MULTIPLE IMAGE SHARING SCHEME USING VISUAL CRYPTOGRAPHY

Software Requirements Specification submitted in partial fulfilment of Academic requirements for the award of the Degree of

BACHELOR OF ENGINEERING
IN
INFORMATION TECHNOLOGY

By

N. Vinay Kumar (2451-15-737-012)
Y. Shiva Sankeerth (2451-11-737-013)
G. Sai Kiran (2451-15-737-021)

Under the guidance of
DR. Ch. Samson
Head of Department, Dept. of I.T
MVSR Engineering College
Nadergul, Hyderabad.

April,2018

DEPARTMENT OF INFORMATION TECHNOLOGY
MVSR ENGINEERING COLLEGE
(Affiliated to Osmania University, Hyderabad. Recognized by AICTE)
Nadergul, Saroornagar Mandal, Hyderabad-501510

2017-2018

INTRODUCTION

Problem Definition

- Conventional encryption-systems can encrypt and transmit only single image. We use multiple image sharing scheme for secure transmission of multiple images.
- The proposed scheme is applied for visual information data to provide secure transmission in such a way that only the receiver gets appropriate
- The system secretly generates 'N' number of shares on multiple images by applying simple bitwise XOR operation in such a way that single share doesn't reveal any information about secret images.
- The existing system in only used for grey level images whereas the proposed system can be used on color-images.

Objectives

- To securely transfer visual-data more efficiently.
- To develop a hack-proof encryption algorithm.
- To develop a decryption algorithm without having loss in data.
- To reduce data redundancy and inconsistency.
- To make the software interface as user-friendly as possible.
- To reduce expenses of transmitting multiple images in a systematic manner.

Definitions

- Cryptography - It is a method of storing and transmitting data in a form so that only those for whom it is intended can read and process it.
- Visual Cryptography – Visual cryptography is a cryptographic technique which allows visual information to be encrypted in such a way that decryption becomes the job of the person to decrypt via sight reading.
- Management – The body managing the overall working of the system.
- Database – The total data stored through which the various actors interact and get required information. It is continually updated for each procedure completed.
- UML - **UML** (Unified Modeling Language) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.
- DBA – Person in-charge of managing the database

GENERAL DESCRIPTION

Product Perspective

This system is dependent on the Encryption and Decryption algorithms which make the transmission of images consistent and loss-less. It interfaces the users with a special user screen that has unique features than the normal user screens.

Product Function

The various functions performed by the software are:

- Login – Qualifies users according to their username and password which provides security and improves the effectiveness of the software.

- Encryption Algorithm – Allows images to be encrypted using a simple XOR operation with the help of a key-image.
- Decryption Algorithm – Allows images to be decrypted at diverse levels using a key-image.
- Selection Prompt – Helps the user to input various images depending on their purpose.
- Results – Which displays the encryption and decryption details to the user.
- Management of database – Assists the DBA in maintaining the database.

User Characteristics

The users are required of possessing the following qualities: highly motivated and conscientious to security, computer literate not necessarily professionals.

SPECIFIC REQUIREMENTS

Functional Requirements

1. *Login*

   Accessing the database to read data. Any user who logs in does so under his user name and password.

2. *Encryption*

   Steps involved
   - Implementing the multiple-image sharing algorithm.
   - Updating the changes according to the selection prompt.

   Inputs
   - Input-image, Key-image and Secret-images.

   Processing
   - The user provider logins to send the images available in their respective systems.
   - The system checks for the validity of the user name and password and allows access to the required view.
   - The database is updated according to the images provided by the user.

   Outputs
   - The changes are saved and displayed on the results window.

   Constraints
   - The user must have a valid user name and password

3. *Decryption*

   Steps involved
   - Checking the validity of the receiver and verifying the password.
   - Updating the changes according to the details provided.

   Inputs
   - User name and password of the Receiver.

Processing

- The receiver logins to receive the images at distinct levels.
- The system checks for the validity of the user name and password and allows access to the required view.
- The results window is updated according to the changes made by the receiver.

Outputs

- The changes are saved and displayed on the results window.

Constraints

- The user must have a valid user name and password

5. *Results*

Steps involved

- Accepting the user's login username and password.
- Display the results for a given username and password.

Inputs

- user's login username and password.

Processing

- The user accesses the application for the results.
- The system checks for the availability of image material and if available displays the results.
- The receiver can only check the image material but can't update.

Outputs

- For valid input, display the result.

6. *Management of Database*

Steps involved

- Adding, updating or deleting user information in the database.
- Adding, updating or deleting visual information.
- Allocation and de-allocation of roles and privileges.
- Modifying the information, constraints governing the application.

Inputs

- User name and password to authorize the DBA.
- Selection of task.
- Unique identification key incase the DBA selects to update database so that concerned data item can be accessed.

Processing

- The DBA uses this function to manage the changes in database, which occur due to other user cases, and to perform other tasks specified for a DBA.
- The DBA logs into the system and then performs the required task.

Outputs

- Confirmation to the changes made.
- In case any failure, DBA is informed and the changes are cancelled.

Constraints

- Atomicity should be ensured, that a task should perform totally or should be completely aborted.
- The changes made in the database should not conflict with the constraints specified.

External interface requirements

- External interfaces must be maintained by the central database as interpreted by the operator.
- Soft copy reports shall be the only output maintained.
- These outputs shall be controlled by authorized management personnel.

Performance Requirements

All user requests shall be responded to within 3 to 5 seconds.
Response shall be measured from the last entry of image to the appearance of an on-screen reply.

Design Constraints

Design and development of the system shall confirm to Security standards.
The system is a basic MATLAB application which runs with the help of a reliable database management system.

Attributes

1. Security

- Security of all files shall be a major design requirement. Only authorized users shall be permitted to access the database.

- Passwords shall be assigned only by the authorized management personnel.

2. Maintainability

- Software product documentation shall include design specification, a commented listing (a comment for every three executable lines of code), a set of executable test cases and appropriate results.
- A user manual shall be provided in a reproducible form.

3. Reliability

- The company largely relies upon the database and so the system needs to be reliable.

OTHER REQUIREMENTS

Database

- An efficient database management system shall be used as a back end, which facilitates reliable and affective transactions.

Operations

- A backup of the database is created after each interval of time, which can be used for recovery if needed.

TESTING:

- All modules of the systems need to be tested separately and the system to assure that the system works properly as per the requirement specification.
- Alpha and beta testing can be used for this purpose.

Following are the some of the testing methods applied to this effective project:

Specification Testing:
We can set what the program should do and how it should perform under various conditions. This testing is a comparative study of evolution of system performance and system requirements.

Module Level Testing:
In this the error will be found at each individual module, it encourages the programmer to find and rectify the errors without affecting the other modules.

Unit Testing:
Unit testing focuses on verifying the effort on the smallest unit of software module. The local data structure is examined to ensure that the date stored temporarily maintains its integrity during all steps in the algorithm's execution. Boundary conditions are tested to ensure that the module operates properly at boundaries established to limit or restrict processing.

Integration Testing:
Data can be tested across an interface. One module can have an inadvertent, adverse effect on the other. Integration testing is a systematic technique for constructing a program structure while conducting tests to uncover errors associated with interring.

Validation Testing:
It begins after the integration testing is successfully assembled. Validation succeeds when the software functions in a manner that can be reasonably accepted by the client. In this much of the validation is done during the data entry operation where there is a maximum possibility of entering wrong data. Other validation will be performed in all process where correct details and data should be entered to get the required results.

Recovery Testing Recovery:
Testing is a system that forces the software to fail in variety of ways and verifies that the recovery is properly performed. If recovery is automatic, reinitialization, and data recovery are each evaluated for correctness.

Security Testing:
Security testing attempts to verify that protection mechanism built into system will in fact protect it from improper penetration. The tester may attempt to acquire password through external clerical means, may attack the system with custom software design to break down any defenses to others, and may purposely cause errors.

Performance Testing:

Performance Testing is used to test runtime performance of software within the context of an integrated system. Performance test are often coupled with stress testing and require both software instrumentation.

System Testing:

Testing the entire system and checking for its correctness is system testing. The system is listed for dispensaries between the system and its original objectives. This project was effective and efficient.

Output Testing:

After performing the validation testing, the next step is output testing of the proposed system since no system would be termed as useful until it does produce the required output in the specified format. Output format is considered in two ways, the screen format and the printer format.

User Acceptance Testing:

User Acceptance Testing is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system users at the time of developing and making changes whenever required.

The flowing are the testing points:

- Input Screen design
- Output Screen design
- Menu-driven System

Implementation is the process of bringing the developed system into operational use and turning it over to the user. The implementation of computer based system requires that test be prepared and that the system and its elements be tested in planned and structured manner.

All the above schemes can be used only to share the black and white secret images, but it is demand of time that schemes should also support color images. To meet this demand researches have been made to share the color images.

UML DIAGRAMS

1.Use case Diagrams

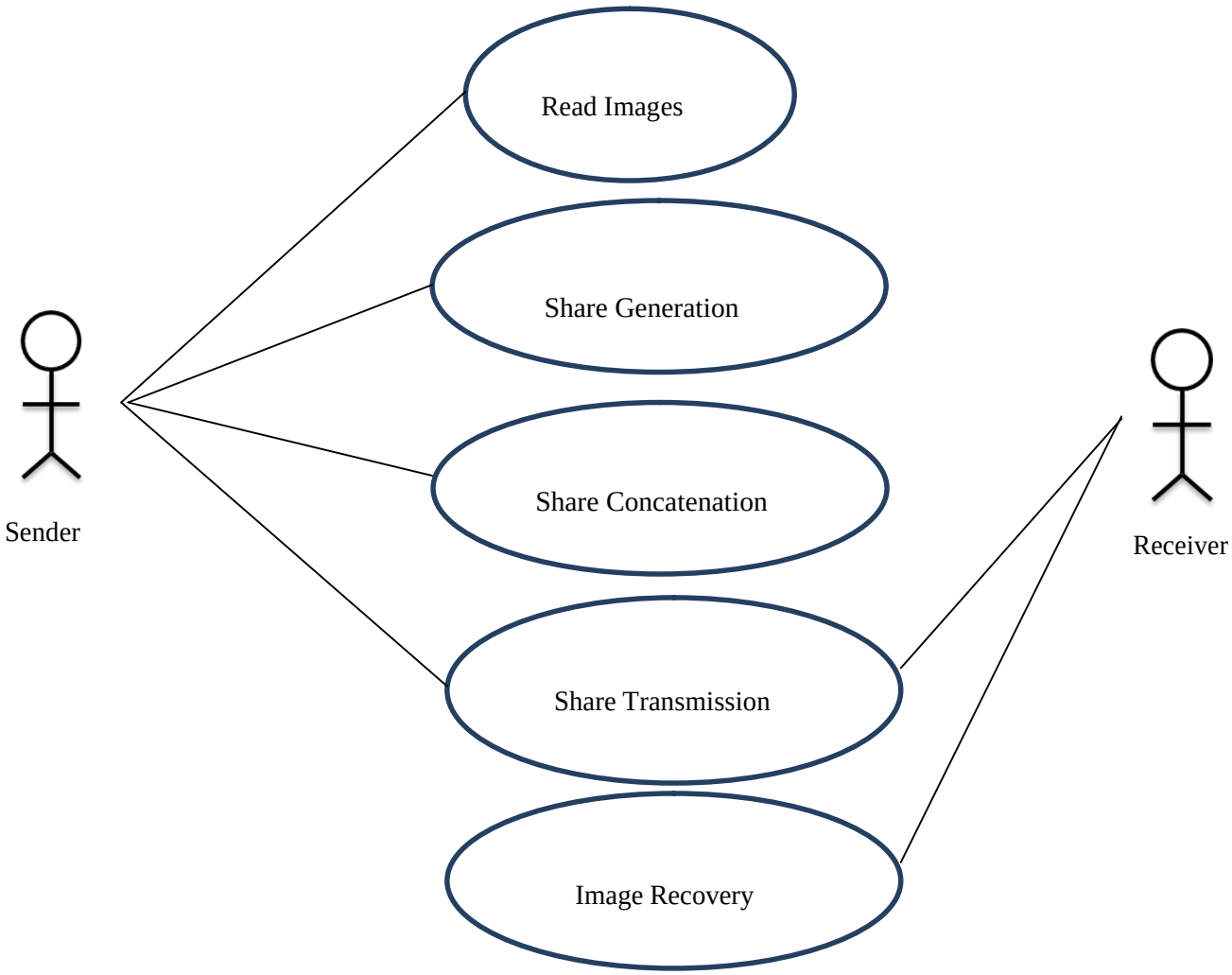

Fig 1: Use case diagram for System

2.Class Diagram

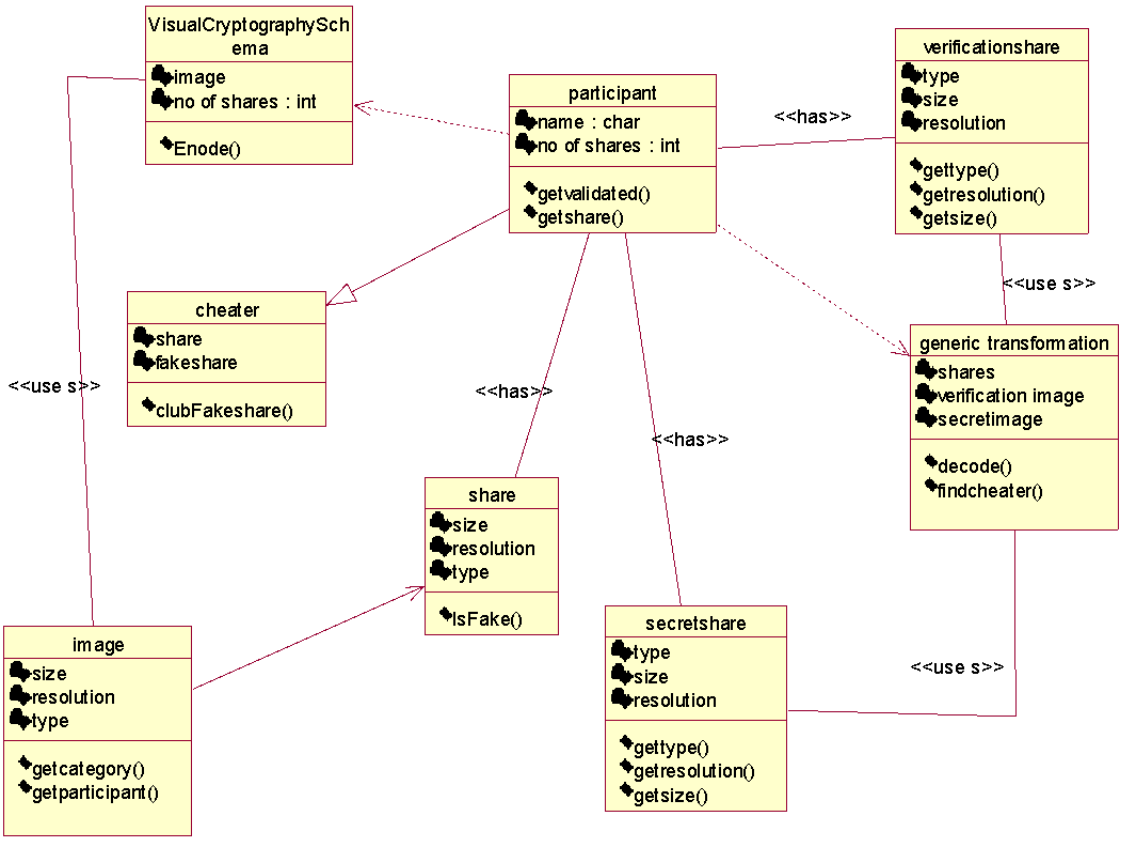

Fig 2: Class Diagram for System

3.Sequence Diagram

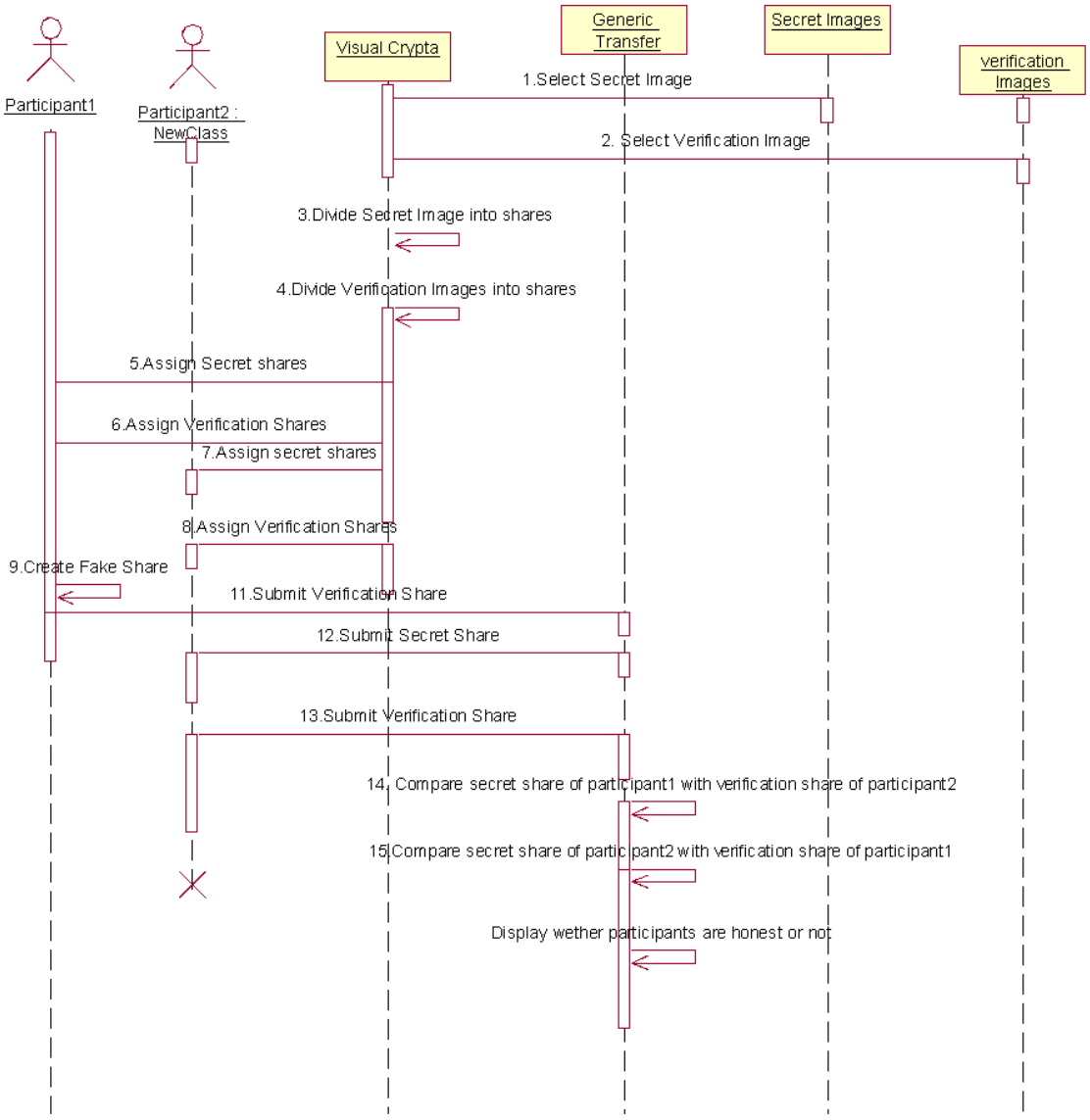

Fig: Sequence diagram for the system

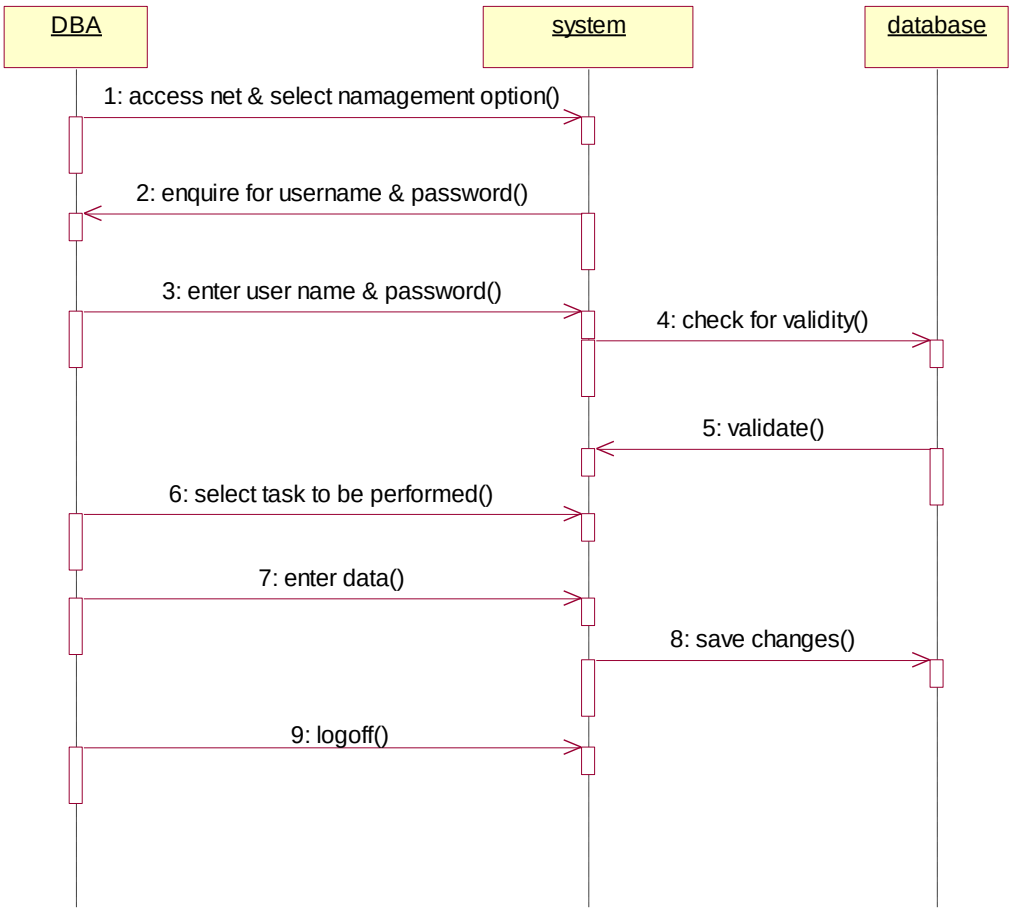

Fig: Sequence diagram for the management of database

4.Collaboration Diagrams

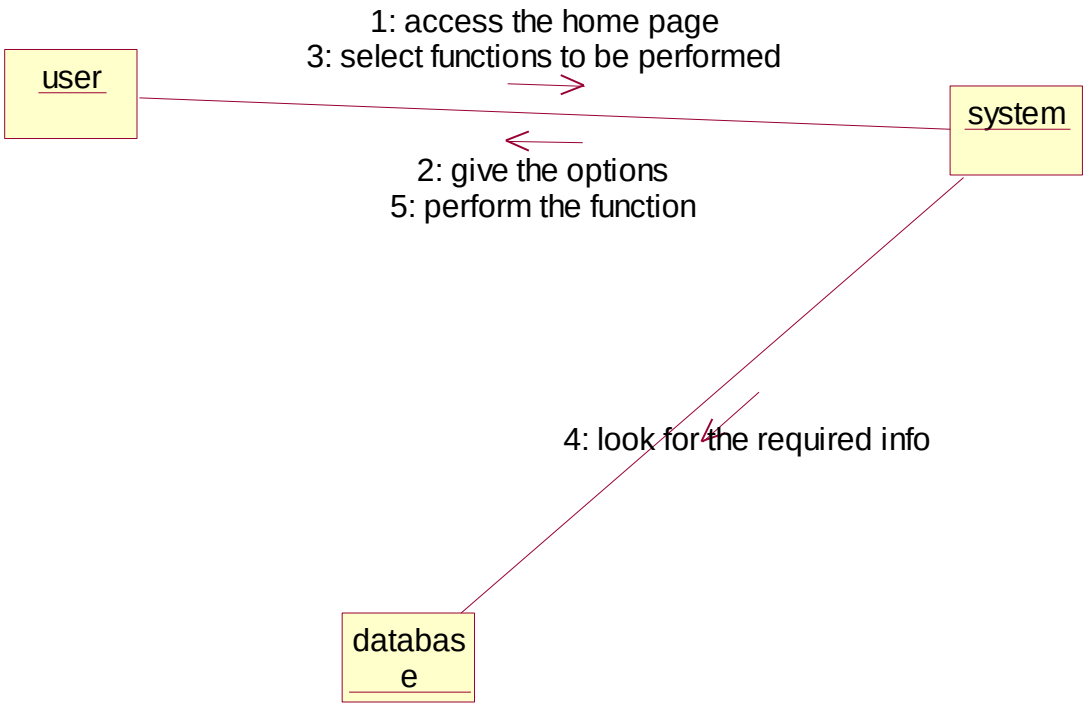

Fig: Collaboration diagram for the system

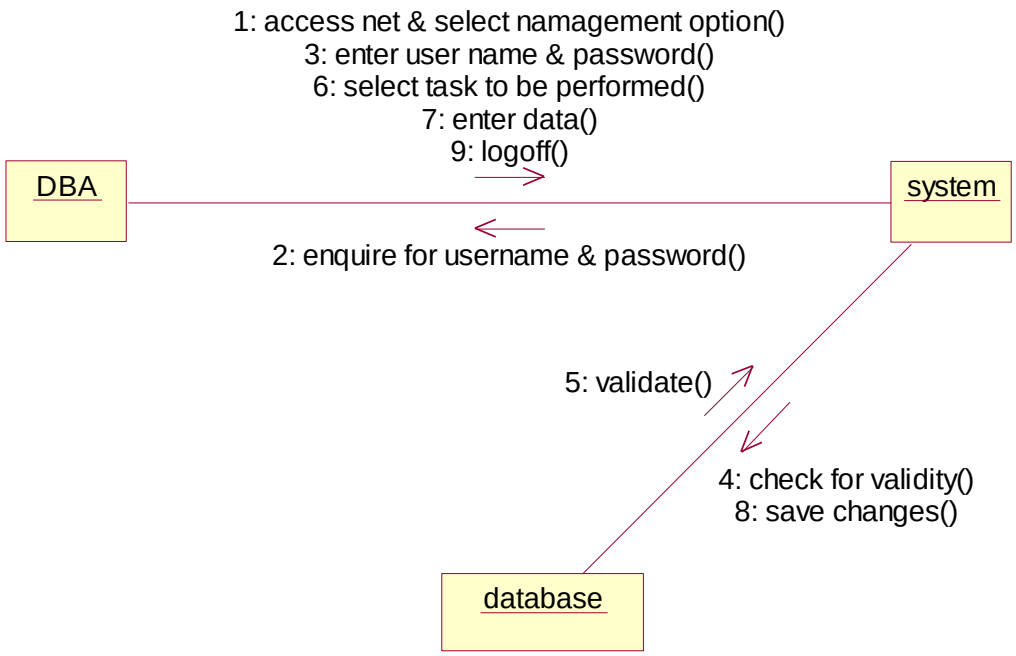

Fig: Collaboration diagram for Management of database

5.Activity diagrams:

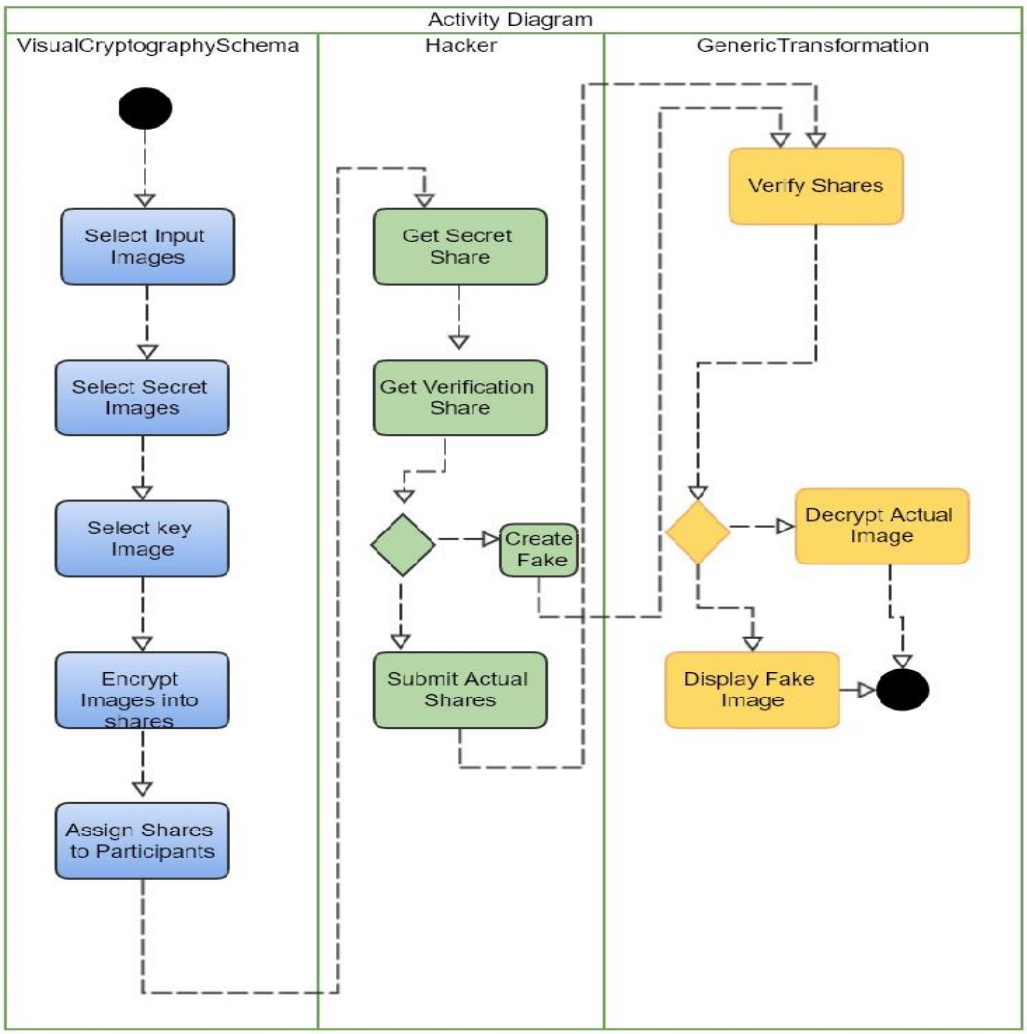

Fig: Activity diagram for system

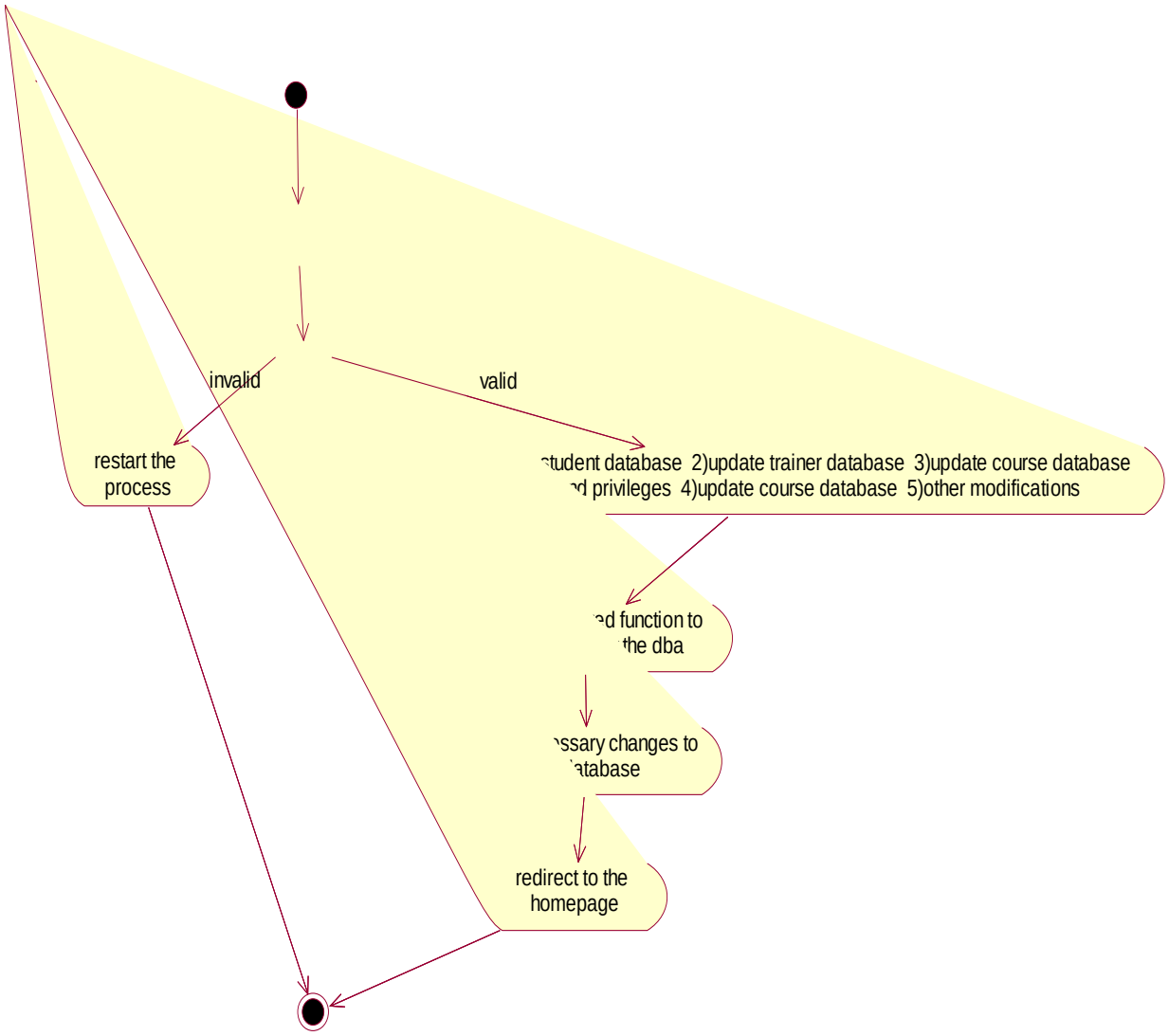

Fig: Activity diagram for DBA functionality

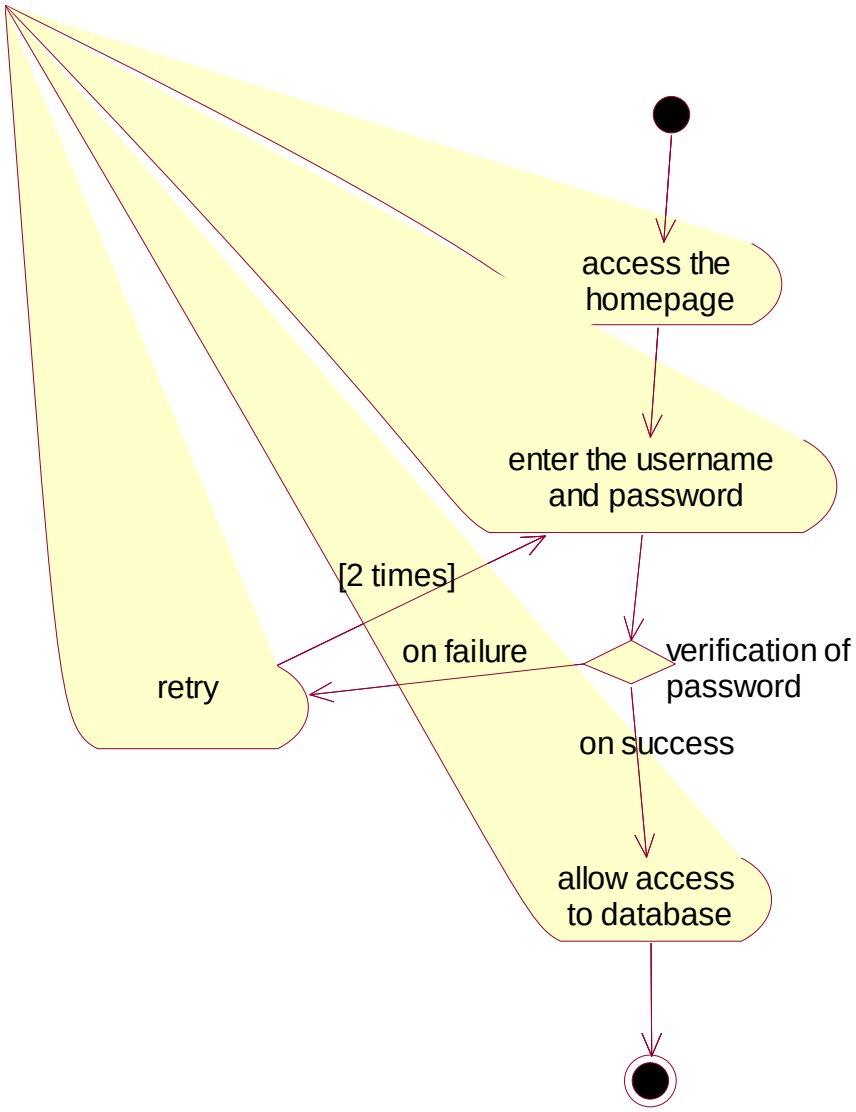

Fig: Activity diagram for login

6.State chart diagram:

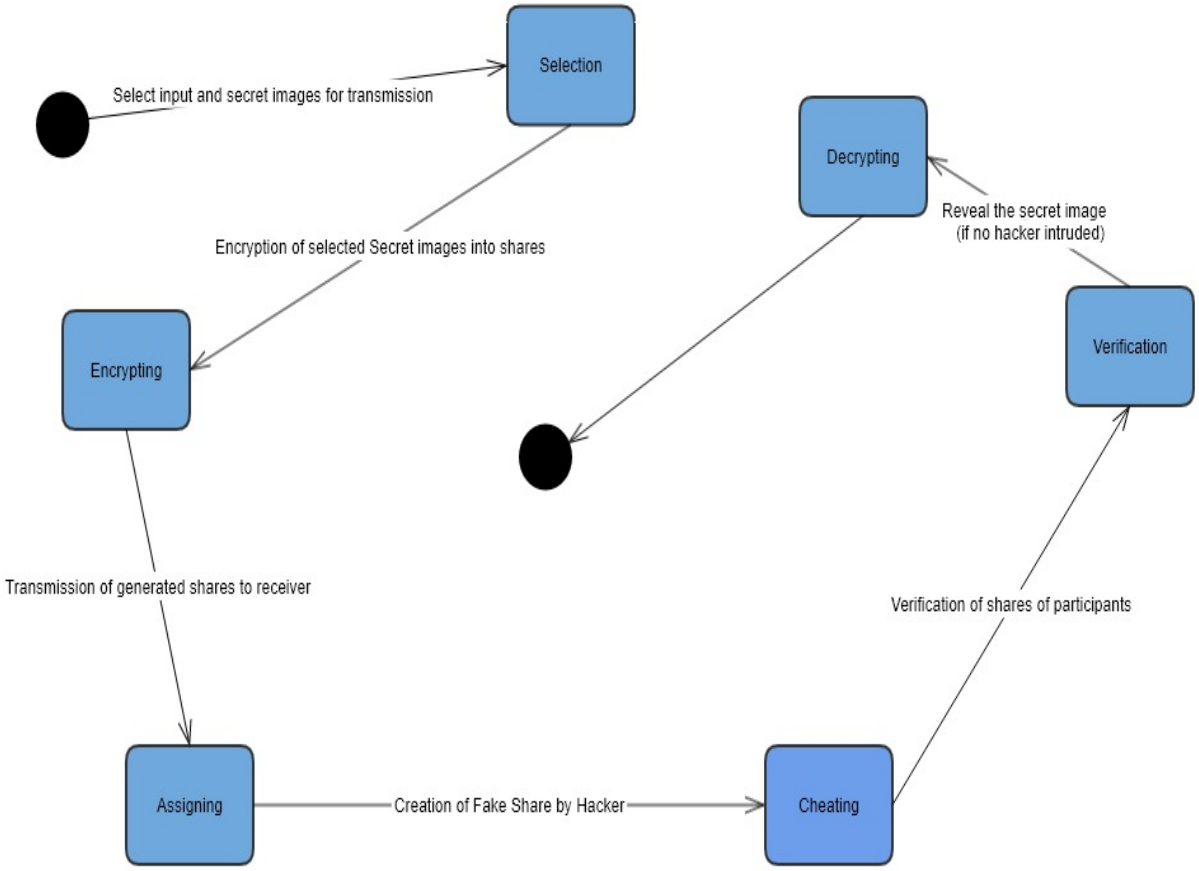

Fig: State chart diagram for the system

7.Component diagram:

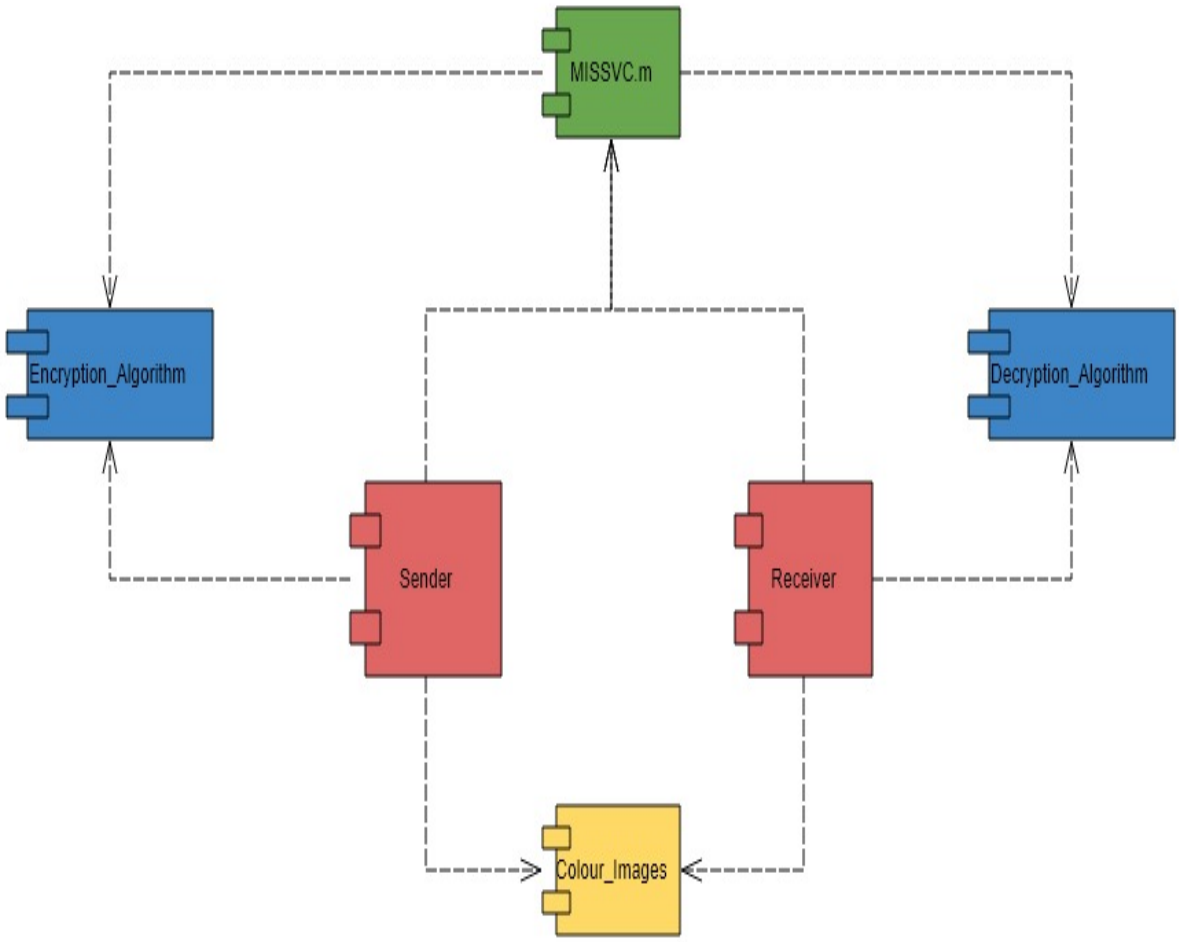

Fig: component diagram for the system

8.Deployment Diagram:

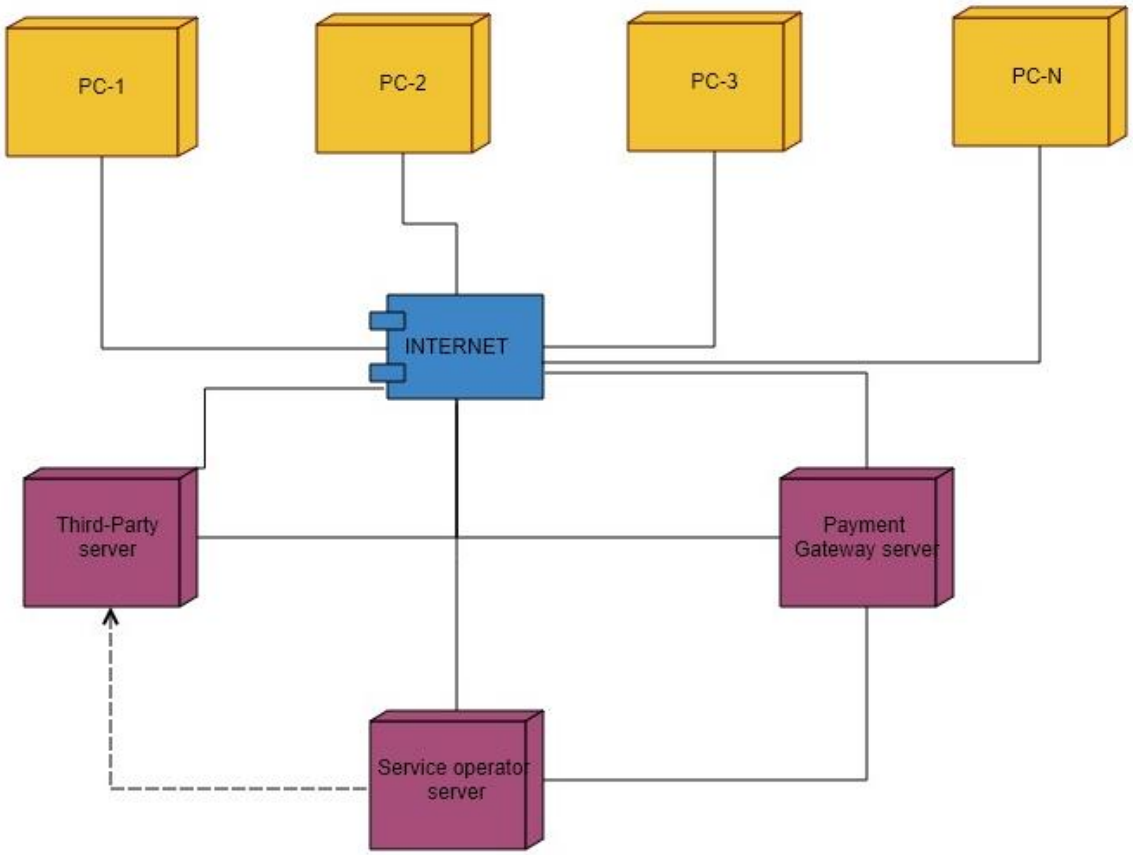

Fig: deployment diagram for system