

**ST. JOHN COLLEGE OF ENGINEERING AND TECHNOLOGY**

Vevoor , Manor Road, Palghar (E), District Thane – 401404

2014-2015

JNS Encryption Algorithm

**B. E. COMPUTER ENGINEERING (A)**

By

**Neil Pereira 44**

**Sneha Varghese 57**

**Joel Ulahanna 62**

Under the guidance of

**Mr. Sankusu Sharma**

Department of Computer Engineering

**ST. JOHN COLLEGE OF ENGINEERING AND TECHNOLOGY**

**Department of Computer Engineering**

**Year 2014-2015**



**CERTIFICATE OF APPROVAL FOR**

**Project Synopsis**

This to certify that

Neil Pereira 44

Sneha Varghese 57

Joel Ulahanna 62

Have satisfactorily carried out the Project work entitled “Implementation of JNS Algorithm“ in partial fulfillment of Bachelor of Engineering in Computers as laid down by University of Mumbai during the academic year 2014-2015

**Internal Guide: Mr. Sankusu Sharma HOD: Mrs Neeta Patil**

**Internal Examiner External Examiner**

**Principal**

**Acknowledgement**

Today, we cannot find appropriate words that will express deep sense of gratitude and satisfaction.

We are indebted to our inspiring Mr. Sankusu Sharma who has extended all valuable guidance, help and constant encouragement through various difficult stages for the development of the project.

We express our sincere gratitude to our respected principle Dr.Satish Takalikar, Mr. Gunshekhar sir, Anita Sundarrajan ma’am and Neeta Patil ma’am for encouragement and facilities provided to us.

We would also like to acknowledge the patience that our ever beloved parents have shown during our efforts and the encouragement we have received from them.

Thus we are fully obliged and convey our thanks to the teaching and as well as non-teaching staff of the department. Special thanks to all the lab assistants for helping us with and problem developed by the computers in the lab and assisting, helping us to solve any problems generated on the spot. Last but not the least we would like to thank all direct and indirect identities of the college with whom we took the strides for this successful project.

**Abstract**

The Proposed project is based on the RSA and Modified RSA encryption algorithms that had been developed or even researched previously. The Proposed project aims at providing a better counterpart of the above two algorithms on the basis of the time complexity and security. For secure communication over the network, there are various cryptographic techniques but the most famous public cryptographic technique is RSA cryptography. Cryptography is knowledge of protecting the information for providing encryption techniques. It is the most advantageous solution for the security of information in computer network. For providing confidentiality in the network and to improve the Security of RSA algorithm, development and implementation of Modified RSA (MREA) algorithm was done.

The proposed method tries to decrease the time taken by the MREA for encryption and decryption. This is done with the help of Offline Storage and GNU library. Offline storage basically means generation of keys prior to the commencement of the encryption process and stores it in a database.

Use of GNU library aims at using pre defined functions defined in BigInteger library so as to decrease the time taken by the MREA algorithm. The proposed system will be trying to use either each of them independently or in combination so as to get the best result. MREA is one of the strongest algorithm when it comes to security but the time taken by it is a little longer as compared to other algorithms. So the basic focus of this system will be to decrease the time taken in the entire process taking the help of above mentioned three concepts.

.

**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Content** | **Page No** |
| 1.0 | Introduction | 7 |
| 2.0 | Literature Survey | 8 |
| 3.0 | Existing Systems | 9 |
| 4.0 | Problem Statement | 11 |
| 5.0 | Scope Of The Project | 12 |
| 6.0 | Constraints Of the Project | 13 |
| 7.0 | Proposed System | 14 |
| 8.0 | Methodology | 16 |
| 9.0 | Requirements Gathering And Planning | 19 |
| 10.0 | Analysis | 23 |
| 11.0 | Design | 27 |
| 12.0 | Appendix | 28 |
| 13.0 | Bibliography And References | 29 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Fig. No** | **Name of Figure** | **Page No** |
| 1 | Basic Architecture of proposed system | 14 |
| 2. | Architecture diagram of the algorithm | 15 |
| 3 | Methodology (incremental diagram) | 16 |
| 4 | Use case diagram | 19 |
| 5 | Block diagram | 22 |
| 6 | Class Diagram | 23 |
| 7 | State diagram | 24 |
| 8 | Activity Diagram | 25 |
| 9 | Sequence Diagram | 26 |
| 10 | UI Design | 27 |

**CHAPTER 1: INTRODUCTION**

This chapter would just briefly look into the introduction of the project requirements needed for it and its purpose and aim. Also, a simple development plan for the prototype system which was drafted out is being presented here in this chapter. An overview of the system initially planned to be developed is also being presented here.

**1.1 AIM AND OBJECTIVES**

As mentioned earlier RSA is one of the most popular encryption algorithm that was being devised. MREA was a further advancement in the RSA algorithm which increased the strength of the security that RSA was providing the users with. On the other hand MREA took more time to encrypt and decrypt data.

The main objective of this project is to devise a time efficient strong cryptographic algorithm. The main feature of this proposed system will be that it will be more secure compared to its counterparts and less time consuming than the MREA algorithm. The aim of the project is to assure secure transmission of data over a network without a third party intervention or in other words the messages sent would only be understood by the sender and to the intended receiver. That is, it would be difficult for a third party to understand the message even if he/she the cipher text. Apart from the security point of view the other main domain which our project aims is the time taken by the algorithms to compute the key generation, encryption and decryption calculations. The proposed system will try to decrease the time taken by the MREA algorithm

**CHAPTER 2: LITERATURE SURVEY**

Several encryption algorithms were being studied that were been researched and implemented by other researchers. For further understanding on other methods and techniques, refer to the reference page at the end of this report to search for the published papers.

The first paper is on the famous encrypting RSA algorithm which basically deals with the RSA algorithm and its merits and demerits. The second paper is on the MREA algorithm which is basically the modification on the RSA algorithm and it uses four prime numbers for key generation. Also it has many keys as compared to only two keys in RSA the encryption and decryption. These in turn increases the security of the algorithm but on the other hand the time required for encryption and decryption is increased. The time complexity is six times more than that of RSA algorithm.

The next paper is based on using the offline Storage on RSA algorithm with the purpose of decreasing the time required by the RSA algorithm. This strategy aims at computing all the keys required by the algorithm and storing then in tables prior to the commencement of the encryption process so that when the actual process starts the system can retrieve the keys and speed up the process of encryption and decryption. Thus the time required for the key generation can be saved and the algorithm works faster than normal RSA.

GNU MP Library is basically a library which has lot of pre defined arithmetic functions. The GNU MP library suggests that the use of these functions to compute the cipher text will speed up the process of encryption and decryption thereby reducing the Time complexity of the RSA algorithm. These functions include modulus, addition, subtraction, power functions and many more functions.

**CHAPTER 3: EXISTING SYSTEM**

There are many encrypting algorithms which help in transmitting secured information around an in secured network. Following are already existing algorithms that has been considered for this project

* **RSA Algorithm**

Named after its inventor RSA encryption algorithm is one of the most popular algorithm used for the security of the information sent across a network. It is a public key or asymmetric key cryptography which basically uses two prime numbers for its key generation. Two keys are generated which are public 'e' and private 'd' keys which are related to each other. The strength of the RSA lies in the fact that it is very hard to find factors for large numbers. Even a powerful hacking attack like brute force will also take a lot number of tries in order to get the right numbers used for key generation. The encryption and the decryption of the RSA takes only less numbers of modulus and the keys generated for the same are also very less as compared to others.

Algorithm for the same can be given as:

1. Select two prime numbers p and q.

ii. Find n=p\*q, Where n is the modulus that is made public. The length of n is as the considered RSA key length.

iii. Choose a random number ‘e’ as a public key in therange 0<e<(p-1)(q-1) such that

gcd(e,(p-1)(q-1))=1.

iv. Find private key d such that ed\_1(mod (p-1)(q-1)).

**Encryption**

i. Consider the device A that needs to send a message to B securely.

ii. Let e be B’s public key. Since e is public, A has access to e.

iii. To encrypt the message M, represent the message as an integer in the range 0<M<n.

iv. Cipher text C = Me mod n, where n is the modulus.

**Decryption**

i. Let C be the cipher text received from A.

ii. Calculate Message M = Ce mod n, where d is B’s private key and n is the modulus.

* **Modified RSA Encryption Algorithm (MREA)**

To increase the security of the existing RSA a Modified version of the RSA was being devised. Modified algorithm introduced additional keys so as to encrypt and decrypt the data. It also introduced lot of modulus operation which makes the computation much more complex but the security of the entire system is increased to great extend. Because if it was difficult to find two factors of the a particular number then it becomes a lot more difficult to compute four factors. So thus MREA provides better security as compared to RSA.

Algorithm for the same can be given as:

**A. Key Generation Algorithm:**

i. Choose four large prime numbers p, q, r and s randomly and independently of each other. All primes should be of equivalent length.

ii. Compute n = p x q, m= r x s, phi = (p-1) x (q-1) and λ=(r-1) x (s-1).

iii. Choose an integer *e*, 1 < e <phi, such that gcd (e,phi) = 1.

iv. Compute the secret exponent *d*, 1 < d <phi, such that e x d mod phi=1.

v. Select an integer g=m+1.

vi. Compute the modular multiplicative inverse: µ = µ-1 mod m.

vii. The public (encryption) key is (n, m, g, e).

viii The private (decryption) key is (d, µ, λ).

**B. Encryption:**

i. Let m be a message to be encrypted where 0<mesg < n.

ii. Select random r where r < m.

iii. Compute cipher text as: c=g mesg^emod nx rm mod m2

**C. Decryption**

i. Compute message: m = (((cλ mod m2 – 1)/ m) xµmod m)dmod n.

**CHAPTER 4: PROBLEM STATEMENT**

The RSA and the MREA algorithm are the two existing algorithms that have been considered for this proposed project. MREA algorithm makes use of four prime numbers for key generation. Also the algorithm has some additional number of variables like µ, λ, g whose computation again require time. The Time Complexity of RSA is O(k3) whereas that of MREA is O(k3+k), this extra computation time is because of the additional operation like MOD, POWER, INVERSE MOD, etc. Because of the above mentioned computations the simulation results shows that MREA takes six times more *time* for the entire process of encryption and decryption than that of original RSA.

**CHAPTER 5: SCOPE OF THE PROJECT**

The Scope of this project is to increase the time efficiency of MREA by using ideas such as Offline Storage and GNU library. In Offline Storage method the system will compute all the keys required for the algorithm before the encryption process and store in a database so as to saves the computation time for key generation for the next encryption process.

**CHAPTER 6: CONSTRAINTS OF THE PROJECT**

* **Data:**
* The maximum input block size is 128 bits.
* The key size can be either 128,192 or 256 bit only.
* **Language**:
* The application and the algorithm will be designed using JAVA language.

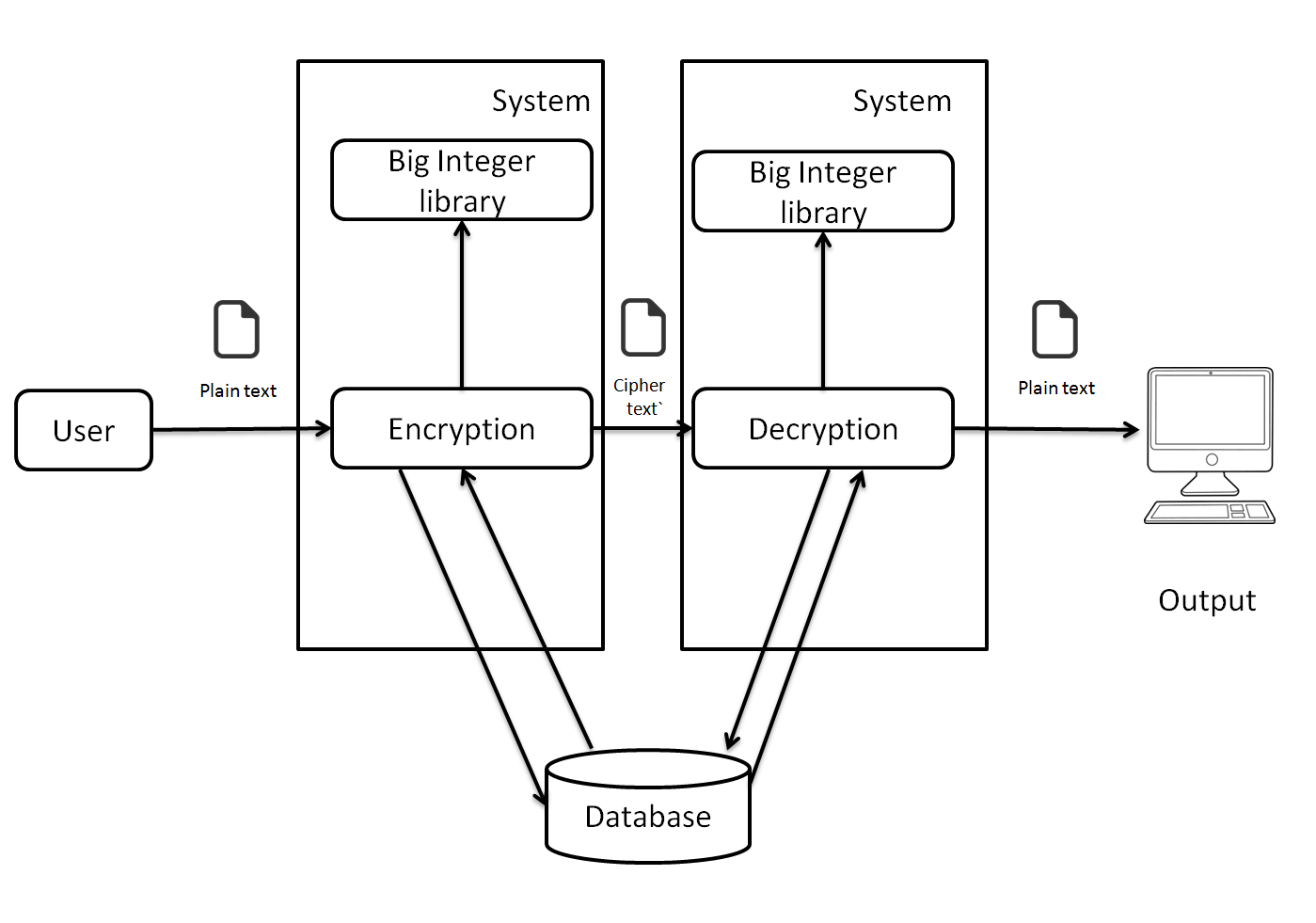
**CHAPTER 7: PROPOSED SYSTEM**

The proposed work makes use of following three main concepts which will help in decreasing the time taken by the existing MREA algorithm:-

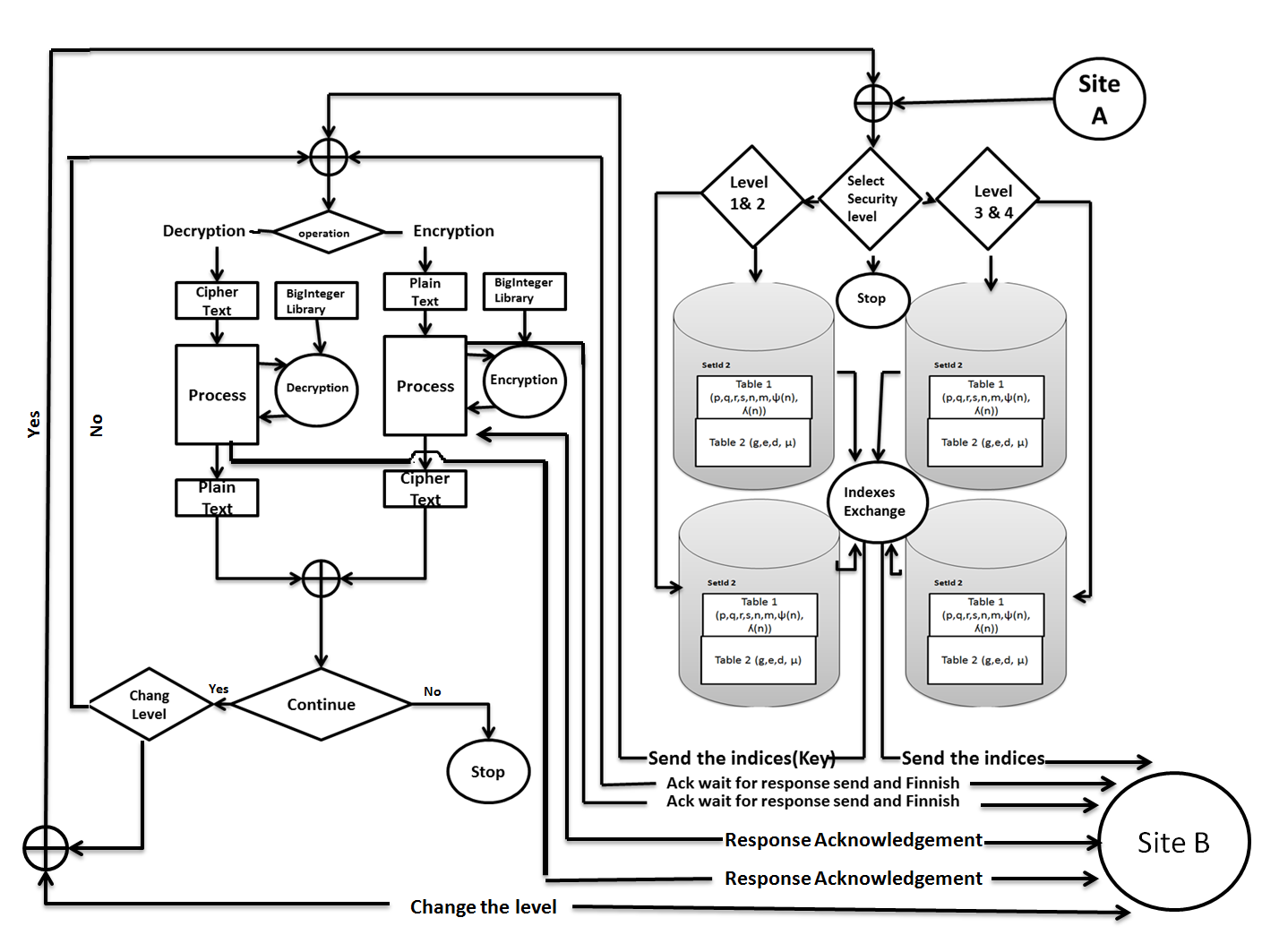
1. GNU MP Library

2. Offline Storage

MREA consists of many new variables that have to be computed during key generation. The computation of the same requires some modulus of large numbers and other arithmetic calculation. GNU MP Library makes use of pre defined functions so as to ease and decrease the time taken by the algorithm for calculation. Whereas Offline Storage intends to calculate and store the required keys in a database prior to the commencement of the encryption process. Thus in this case all the keys are already ready when the algorithm is working and just encryption and decryption has to be done which increases the speed.



**Fig 1: Basic Architecture of proposed system**



**Fig 2 : Architecture diagram**

**Proposed System :**

1. **Select Security Level :**

|  |  |
| --- | --- |
| **Security Level** | **Key Length(Bit)** |
| Level 1 | 512 |
| Level 2 | 1024 |
| Level 3 | 2048 |
| Level 4 | 4096 |

1. **Key Generation**

* Choose four large prime numbers p, q, r and s randomly and independently of each other. All primes should be of equivalent length.
* By using the database provided the values of n, m , phi, λ is decided
* .They are computed as
  + Compute n = p x q, m= r x s, phi = (p-1) x (q-1) and λ=(r-1) x (s-1).
  + Choose an integer *e*, 1 < e <phi, such that gcd (e,phi) = 1.
  + Compute the secret exponent *d*, 1 < d <phi, such that e x d mod phi=1.
  + Select an integer g=m+1.
  + Compute the modular multiplicative inverse: µ = µ-1 mod m.

1. **Index Exchange :**

* We now exchange Nid ,Mid, Eid ,Did, gid, λid (using Deffi-Hellmen)

1. **Encryption**

* Retrieve the public key (n, m, g, e) from the database.
* Encrypt the data by using the functions of BigInteger library.
* That is c=g mesg^emodnxrm mod m2

1. **Decryption**

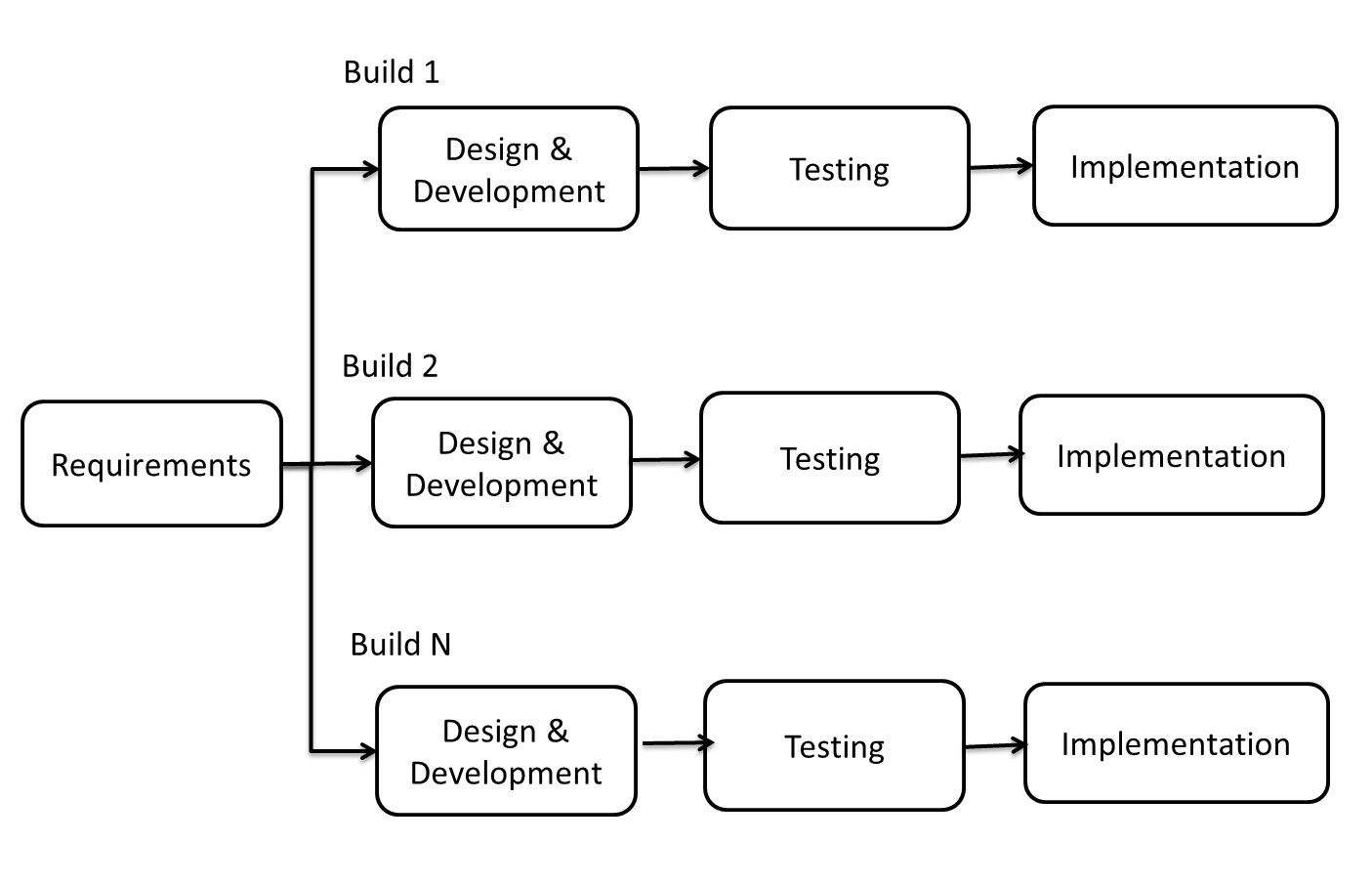
* Retrieve the public key (d,µ, λ) from the database.

1. Decrypt the data by using the functions of bigInteger library.

That is m== (((cλ mod m2 – 1)/ m) xµmod m)dmod n

**CHAPTER 8: METHODOLOGY**

The Proposed System will be built on the incremental model. The incremental model is a risk-driven [process model](http://en.wikipedia.org/wiki/Software_development_process) generator for software projects. Based on the unique risk patterns of a given project, the incremental model guides a team to adopt elements of one or more process models, such as [waterfall](http://en.wikipedia.org/wiki/Waterfall_model), or [evolutionary prototyping](http://en.wikipedia.org/wiki/Evolutionary_prototyping).



**Fig 3 : Methodology (Incremental diagram)**

**Steps involved in the System Development Life Cycle :**

Below are the steps involved in the System Development Life Cycle.  Each phase within the overall cycle may be made up of several steps.

**Step 1: Software Concept**

The first step is to identify a need for the new system.  This will include determining whether a business problem or opportunity exists, conducting a feasibility study to determine if the proposed solution is cost effective, and developing a project plan.

This process may involve end users who come up with an idea for improving their work. Ideally, the process occurs in tandem with a review of the organization's strategic plan to ensure that IT is being used to help the organization achieve its strategic objectives.  Management may need to approve concept ideas before any money is budgeted for its development.

**Step 2: Requirements Analysis**

Requirements analysis is the process of analyzing the information needs of the end users, the organizational environment, and any system presently being used, developing the functional requirements of a system that can meet the needs of the users.  Also, the requirements should be recorded in a document, email, user interface storyboard, executable prototype, or some other form.  The requirements documentation should be referred to throughout the rest of the system development process to ensure the developing project aligns with user needs and requirements.

Professionals must involve end users in this process to ensure that the new system will function adequately and meets their needs and expectations.

**Step 3: Architectural Design**

After the requirements have been determined, the necessary specifications for the hardware, software, people, and data resources, and the information products that will satisfy the functional requirements of the proposed system can be determined.  The design will serve as a blueprint for the system and helps detect problems before these errors or problems are built into the final system. Professionals create the system design, but must review their work with the users to ensure the design meets users' needs.

**Step 4: Coding and Debugging**

Coding and debugging is the act of creating the final system.  This step is done by software developer.

**Step 5: System Testing**

The system must be tested to evaluate its actual functionality in relation to expected or intended functionality.  Some other issues to consider during this stage would be converting old data into the new system and training employees to use the new system.  End users will be key in determining whether the developed system meets the intended requirements, and the extent to which the system is actually used.

**Step 6: Maintenance**

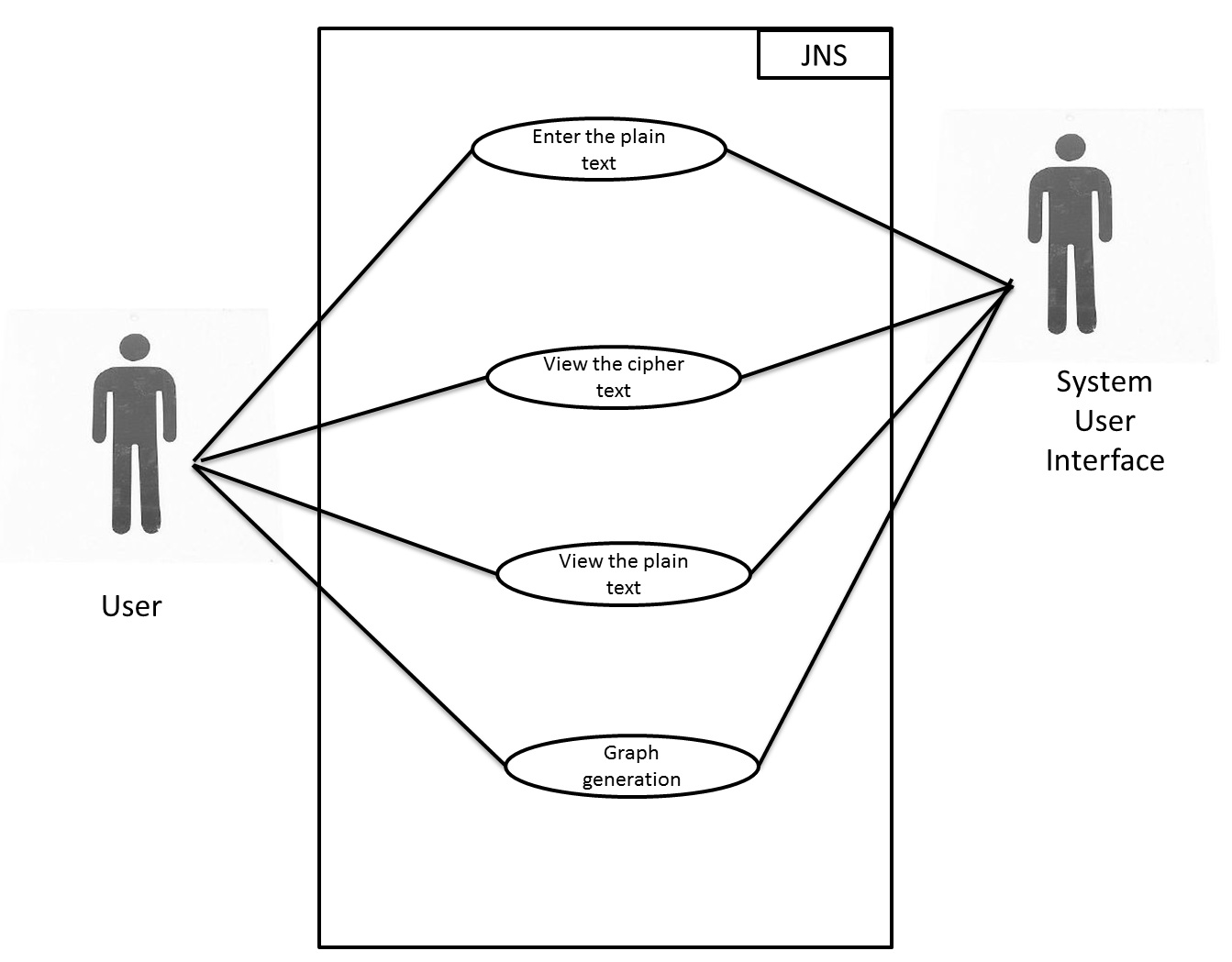
Inevitably the system will need maintenance. Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period.

**CHAPTER 9: REQUIREMENT GATHERING AND**

**PLANNING**

**9.1 Requirement Elicitation**

**9.1.1 Use Case Diagram and description**

****

**Fig 4 : Use case diagram**

|  |  |  |
| --- | --- | --- |
| **SR.NO** | **ACTORS** | **USE CASE DESCRIPTION** |
| 1 | User | * Enter the plain text * View the cypher text * View the plain text * Graph generation |
| 2 | System | * Select 4 large prime numbers * Key generation * Storage * Encryption * Decryption |

**9.2 FEASIBILITY STUDY**

The very first phase in any system developing life cycle is preliminary investigation. The feasibility study is a major part of this phase. A measure of how beneficial is our proposed algorithm to reduce the time taken by the MREA for encryption and decryption process

* + 1. **TECHNICAL FEASIBILITY**
    - At least 1.2 GHz Pentium Processor or Intel compatible processor.
    - At least 256 MB RAM.
    - Keyboard.
    - At least 10GB free hard disk space.
    - At least JAVA Software-1.7
    - NetBeans IDE 7.4.
    1. **ECONOMICAL FEASIBILITY**

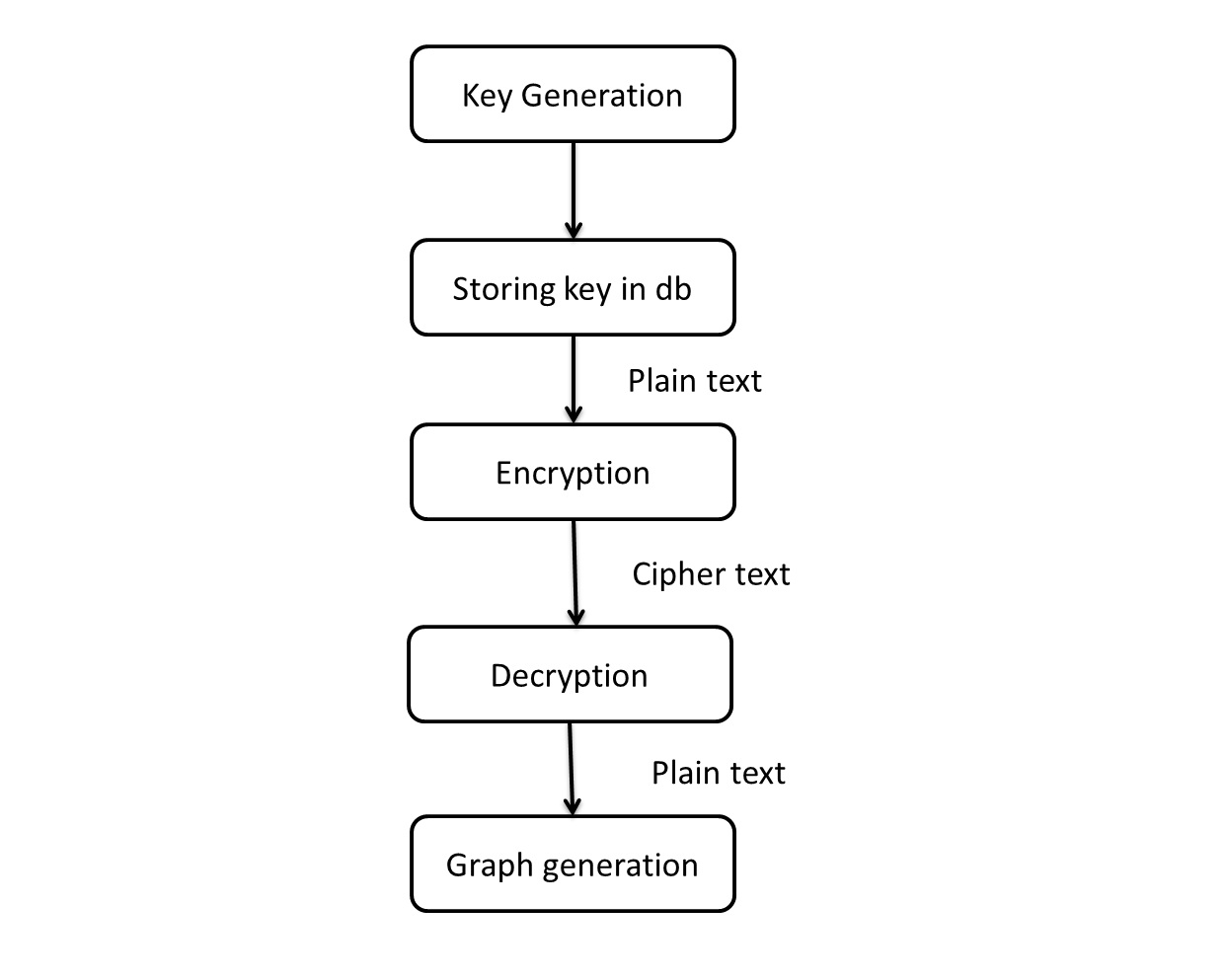
The Proposed algorithm reduces the time taken for encryption and decryption as compared to MREA, so the CPU time thus can be effectively used for other time consuming process.

Once the hardware and software requirements get fulfilled, there is no need for the user of our system to spend for any additional overhead.

For the user, the proposed system will be economically feasible in the following aspects:

* The System will reduce a lot of time. Hence the cost will be reduced.
  1. **REQUIREMENT ANALYSIS**

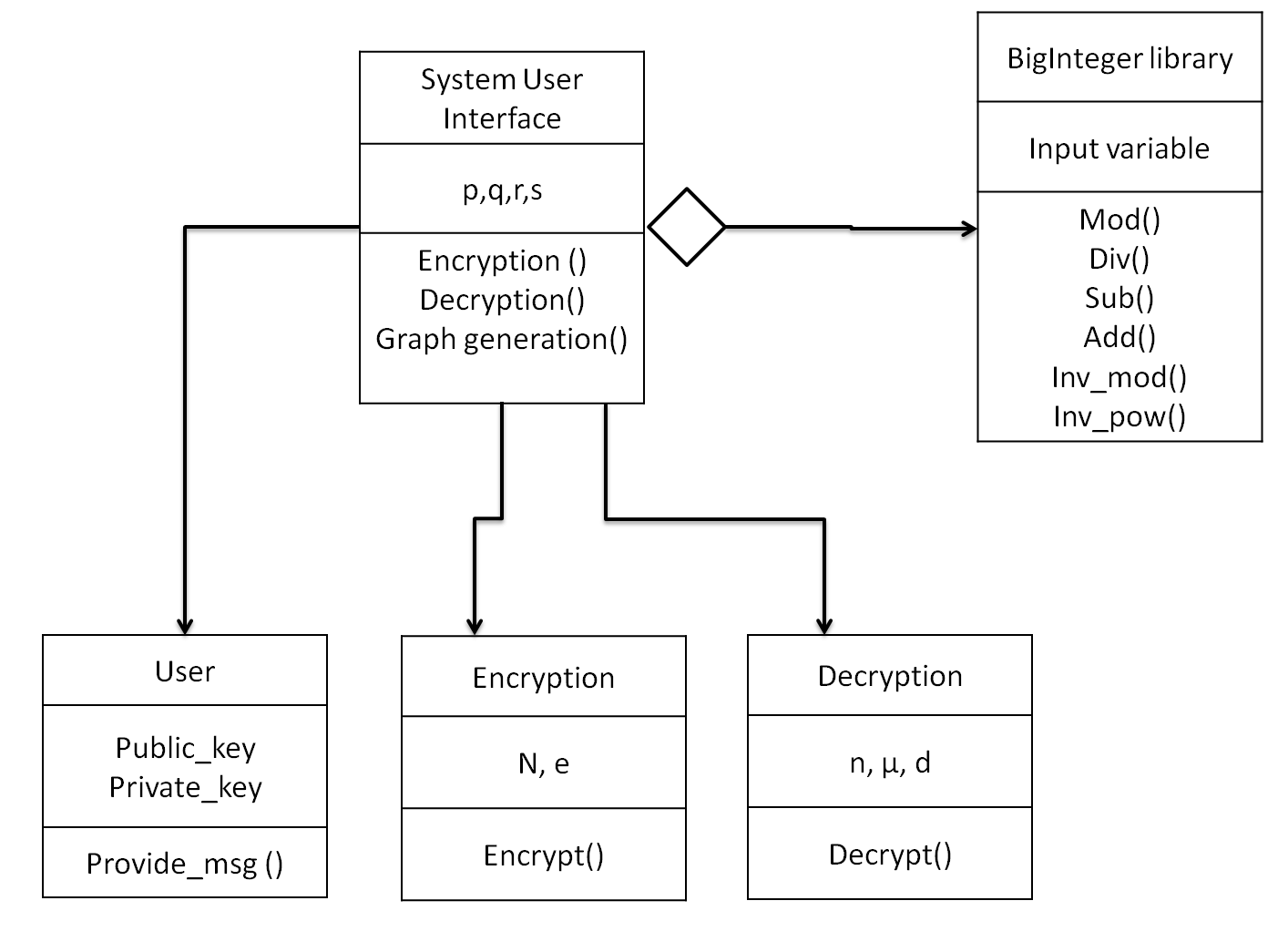
**9.3.1 BLOCK DIAGRAM**



**Fig 5 : Block Diagram.**

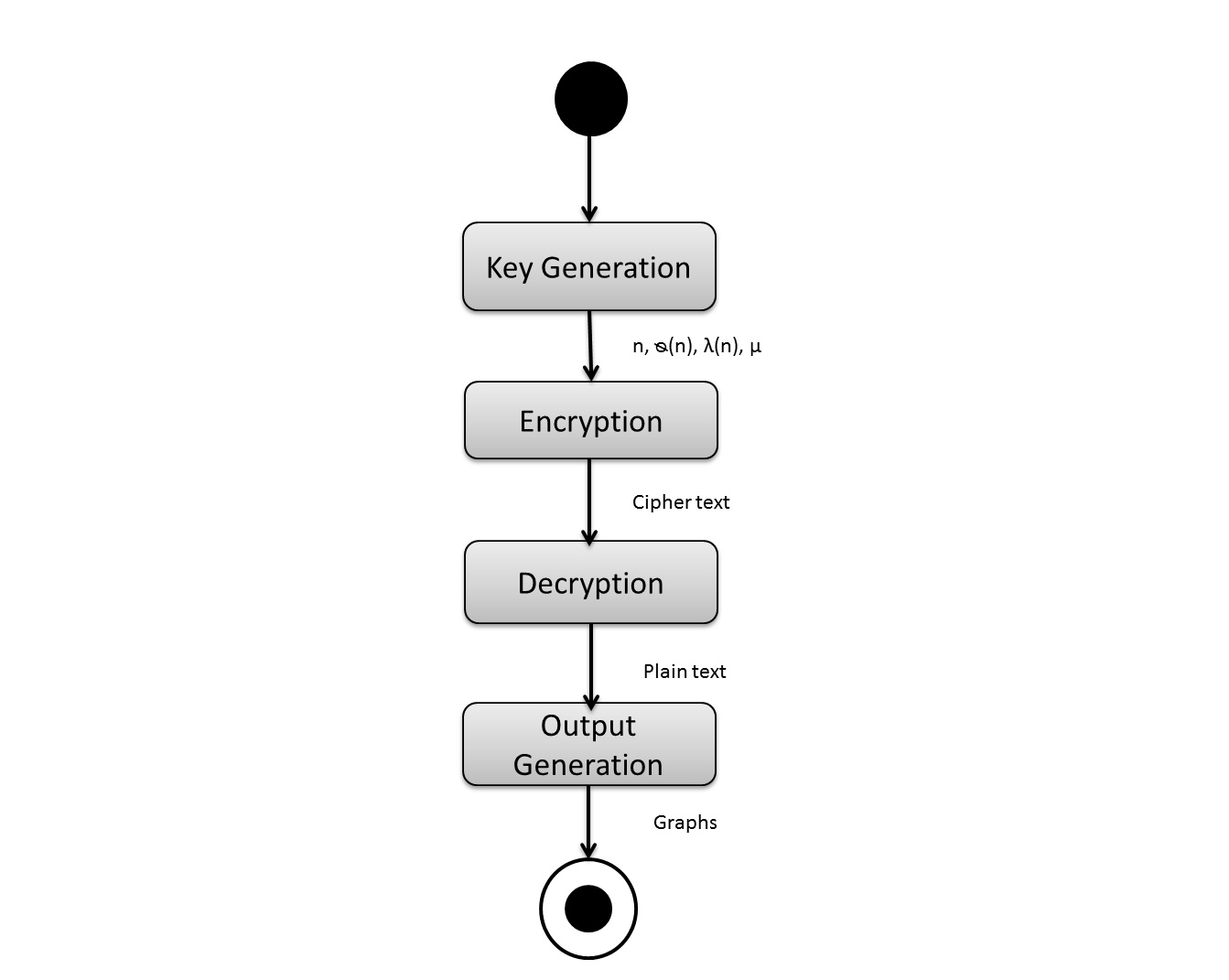
**10.0 Analysis**

**10.1 Class Diagram**

****

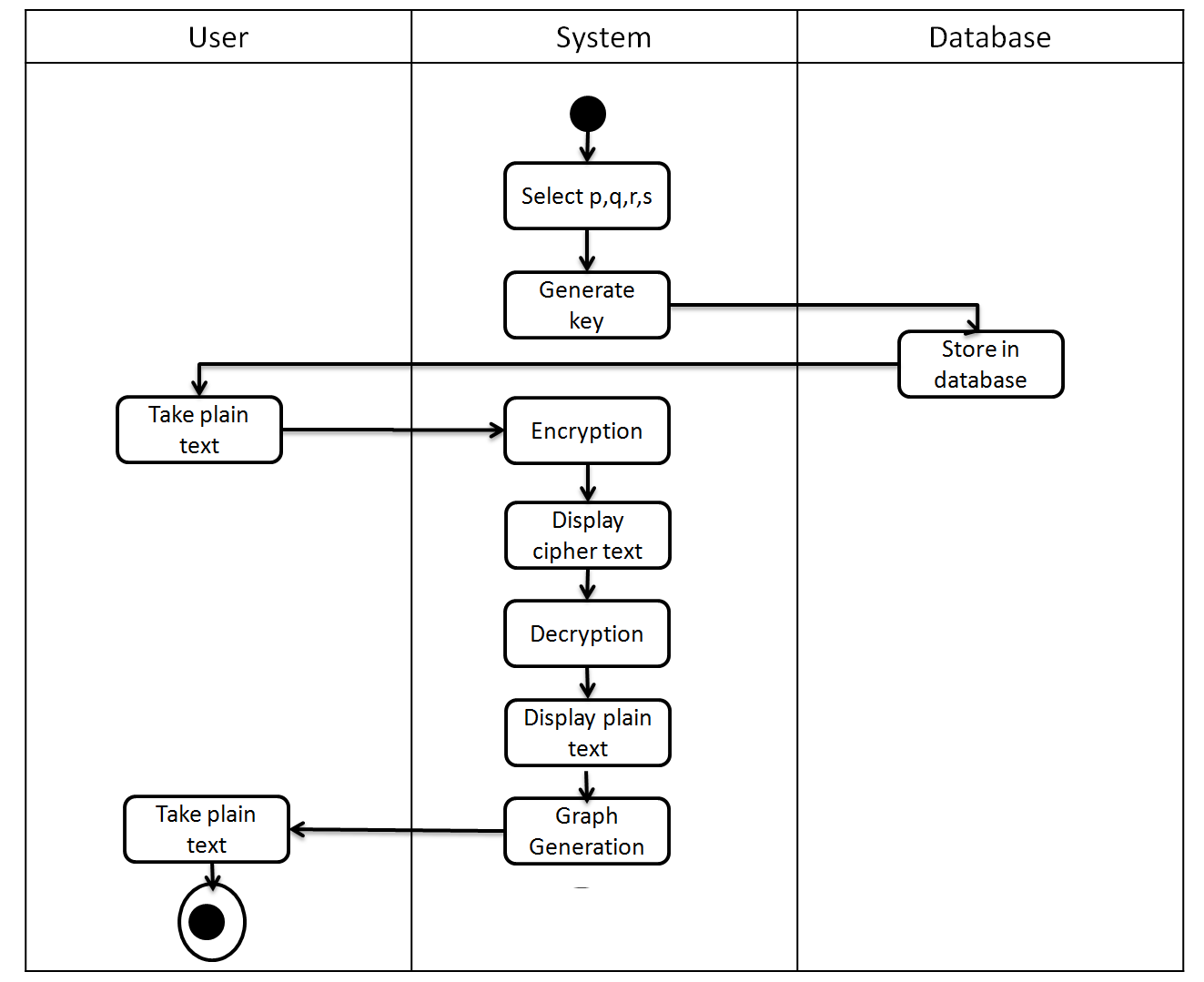
**Fig 6 : Class diagram**

**10.2 State Diagram**



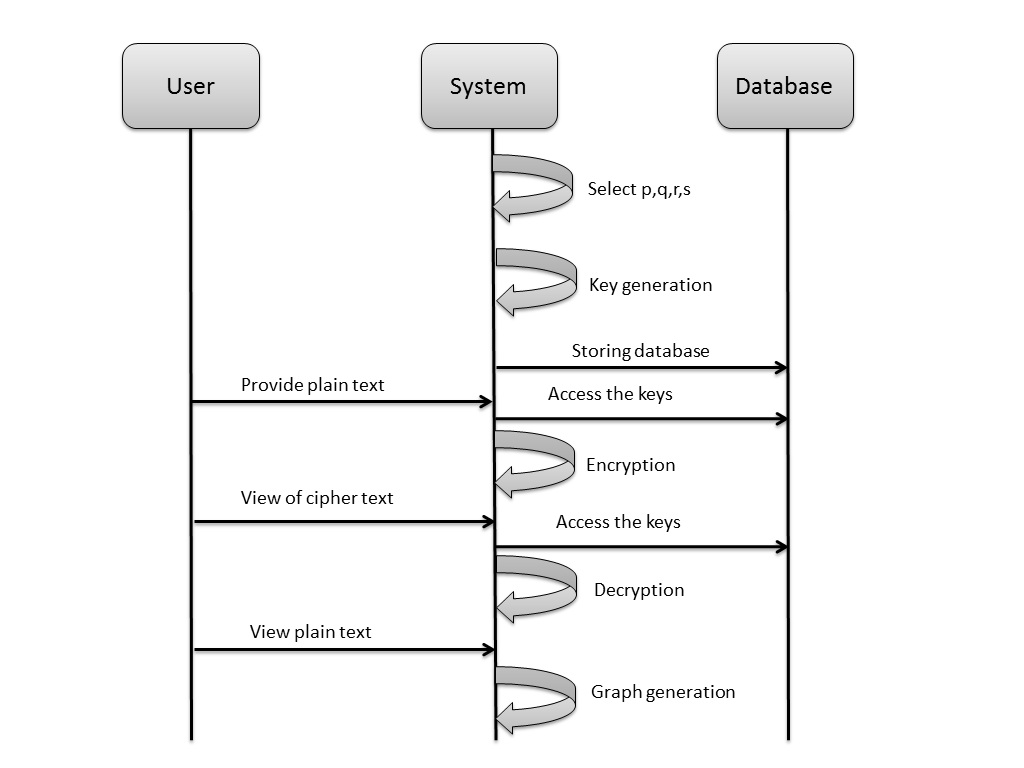
**Fig 7 : State diagram**

**10.3 Activity Diagram**

****

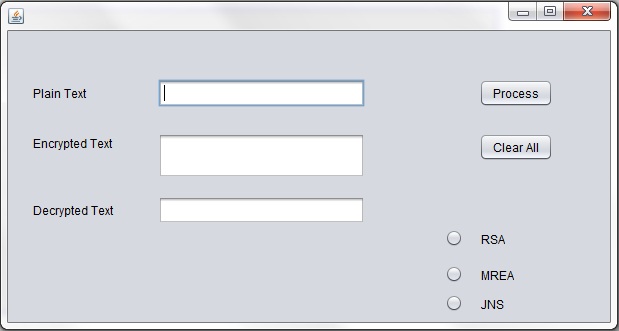
**Fig 8 : Activity diagram**

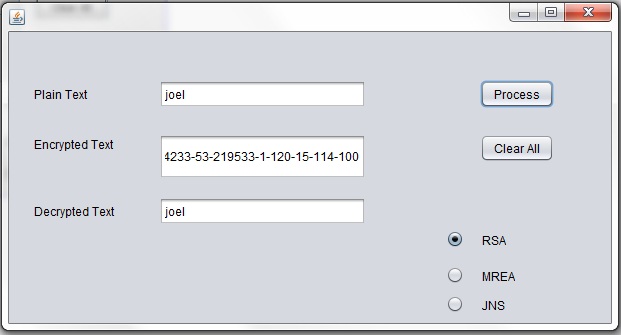
**10.4 Sequence Diagram**



**Fig 9 : Sequence diagram**

**11.0 Design**

**11.1 UI Design **

****

**Fig 10: User Interface**

**12.0 APPENDIX**

**RSA: -**

Encryption algorithm using two prime numbers

**MREA : -**

Encryption algorithm using four prime numbers

**JNS : -**

Proposed Algorithm in which we use Big integer libraries and offline storage system

1. **Bibliography and References**

[1] Ravi Shankar Dhakarand and Prashant Sharma, “modified RSA encryption algorithm (MREA)”. Amit Kumar Gupta Sbcet, Jaipur, Rajasthan, India, 2012.

[2] Rajorshi Biswas Shibdas and Bandyopadhyay Anirban Banerjee, “A Fast Implementation Of

The RSA Algorithm Using the GNU MP Library”, iiit-Calcutta, 2010.

[3] MS. Ritu Patidar and Mrs. Rupali Bhartiya, “Offline Storage and Prime Number “, Institute of

Technology & Science, Indore, India

[4] Salah Zaher, amr badr & Ibrahim Farag, “Performance Enhancement Of RSA Cryptography

Algorithm By Membrane Computing”*, Cario Egypt.*

[5] Evgeny Milanov,” The RSA Algorithm”, 3 June 2009.

[6] *William Stallings,”Cryptology and Network Security”, an imprint of Pearson Education.*