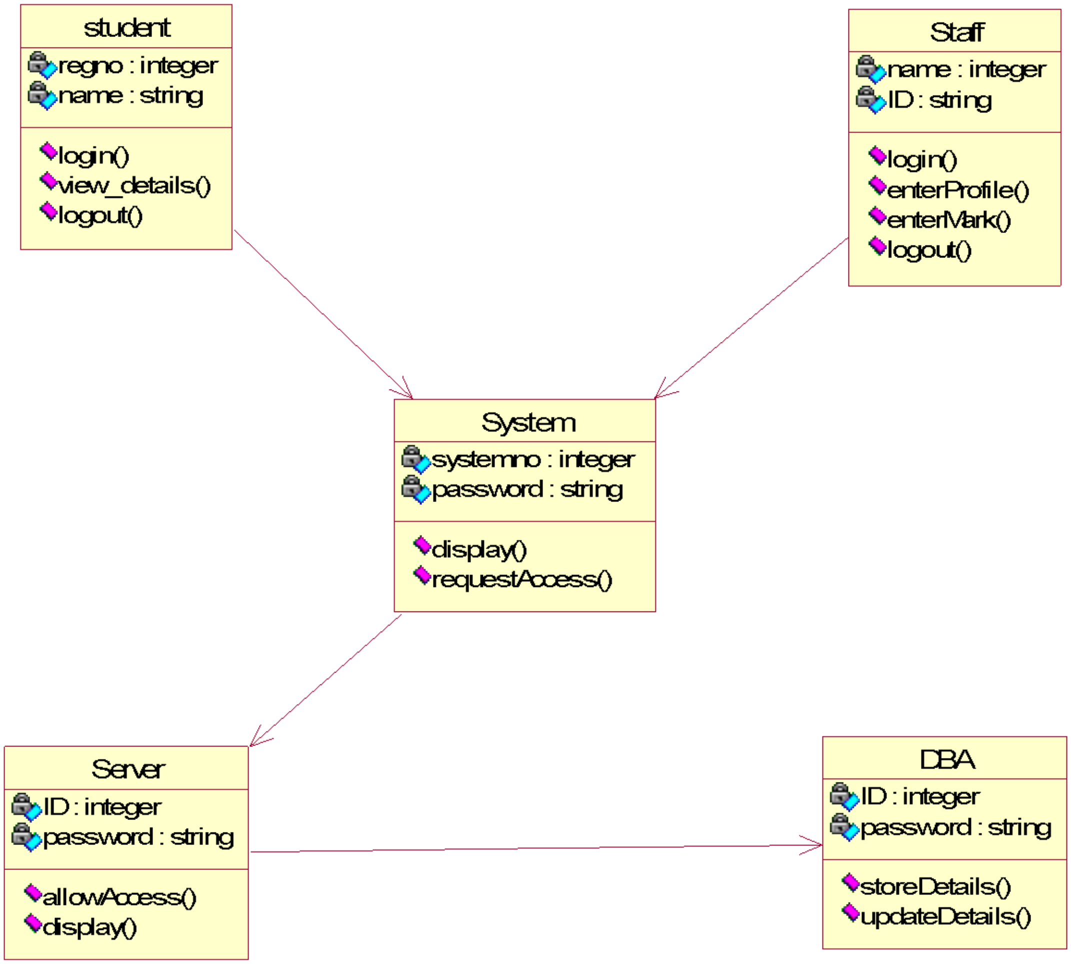
To develop a project Student information system using the Rational Rose Software and to implement the software in Java.

PROBLEM STATEMENT

The student must register by entering the name and password to login the form. The admin select the particular student to view the details about that student and maintaining the student details. This process of student information system is described sequentially through following steps. The student registers the system. The admin login to the student information system. He/she search for the list of students. Then select the particular student. Then view the details of that student. After displaying the student details then logout.

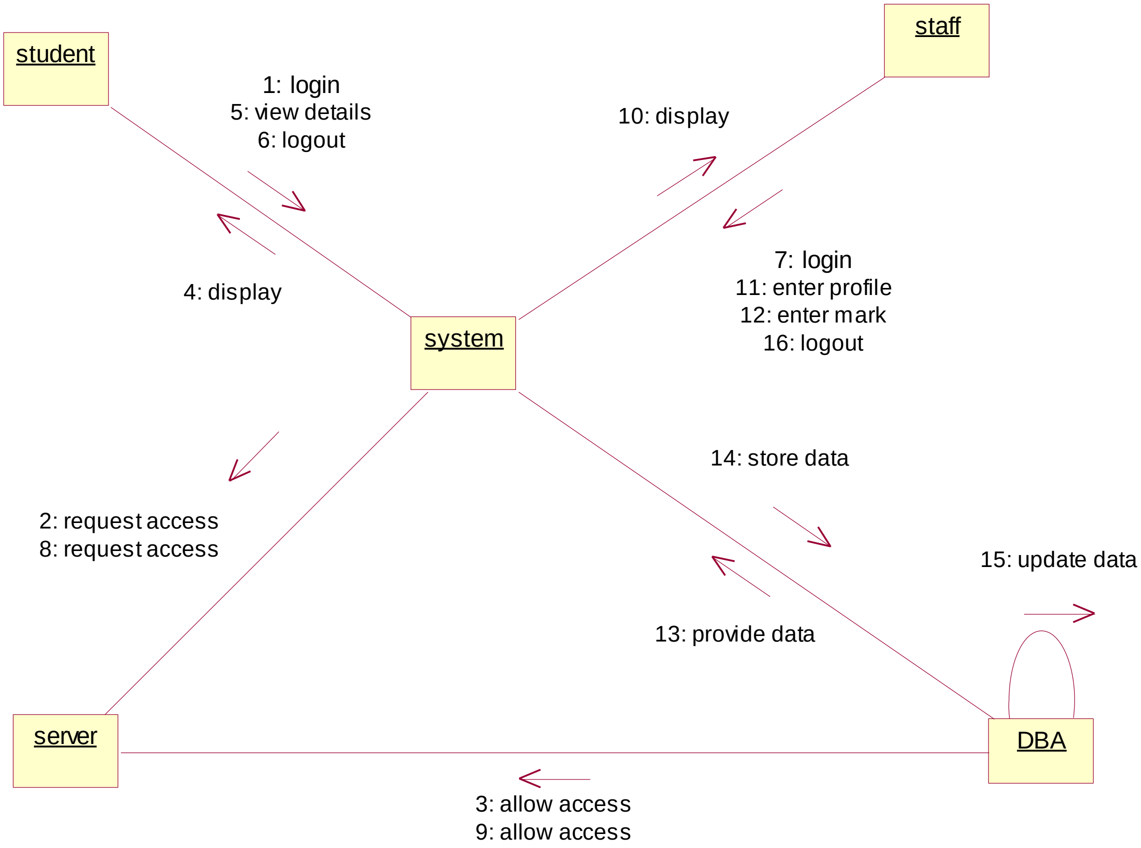
CLASS DIAGRAM

The class diagram is the graphical representation of all classes used in the system. The class diagram is drawn as rectangular box with three components or compartments like class name, attributes and operations. The student information system makes use of the following classes like student, staff, system, DBA and server.



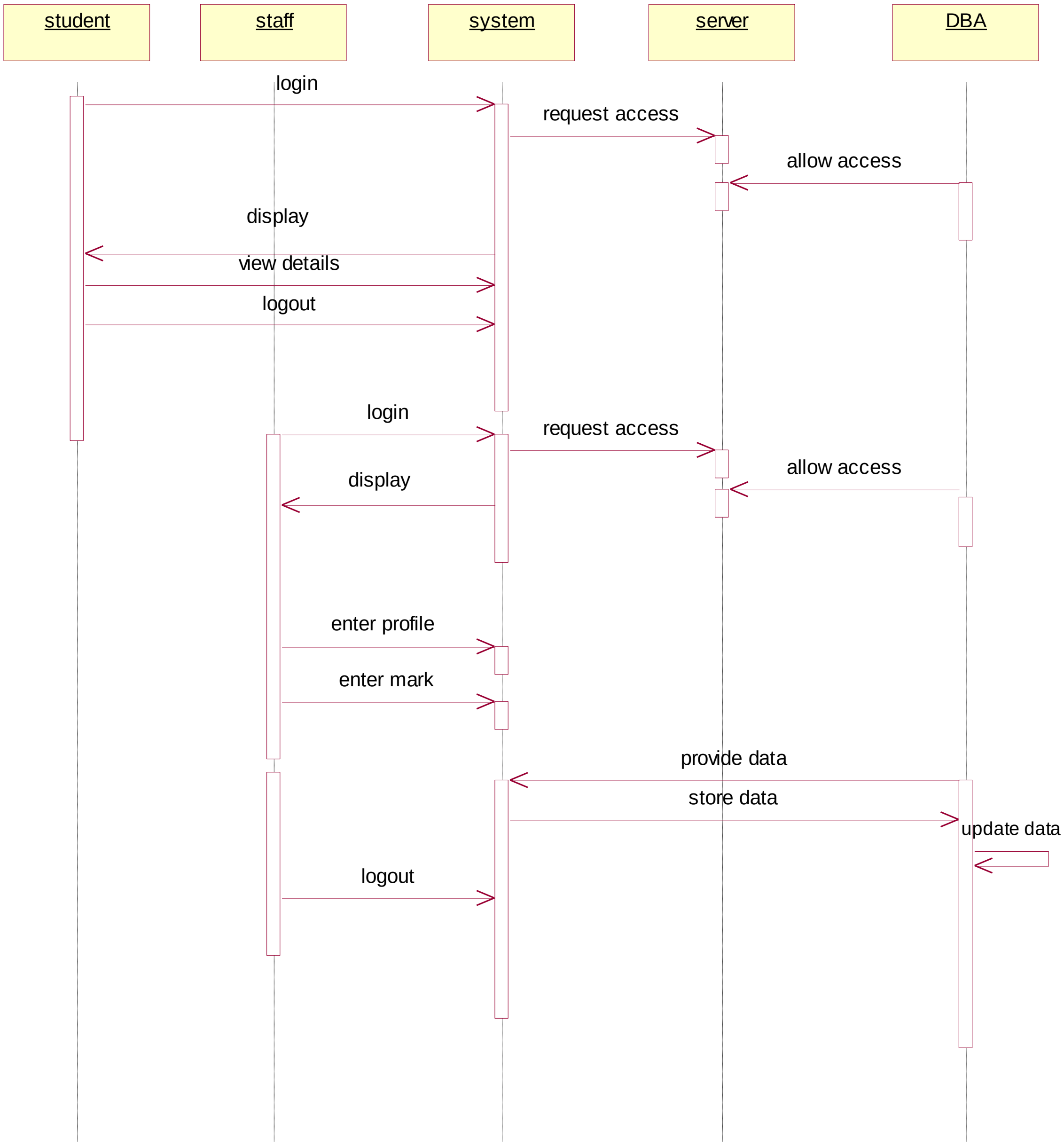
COLLABORATION DIAGRAM

A Collaboration diagram represents the collaboration in which is a set of objects related to achieve a desired outcome. In collaboration, the sequence is indicated by numbering the message several numbering schemes are available. Login, request access, allow access, display, view details, logout, login, request access, allow access, display, enter profile, enter mark, provide data, logout, store data, update data.



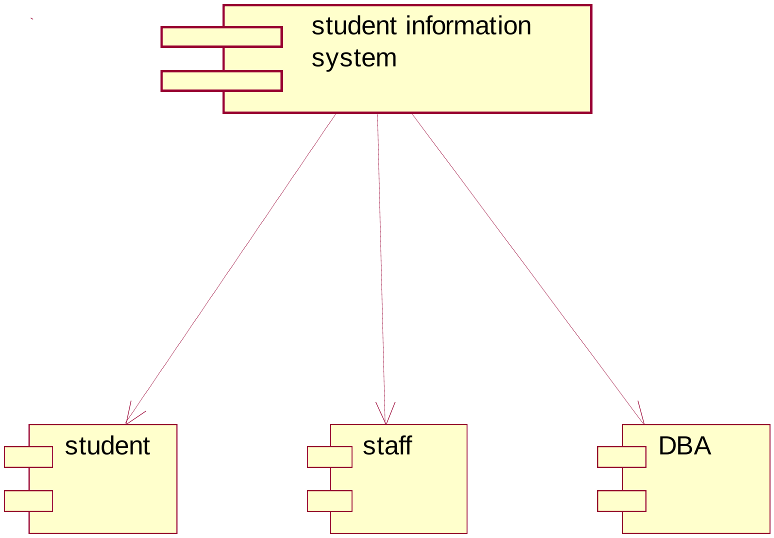
SEQUENCE DIAGRAM

A Sequence diagram represent the sequence and interaction of a given usecase or scenario. Sequence diagram capture most of the information about the system. Here the sequence starts between the student and the system. The second half of interaction takes place between staff and system then by police and followed by database. The student first login to the system and then view the details of the details. Staff login to the system enter mark and enter the details of the student. DBA store and update the details of the student.



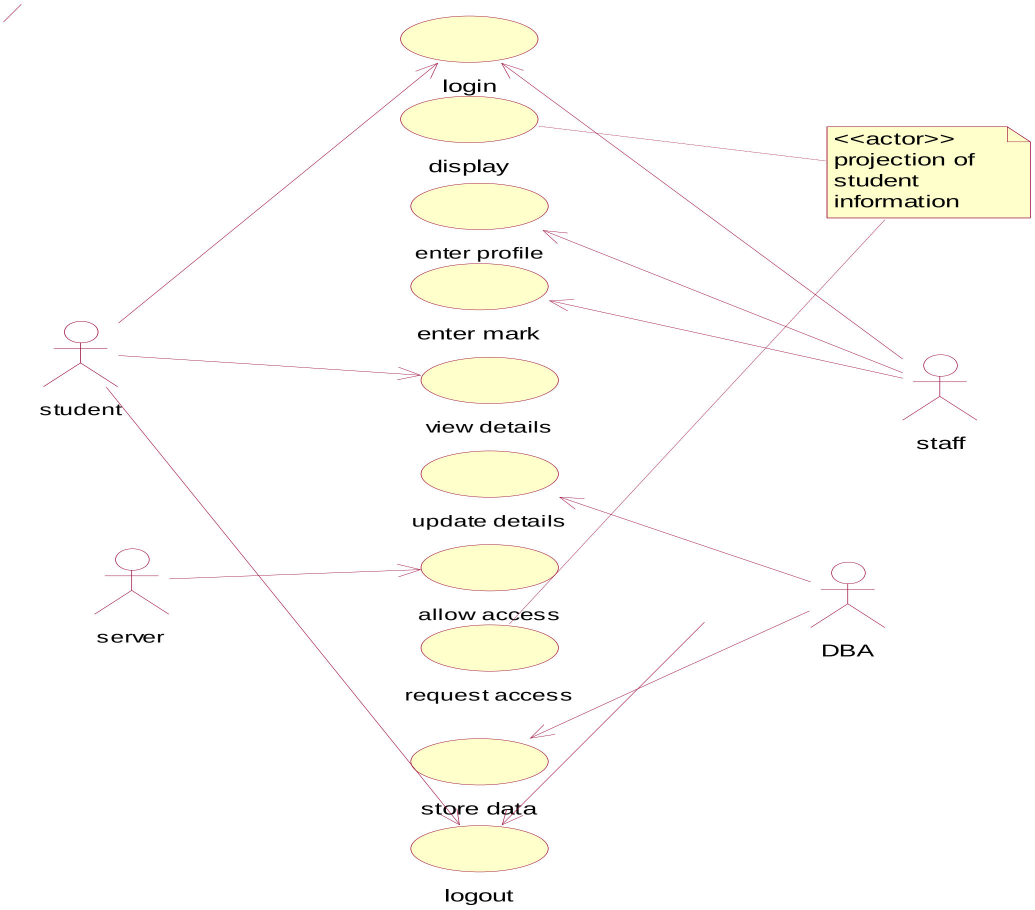
COMPONENT DIAGRAM

Component diagram carries the major living actors of the system. The component diagram main purpose is to show the structural relationship between components of the system.The main component of the system is student information system and the other components of the system are student, staff and DBA.


student information system
student
staff
DBA

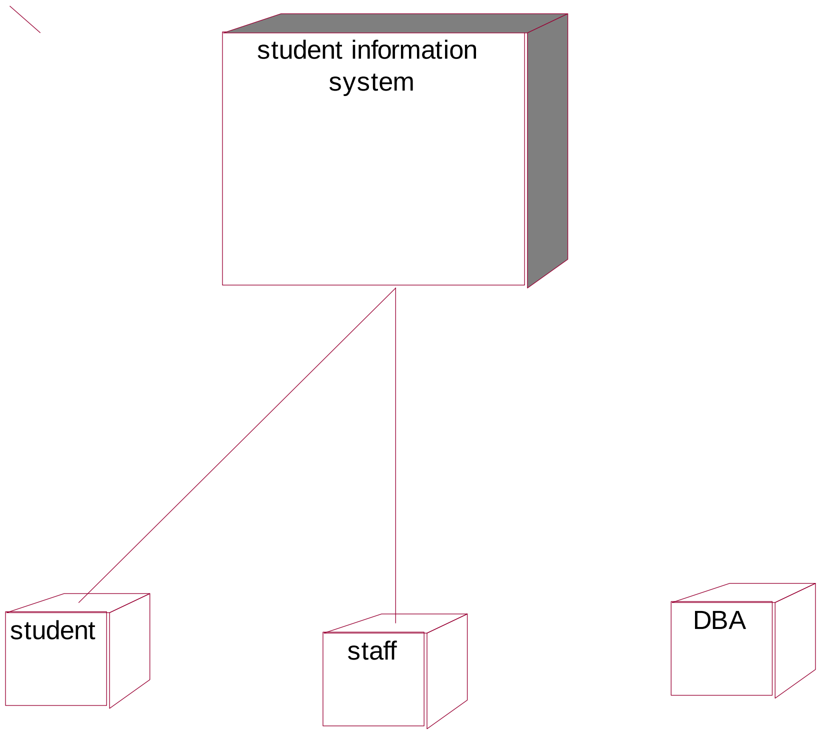
USE CASE DIAGRAM

Use case diagram is a graph of actors, a set of use cases, association between the actors and the use cases and generalization among the cases. Use case diagram is a list of actions or events. Use case diagram was drawn to represent the static design view of the system. Steps typically defined the interactions between a role and a system to achieve a goal. The use case diagram consists of various functionality performed by the actors like student, staff, system, DBA and server. The use case diagram consists of various functionality like login, display, enter profile, enter mark, view details, update details, allow access, request access, store details, logout.



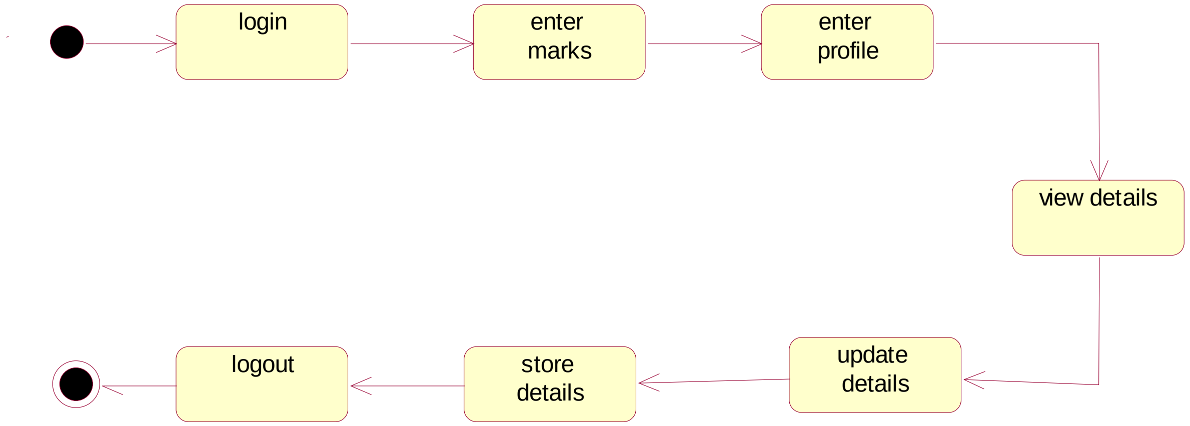
DEPLOYMENT DIAGRAM

Deployment diagram shows the configuration of runtime processing elements and the software components processes and objects that live in them. Component diagram are used in conjunction with deployment diagram to show how physical modules code are distributed on various hardware platform. The processor node in the system is student information system and the execution environment nodes or device nodes are student, staff and DBA.



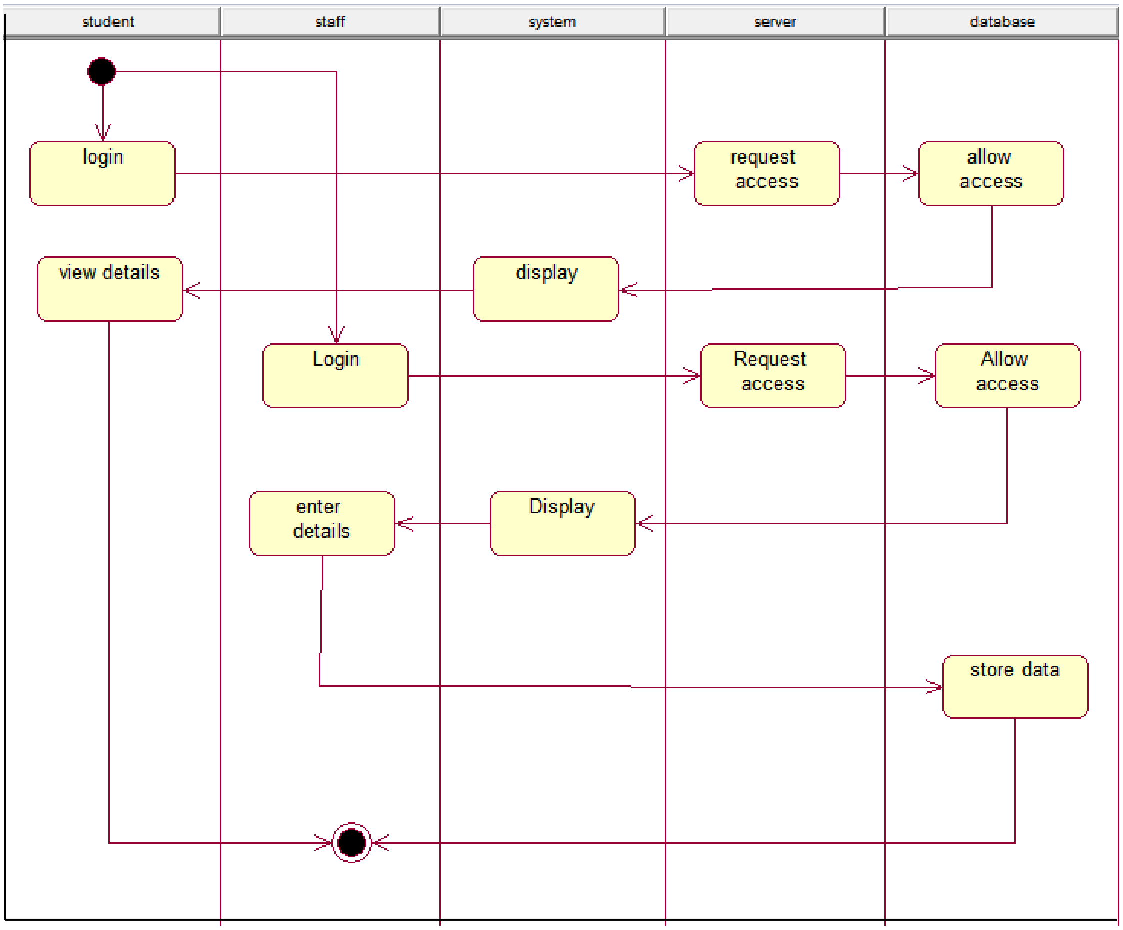
STATE CHART DIAGRAM

A State chart diagram is also called as state machine diagram. The state chart contains the states in the rectangular boxes and the states are indicated by the dot enclosed. The state chart diagram describes the behavior of the system. The state chart involves six stages such as login, enter mark, enter profile, view details, provide details, update details, store details and logout.



ACTIVITY DIAGRAM

Activity diagram are graphical representation of stepwise activities and actions with support for choice, interaction and concurrency. Here in the activity diagram the student login to the system and view the details of the student. The staff login to the system for entering the student details and update the details in the database. The final interaction is the DBA store the details of the student.



OUTPUT

```
//Source file: C:/Users/IT/Desktop/kj/DBA.java

public class DBA

{

  private integer ID;

  private string password;

  public DBA()

  {

  }

    /**

  @roseuid 50EE7E61008B

  */

  public void storeDetails()

  {

  }

  /**

  @roseuid 50EE7E6502C6

  */

  public void updateDetails()

  {

  }

}


//Source file: C:/Users/IT/Desktop/kj/Server.java

public class Server
```

```
{
  private integer ID;
  private string password;
  public DBA theDBA;
   public Server()
  {
  }
  /**
  @roseuid 50EE7E1B01DC
  */
  public void allowAccess()
  {
  }
  /**
  @roseuid 50EE7E2301CF
  */
  public void display()
  {
  }
}

//Source file: C:/Users/IT/Desktop/kj/Staff.java
public class Staff
{
  private integer name;
  private string ID;

  public System theSystem;

  public Staff()

  {

  }

  /**

  @roseuid 50EE7CE70203

  */

  public void login()

  {

  }

  /**

  @roseuid 50EE7CE70204

  */

  public void enterProfile()

  {

  }

   /**

  @roseuid 50EE7CE70205

  */

  public void enterMark()

  {

  }

  /**

  @roseuid 50EE7CE70206
```

```
   */
   public void logout()
   {
   }
}

//Source file: C:/Users/IT/Desktop/kj/System.java
public class System
{
   private integer systemno;
   private string password;
   public Server theServer;
   public System()
   {
   }
   /**
   @roseuid 50EE7CE7020A
   */
   public void display()
   {
   }
   /**
   @roseuid 50EE7CE7020B
   */
   public void requestAccess()
```

```
  {
  }
}
```

RESULT

Thus the various UML diagrams for student information system was drawn and code was generated successfully.