

# **SECURE DATA ENCRYPTION**

*A*

*Major Project Report*

*Submitted*

*In partial fulfillment*

*For the award of the Degree of*

## **BACHELOR OF TECHNOLOGY**

*In Department of Computer science and Engineering*

**Submitted By:**

PIHU JAIN  
ISHIKA JAIN  
KOMOLIKA  
AGARWAL

**Submitted To:**

Mr. Aditya Maheshwari



**Department of Computer Science and Engineering**  
**TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY**

**RAJASTHAN TECHNICAL UNIVERSITY**

**May 2022**



# TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY

## CERTIFICATE

This is to certify that this project report “**Secure Data Encryption**” is the confide work of “**Komolika Agarwal , Pihu Jain , Ishika Jain**” who have carried out the project work under my supervision. I approve this project for submission of the Bachelor of Technology in the **Department of Computer Science and Engineering, Techno India NJR Institute of Technology**, affiliated to Rajasthan Technical University, Kota.

**Mr. Aditya Maheshwari**

**Project In charge**

Department of Computer Science

# **ABSTRACT**

## **1. Purpose**

### ***1.1. Introduction***

This Software Requirements Specification provides a complete description of all the functions and specifications of the Secure Data Encryption Application.

The main objective of secure data encryption is to defend private data , sensitive information and to enhance the security of communication between client apps and server.

### ***1.2. Scope***

Scope of this project is very broad in terms of data protection and security of communication .Few of them are:-

- It is used in cybersecurity to defend against brute-force and cyber-attacks
- It ensures the confidentiality, integrity and confirmation of data transmission over the network.

## **2. Document overview**

The remainder of this document is 8 chapters, the first providing introduction of the project. It lists all the functions performed by the application. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter gives testing for the project. The seventh chapter tells about the conclusion and future enhancements of the project. The final chapter concerns with the bibliography.

## **ACKNOWLEDGEMENT**

It gives me immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed **guide Mr. Aditya Maheshwari (Project In charge, TINJRIT)** for their valuable guidance, encouragement and help for completing this work. Their useful suggestions for this whole work and co-operative behavior are sincerely acknowledged.

I would like to express my sincere thanks to Mr. Aditya Maheshwari (Project In charge, TINJRIT) for giving me this opportunity to undertake this project.

I also wish to express my indebtedness to my parents as well as my family member whose blessings and support always helped me to face the challenges ahead.

At the end I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

**Place: Udaipur**

**Date: 28/05/22**

**Pihu Jain**

**Ishika Jain**

**Komolika Agarwal**

# TABLE OF CONTENTS

## Table of Contents

Abstract .....	iv
Acknowledgement.....	v
Table of Contents.....	vi
List of Tables.....	viii
List of Figures .....	ix
List of Symbols.....	x
<b>CHAPTER – 1 Introduction.....</b>	<b>1-5</b>
Introduction.....	2
1.1.Purpose.....	2
1.2.Overall Description.....	4
<b>CHAPTER – 2 Software Requirement Specification.....</b>	<b>6-20</b>
2.1. Purpose.....	7
2.2. Overall Description .....	9
2.2.1. Functional Requirements definitions .....	9
2.2.2. Use Cases .....	9
2.2.3. Non – Functional Requirements .....	14
2.3. Requirement Specification .....	15
2.3.1. External interface specifications .....	15
2.3.2. Functional Requirements .....	15
2.4. Hardware Specification.....	19
2.5. Software Specification .....	19
2.6. Hardware and Software Requirements in detail.....	20
<b>CHAPTER – 3 System Analysis and Design.....</b>	<b>21-39</b>
3.1. Study & Weaknesses of current system.....	22

3.2 Requirements of new system .....	23
3.3. Feasibility Study .....	23
3.4. Features of new System .....	24
3.5. UML Modeling.....	34
3.5.1. Context Diagram .....	36
3.5.2. Sequence Diagram.....	37
<b>CHAPTER – 4 Data Dictionary .....</b>	<b>40-45</b>
4.1. Table Details .....	41
4.2. E-R Diagram .....	45
<b>CHAPTER - 5 Screen Shots .....</b>	<b>46-61</b>
5.1. Admin Master.....	47
5.2. User Master.....	54
<b>CHAPTER – 6 Testing .....</b>	<b>62-67</b>
<b>CHAPTER – 7 Conclusion and Future Enhancements .....</b>	<b>68-71</b>
7.1. Limitations .....	69
7.2. Future Enhancements .....	70
7.3. Conclusion .....	71
<b>CHAPTER – 8 Bibliography .....</b>	<b>72</b>

## List of Tables

<b>Table No.</b>	<b>Table Name</b>	<b>Page No.</b>
Table 2.1	Glossary	7
	<b>Use Cases</b>	
Table 2.2	Register page	15
Table 2.3	User login	15
Table 2.4	Private key import	16
Table 2.5	File import	18
Table 2.6	Folder Import	25
Table 2.7	Create new encrypted file	41
	<b>Database Design</b>	
Table 4.1	Admin login	43
Table 4.2	Store Encryption key	44

## List of Figures

<b>Figure No.</b>	<b>Figure Name</b>	<b>Page No.</b>
	<b>Use Cases</b>	
Figure 1.1	Register page	10
Figure 1.2	Enter Certificate	10
Figure 1.3	User login	12
Figure 1.4	Private key import	13
	<b>UML Modelling</b>	
Figure 2.1	Context Diagram	36
	<i>Sequence Diagram</i>	
Figure 2.2	AES 1	37
Figure 2.3	AES 2	37
Figure 2.4	RSA	38
Figure 2.5	E-R Diagram	45

***CHAPTER – I***  
***INTRODUCTION***

# **Introduction**

## **1.1. Purpose**

### ***1.1.1. Introduction***

This Software Requirements Specification provides a complete description of all the functions and specifications of the secure data encryption application.

The main objective of secure data encryption application is to defend the sensitive information against the brute force attack including cyber attacks, including malware and ransomware.

### **1.1.2. Scope**

Scope of this project is very broad in terms of the security of data on the network layer .Few of them are:-

- It can be used by the individual users and large corporations to protect user information sent between browser and server.
- It can also use to decrypt the logs of machines and software's that are going on live site which is very useful for engineers.

### **1.1.3. References**

[1] D. Seth, L. Ramanathan, and A. Pandey, “Security enhancement: Combining cryptography ,” International Journal of Computer Applications (0975–8887) Volume, 2010.

[2] M. H. Rajyaguru, “Cryptography-combination of cryptography with rapidly changing keys,” International Journal of Emerging Technology and Advanced Engineering, ISSN, pp. 2250–2459, 2012.

[3] [Wiley] ASP.NET Bible by MridulaParihar.

### **1.1.4. Document overview**

The remainder of this document is 8 chapters, the first providing introduction of the project. It lists all the functions performed by the system. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter gives testing for the project. The seventh chapter tells

about the conclusion and future enhancements of the project. The final chapter concerns with the bibliography.

This document is meant for describing all the features and procedures that were followed while developing the system.

This document specially mentions the details of the project how it was developed, the primary requirement, as well as various features and functionalities of the project and the procedures followed in achieving these objectives.

Digital communication witnesses a noticeable and continuous development in many applications in the Internet. Hence, secure communication sessions must be provided. The security of data transmitted across a global network has turned into a key factor on the network performance measures. So, the confidentiality and the integrity of data are needed to prevent eavesdroppers from accessing and using transmitted data. Data encryption and decryption are two important techniques that are used to provide network security.

The IT initiatives have encouraged various Organizations to develop systems to enhance the security of data for making communication's confidentiality and integrity enriched. The Application is developed for hiding the information in the form of cipher text from unauthorized modes.

## 1.2. Overall description

Cryptography is one of the traditional methods used to guarantee the privacy of communication between parties. This method is the art of secret writing, which is used to encrypt the plaintext with a key into ciphertext to be transferred between parties on an insecure channel. Using a valid key, the ciphertext can be decrypted to the original plaintext. Without the knowledge of the key, nobody can retrieve the plaintext. Cryptography plays an essential role in many factors required for secure communication across an insecure channel, like confidentiality, privacy key exchange, and authentication

### 1.2.1. Functional requirements definitions

Because sometimes we just need a simple library to Encrypt(), Decrypt(), Sign() and Verify() without a lot of research into correct settings and arcane flags.

This library came out of personal need to do similar data protection operations on several projects, and getting tired of the boilerplate, and of trying to remember how to set up the same settings every time.

This library wraps the built-in cross platform .NET System.Security.Cryptography libraries, in the following way:

- Provides *safe, modern defaults* for all algorithms
- Provides *easy API with typed containers* instead of plain byte arrays, to prevent common mistakes (see below)
- Provides initialization, boilerplate, and teardown for all operations
- Hides legacy algorithms
- Hides configuration options
- Single-file implementation that can be embedded in projects

Some API highlights:

- Encrypt() function will encrypt some data with a given secret key, and add a signature to detect tampering
  - Signatures are optional and can be skipped without affecting encryption
- Decrypt() function will decrypt data, and if a signature is present, check it and report whether it matches
- Sign() and Verify() functions just compute/check anti-tampering signatures of any byte array
- Create Key\*() functions securely generate encryption keys (from string passwords, from byte array sources, or completely random)
- Hash() computes strong hashes of any byte array
- Random() computes cryptographically random sequence of bytes
- \*.Save() and \*.Load() function for persisting encrypted data to a flat byte array, and loading it back up

Finally, the built-in .NET cryptography APIs can be tricky, with many configuration options, variously operating on streams and at other times on byte arrays, or relying on disposable classes for even simplest operations. This API standardizes access and hides those implementation details.

***CHAPTER – II***

***SOFTWARE REQUIREMENT SPECIFICATION***

# **Software Requirement Specification**

## **2.1. Purpose**

### ***2.1.1. Introduction***

This Software Requirements Specification provides a complete description of all the functions and specifications of the Secure data encryption application.

The main objective of secure data encryption application is to defend the sensitive information against the brute force attack including cyberattacks, including malware and ransomware.

### ***2.1.2. Scope***

Scope of this project is very broad in terms of the security of data on the network layer .Few of them are:-

- It can be used by the individual users and large corporations to protect user information sent between browser and server.
- It can also use to decrypt the logs of machines and software's that are going on live site which is very useful for engineers

### ***2.1.3. Glossary***

**Table 2.1**

<b>Term</b>	<b>Definition</b>
Admin	The only user who has the permission to encrypt and decrypt data in the application.
Entry	Admin stored in the Database
IEEE	Institute of Electrical and Electronic Engineers

QA	Quality assurance
SCMP	Software Configuration Management Plan
SDD	Software Design Document
SQAP	Software Quality Assurance Plan
SRS	Software Requirements Specification
Web Site	A place on the world wide web

#### ***2.1.4. References***

- [1] D. Seth, L. Ramanathan, and A. Pandey, “Security enhancement: Combining cryptography ,” International Journal of Computer Applications (0975–8887) Volume, 2010.
- [2] M. H. Rajyaguru, “Cryptography-combination of cryptography with rapidly changing keys,” International Journal of Emerging Technology and Advanced Engineering, ISSN, pp. 2250–2459, 2012
- [3] [Wiley] ASP.NET Bible by MridulaParihar.

#### ***2.1.5. Document overview***

The remainder of this document is two chapters, the first providing a full description of the project for the owners of the Secure data encryption. It lists all the functions performed by the application. The final chapter concerns details of each of the system functions and actions in full for the software developers’ assistance. These two sections are cross-referenced by topic; to increase understanding by both groups involved.

## **2.2.Hardware Specification**

### **Client Side:**

- ✓ Internet Explorer: 6.0
- ✓ Processor: Pentium IV 2.0 and above.
- ✓ RAM : 512 MB
- ✓ Hard Disk : 80GB

### **Server Side:**

- ✓ Processor: Pentium IV 2.0 and above.
- ✓ RAM : 1 GB
- ✓ Disk space : 4GB

## **2.3. Software Specification**

### **Client Side:**

- ✓ .NET Framework
- ✓ Web Browser
- ✓ Windows XP/Vista/Windows 7

### **Web Server:**

- ✓ .NET Framework

- ✓ Windows XP/Vista/windows 7

**Data Base Server:**

- ✓ SQL Server

#### **2.4. Hardware and Software Requirements in detail**

**Hardware Requirements:**

- ✓ Processor: Pentium IV 2.0 and above.
- ✓ Internet Explorer: 6.0
- ✓ RAM : 512 MB
- ✓ Hard Disk : 80GB
- ✓ Disk space : 4GB

**Software Requirements:**

- Microsoft Visual Studio 2015
  - ✓ Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It can be used to develop console and graphical user interface applications along with Windows Forms applications, web sites, web applications, and web services in both native code together with managed code for all platforms supported by Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silverlight.
  - ✓ .NET Framework 4.1
  - ✓ Visual C#
- Web Browser Internet Explorer 6.0 and above
- Windows XP Service pack 3/Vista/Windows 7

***CHAPTER – III***  
***SYSTEM ANALYSIS AND DESIGN***

## System Analysis and Design

### 3.6.3 Context Diagram

The context diagram is a top-level view of an information system that shows the boundaries and scope. It describes the main objective of the system and the entities involved.

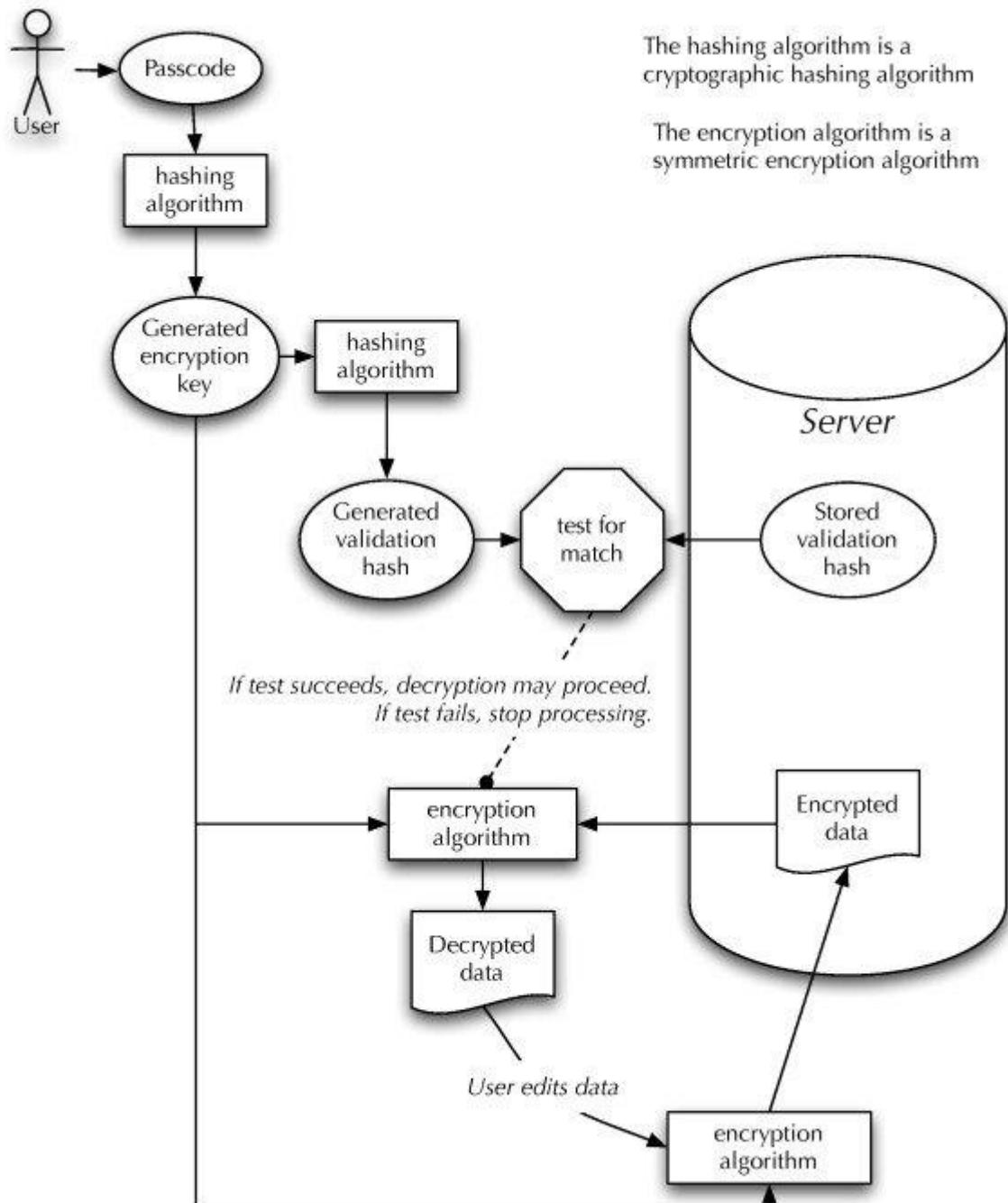
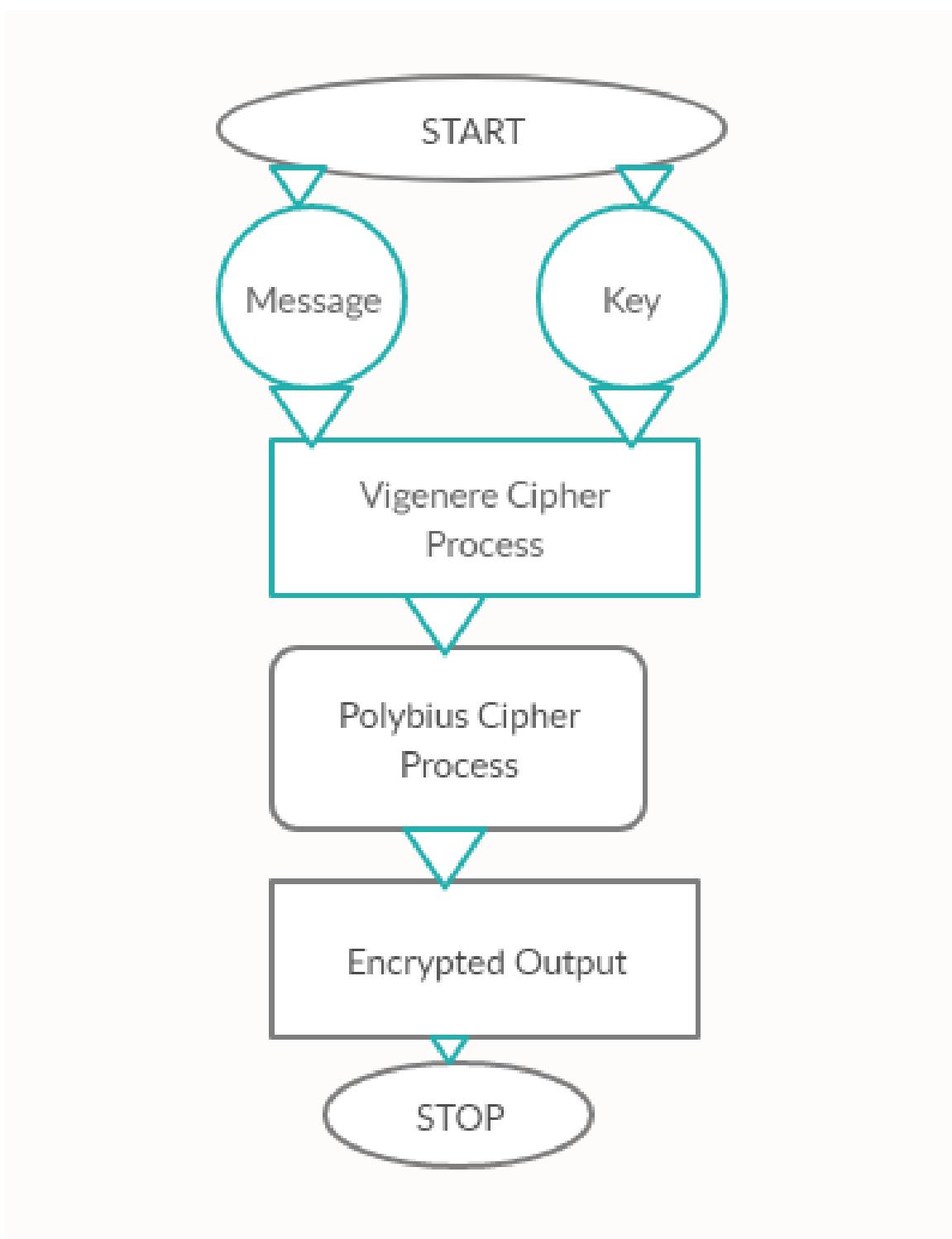
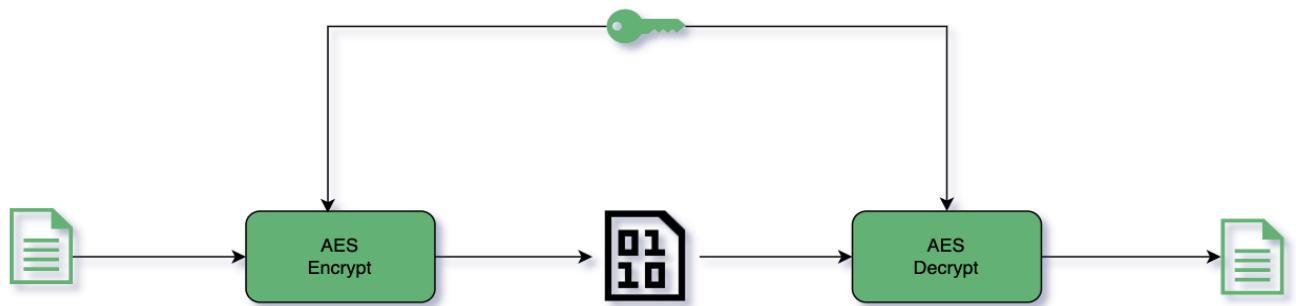


Fig. 3.13 Context Diagram

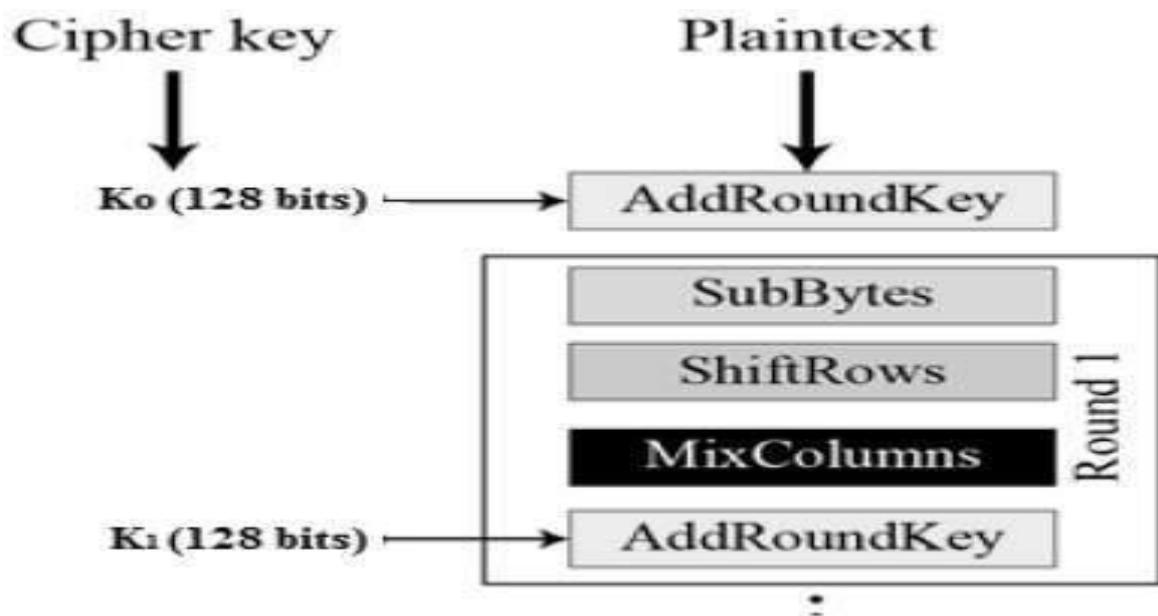
### 3.6.4. Sequence Diagrams



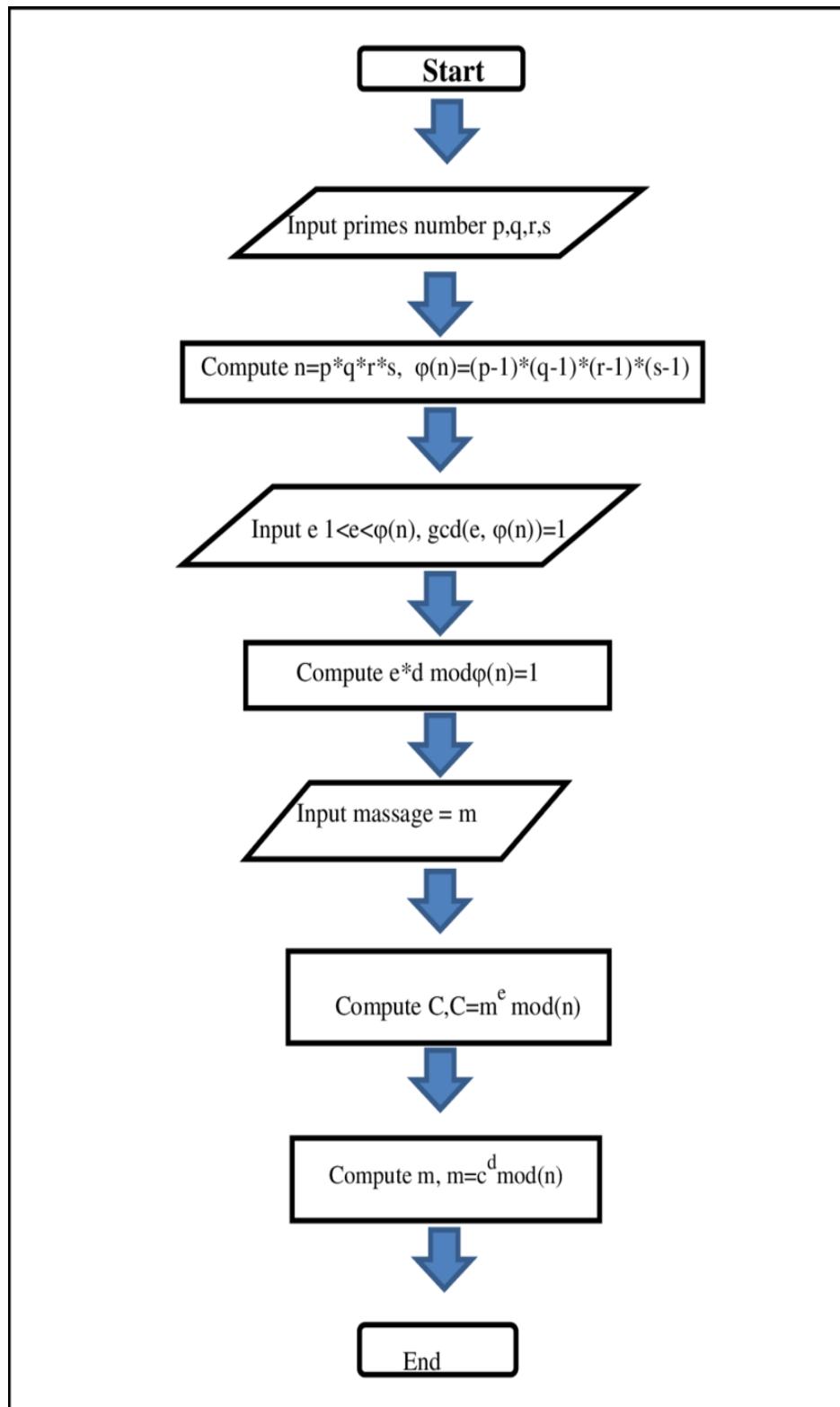
### 3.6.4 a.Sequence Diagram for Aes Algorithm



### 3.6.4 b.Sequence Diagram for Aes Algorithm



### 3.6.4d. Sequence diagram for Rsa Algorithms



***CHAPTER – IV***  
***DATA DICTIONARY***

## **Data Dictionary**

A data dictionary is a catalog-a-repository of the elements in a system. As the name suggests, their elements center on data and the way they are structured to meet user requirements and organization needs. In a data dictionary you will find a list of all the elements composing the data flowing through a system. The major elements are data flows, data stores and processes. The data dictionary stores details and descriptions of these elements.

If analysis want to know characters are in a data item by what other names it is referenced in the system, or where it is referenced in the system, or where it is issued in the system, they should be able to find the answers in issued in the system, they should be able to find the answer in properly developed data dictionary.

The Dictionary contains two types of description for the data following through the system.

### **1. Data Elements**

The most fundamental data is the elements. They are building blocks for all other data in the system. Data elements are also alternatively known as fields, data item or elementary item.

### **2. Data Structure**

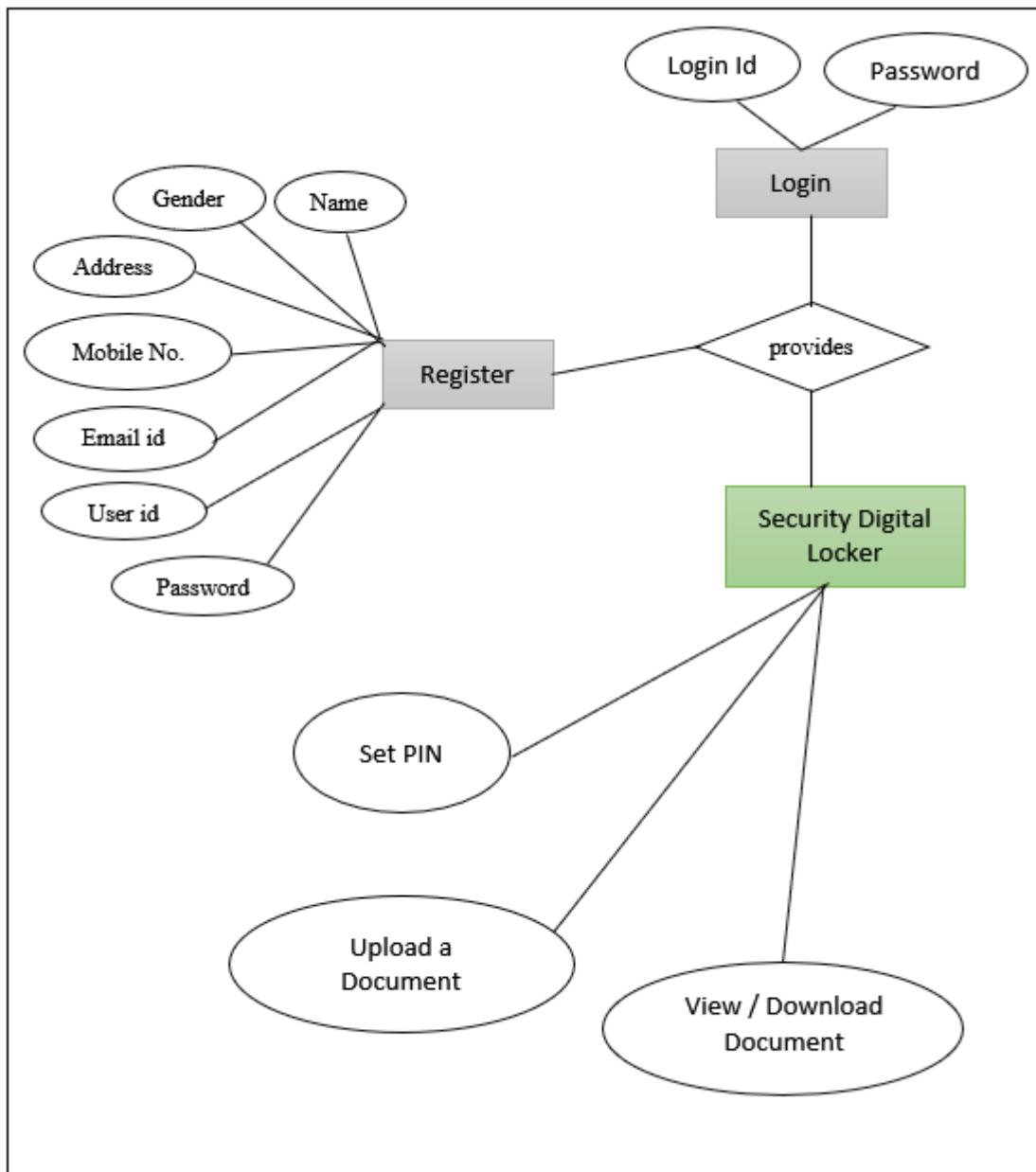
A data structure is a set if items that are related to one another and described a components in the system.

### **4.1. Table Details**

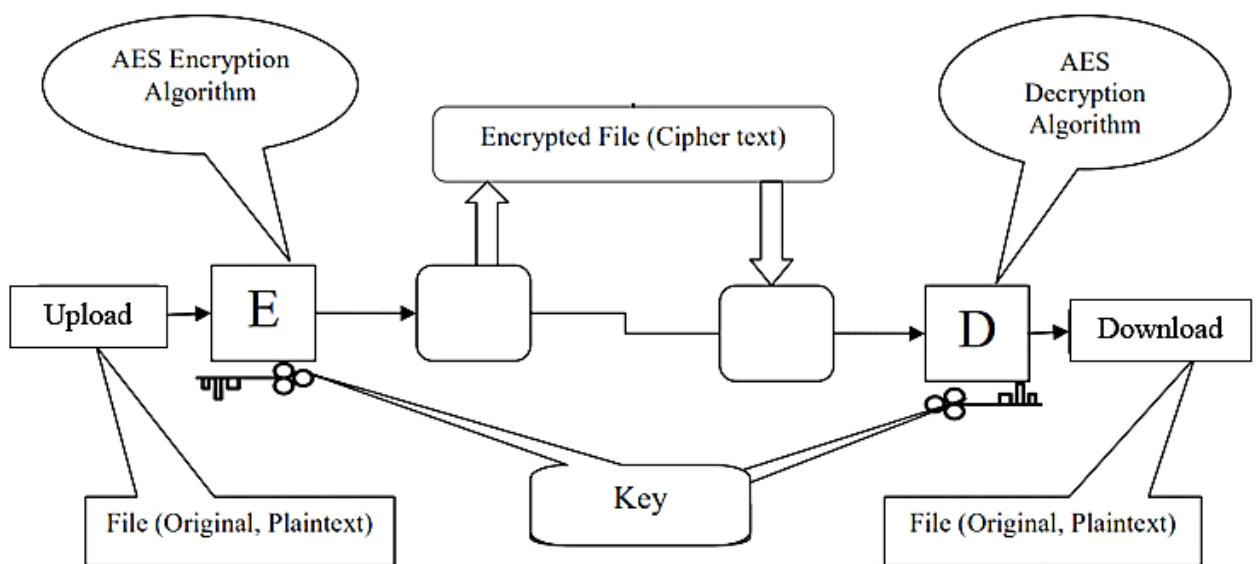
**Table 4.1. Login**

<b>Field Name</b>	<b>Description</b>	<b>Constraints</b>	<b>Size</b>	<b>Data Type</b>
Username	Unique username of the user		50	varchar
Password	User password		50	varchar

#### 4.2. E-R Diagram

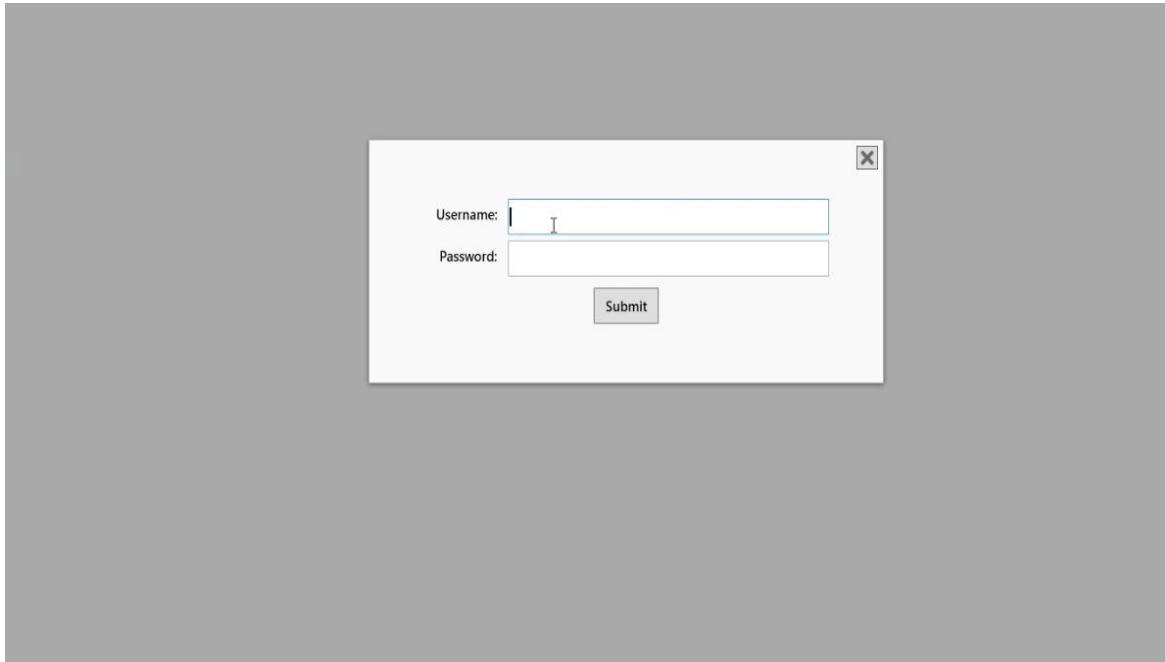


**Fig. 4.1 E-R Diagram**

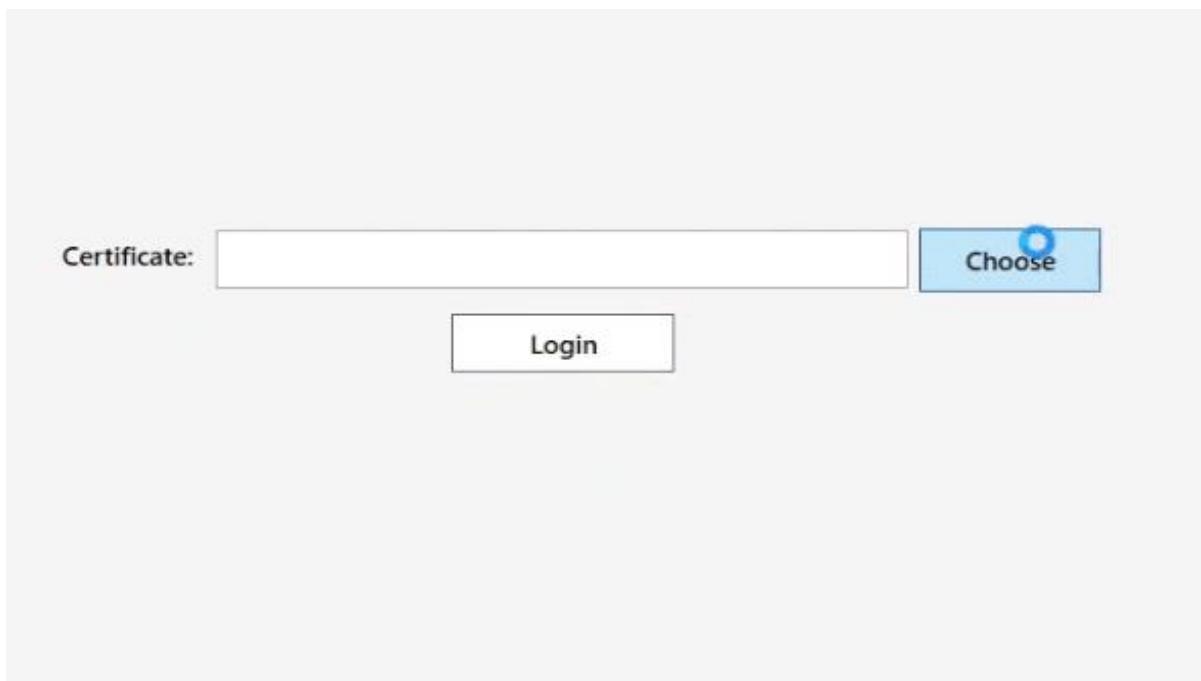


***CHAPTER – V***  
***SCREEN SHOTS***

### **5.1.1. Login**



### **5.1.2. Choose Certificate**



### 5.1.3.

A screenshot of a user interface for naming a folder. On the left, the text "Folder name:" is followed by a text input field containing the letters "ne". To the right of the input field is a grey button labeled "Enter". In the top right corner of the window frame, there is a small square icon with a black "X" inside it.

### 5.1.4.

A screenshot of a user interface for importing files. At the top, there is a text input field labeled "File:" followed by a "Choose" button. The "Choose" button is highlighted with a blue background and has a cursor arrow pointing to its center. Below this, there are two dropdown menus: "Algorithm:" set to "AES-256" and "Hash:" set to "SHA256". Further down, there is another dropdown menu labeled "Mode:" set to "OFB" and a checkbox labeled "Delete original file" which is unchecked. At the bottom center of the interface is a large grey button labeled "Import". In the top right corner of the window frame, there is a small square icon with a black "X" inside it.

File name:

This is my secret text.

Algorithm:  Mode:  Hash:

Private Key:

Key Password:

***CHAPTER – VI***

***TESTING***

## Testing

### Testing Methodology

Companies rely on software more than ever to provide and manage information with strategic and operational importance and to provide key decision support. Rising customer expectations for fault-free, requirements-exact software have increased awareness of the importance of software testing as a critical activity.

We begin the testing process by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process very

fies that the application meets the requirements specified in the system requirements document and is bug free. At the end of each testing day, we prepare a summary of completed and failed tests. Applications are not allowed to launch until all identified problems are fixed. A report is prepared at the end of testing to show exactly what was tested and to list the final outcomes.

Our software testing methodology is applied in three distinct phases: unit testing, system testing, and acceptance were testing.

**Unit Testing:** The programmers conduct unit testing during the development phase. Programmers can test their specific functionality individually or with other units. However, unit testing is designed to test small pieces of functionality rather than the system as a whole. This allows the programmers to conduct the first round of testing to eliminate bugs before they reach the testing staff. In unit testing the analyst tests the programs making up a system.

For this reason, unit testing is sometimes called program testing. Unit testing gives stress on the modules independently of one another, to find errors. This helps the tester in detecting errors in coding and logic that are contained within that module alone. The errors resulting from the interaction between modules are initially avoided.

For example, a hotel information system consists of modules to handle reservations; guest checking and checkout; restaurant, room service and miscellaneous charges; convention activities; and accounts receivable billing. For each, it provides the ability to enter, modify or retrieve data and respond to different types of inquiries or print reports. The test cases needed for unit testing should exercise each condition and option.

Unit testing can be performed from the bottom up, starting with smallest and lowest-level modules and proceeding one at a time. For each module in bottom-up testing a short program is used to execute the module and provides the needed data, so that the module is asked to perform the way it will when embedded within the larger system.

**System Testing:** The objective of system testing is to ensure that all individual programs are working as expected, that the programs link together to meet the requirements specified and to ensure that the computer system and the associated clerical and other procedures work together.

The initial phase of system testing is the responsibility of the analyst who determines what conditions are to be tested, generates test data, produced a schedule of expected results, runs the tests and compares the computer produced results with the expected results with the expected results.

The analyst may also be involved in procedures testing. When the analyst is satisfied that the system is working properly, he hands it over to the users for testing. The importance of system testing by the user must be stressed. Ultimately it is the user must verify the system and give the go-ahead.

During testing, the system is used experimentally to ensure that the software does not fail, i.e., that it will run according to its specifications and in the way users expect it to. Special test data is input for processing (test plan) and the results are examined to locate unexpected results.

A limited number of users may also be allowed to use the system so analysts can see whether they try to use it in unexpected ways. It is preferably to find these surprises before the organization implements the system and depends on it. In many organizations, testing is performed by persons other than those who write the original programs. Using persons who do not know how certain parts were designed or programmed ensures more complete and unbiased testing and more reliable software.

The system is tested as a complete, integrated system. System testing first occurs in the development environment but eventually is conducted in the production environment. Functionality and performance testing are designed to catch bugs in the system, unexpected results, or other ways in which the system does not meet the stated requirements.

The testers create detailed scenarios to test the strength and limits of the system, trying to break it if possible. Editorial reviews not only correct typographical and grammatical errors, but also

improve the system's overall usability by ensuring that on-screen language is clear and helpful to users. Accessibility reviews ensure that the system is accessible to users with disabilities.

System testing consists of the following five steps:

- i. Program testing
- ii. String testing
- iii. System testing
- iv. System documentation
- v. User acceptance testing

### **Program Testing**

A program represents the logical elements of a system. For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. It is the responsibility of a programmer to have an error free program. At

The time of testing the system, there exists two types of errors that should be checked. These errors are syntax and logic.

A syntax error is a program statement that violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted key words are common syntax errors. These errors are shown through error messages generated by the computer. A logic error, on the other hand, deals with incorrect data fields out of range items, and invalid combinations.

Since the logical errors are not detected by compiler, the programmer must examine the output carefully to detect them. When a program is tested, the actual output is compared with the expected output. When there is a discrepancy, the sequence of the instructions, must be traced to determine the problem. The process is facilitated by breaking the program down into self-contained portions, each of which can be checked at certain key points.

### **String Testing**

Programs are invariably related to one another and interact in a total system. Each program is tested to see whether it conforms to related programs in the system. Each part of the system is tested against the entire module with both test and live data before the whole system is ready to be tested.

## **System Testing**

System testing is designed to uncover weaknesses that were not found in earlier tests. This includes forced system failure and validation of total system as it will be implemented by its user in the operational environment. Under this testing, generally we

Take low volumes of transactions based on live data. This volume is increased until the maximum level for each transaction type is reached.

The total system is also tested for recovery and fallback after various major failures to ensure that no data are lost during the emergency.

All this is done with the old system still in operation. When we see that the proposed system is successful in the test, the old system is discontinued.

## **System Documentation**

All design and test documentation should be well prepared and kept in the library for future reference. The library is the central location for maintenance of the new system.

## **User Acceptance Testing**

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system's procedures operate to system specifications and that the integrity of important data is maintained. Performance of an acceptance test is actually the user's show. User motivation is very important for the successful performance of the system. After that a comprehensive test report is prepared. This report shows the system's tolerance, performance range, error rate and accuracy.

**Table 6.1 Test Report with test data**

<b>TEST REPORT WITH TEST DATA</b>		
(To be filled by System Analyst/Programmer)		
<b>Project Name : Online Examination System</b>		
S No.	Testing Parameter	Observations
A.	INTERFACE TESTING 1) User-friendliness 2) Consistent menus	OK NA
B.	CONTROL FLOW TESTING 1) IF-THEN-ELSE 2) DO WHILE 3) CASE-SWITCH	OK OK OK
C.	VALIDATION TESTING 1) Check for improper or inconsistent typing 2) Check for erroneous initialization or default values 3) Check for incorrect variable names 4) Check for inconsistent Data Types 5) Check for relational/arithmetic operators	OK OK OK OK OK
D.	DATA INTEGRITY/SECURITY TESTING 1) Data Insertion/ Deletion/ Updating 2) Boundary condition (Underflow, Overflow Exception) 3) Check for unauthorized access of data 4) Check for data availability	OK OK OK OK

E.	EFFICIENCY TESTING  1) Throughput of the system 2) Response time of the system 3) Online disk storage required by the system 4) Primary memory required by the system	OK OK OK OK
F.	ERROR HANDLING ROUTINES  1) Error description are intelligent/ understandable 2) Error recovery is smooth 3) All error handling routines are tested and executed at least once	OK OK OK

## ***CHAPTER – VII***

### ***CONCLUSION AND FUTURE ENHANCEMENTS***

## **7.1. Limitations**

The new system has been designed to meet almost all of the user requirements but it too has certain limitations some of which can be enhanced in the future enhancements or updates

Encryption proves to be very useful for securing information. This method of protecting data provides confidentiality, authentication, integrity, and non-repudiation of data.

Many of the government and law enforcement officials all around the world are insisting on the encryption back doors. As criminals and terrorists increasingly communicate through encrypted emails, it possesses a challenge to the government to decrypt the information

## **7.2 Future Enhancements**

Enhancements are the perquisite for development of a system. Every existing system has proposed enhancements which make it better and easier to use and more secure. The enhancements that have been proposed for this system are listed here.

- Implementation of converting very large file
- Implement relogin after 5 minutes of inactivity
- Implement password change
- Implement more cryptography algorithms for more security purposes.

## **7.3 Conclusion**

The development of software includes so many people like user system developer, user of system and the management, It is important to identify the system requirements by properly collecting required data to interact with supplier and customer of the system.

Proper design builds upon this foundation to give a blueprint, which is actually implemented by the developers.

On realizing the importance of systematic documentation all the processes are implemented using a software engineering approach. Working in a live environment enables one to appreciate the intricacies involved in the System Development Life Cycle (SDLC).

We have gained a lot of practical knowledge from this project, which we think, shall make us stand in a good state in the future.

***CHAPTER – VIII***  
***BIBLIOGRAPHY***

## Bibliography

- H.Abdulzahra, R. AHMAD, and N. M. NOOR, “Security enhancement; Combining cryptography and steganography for data hiding in images,” ACACOS, Applied Computational Science,pp.978-960,2014.
- P. R. Ekatpure and R. N.Benkar, “A comparative study of steganography & cryptography,”2013.
- M. H. Rajyaguru, “Cryptography-combination of cryptography and steganography with rapidly changing keys ,”International Journal of Emerging Technology and Advanced Engineering,ISSN ,pp.2250-2459,2012
- D. Seth. L. Ramanathan, and A.Pandey, “Security enhancement; Combining cryptography and steganography,” International Journal of Computer Applications(0975-8887)Volume,2010.
- J. V. Karthik and B. V. Reddy, “Authentication of secret information in image steganography,” International Journal of Computer Science and Network Security(IJCSNS), vol. 14, no. 6. P. 58. 2014.
- <https://www.thesslstore.com/blog/types-of-encryption-encryption-algorithms-how-to-choose-the-right-one/>
- <https://www.geeksforgeeks.org/encryption-its-algorithms-and-its-future/>
- <https://www.geeksforgeeks.org/encryption-its-algorithms-and-its-future/>