DATA ENCRYPTION AND DECRYPTION



A Course Project Report

in partial fulfillment of the subject

## Object-Oriented Programming through Java By

**Roll. No**. 19K41A04A9 **Name**: P. SHIVA PRIYA

Under the guidance of

## Dr. Ravi Chaythanya

Assistant Professor, Department of CSE.



# SR ENGINEERING COLLEGE

Ananthasagar, Warangal-506371.



**CERTIFICATE**

This is to certify that this project entitled **“DATA ENCRYPTION AND DECRYPTION**" is the bonafied work carried out by P. SHIVA PRIYAas a course project for the partial fulfillment to award the degree **BACHELOR OF TECHNOLOGY** in **ELECTRONICS & COMMUNICATION ENGINEERING** during the academic year 2021-2022 under our guidance and Supervision.

**Dr. Ravi Chaythanya Dr. Sandip Bhattacharya**

Assistant Professor, CSE, Assoc. Prof. & HOD (ECE),

S R Engineering College, S R Engineering College,

Ananthasagar, Warangal. Ananthasagar, Warangal.

# ACKNOWLEDGEMENT

We owe an enormous debt of gratitude tour project guide **Dr. Ravi Chaythanya, Assistant Professor** for guiding us from the beginning through the end of the Capstone project with their intellectual advices and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive, encouraging and ultimately drove us to the right direction.

We express our thanks to Head of the ECE Department **Dr. Sandip Bhattacharya, Associate Professor,** for his/her encouragement and support.

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved Principal, **Dr. V. Mahesh,** for his continuous support and guidance to complete this project in the institute.

Finally, we express our thanks to all the teaching and non-teaching staff of the department for their suggestions and timely support.

# ABSTRACT

During the last decades, information security has become a major issue. Encrypting and decrypting data have recently been widely investigated and developed because there is a demand for a stronger encryption and decryption which is very hard to crack. But most of the proposed algorithms encountered some problems such as lack of robustness and significant amount of time added to packet delay to maintain the security on the communication channel between the terminals. In this paper, the security goals were enhanced via "A New Approach for Complex Encrypting and Decrypting Data" which maintains the security on the communication

# CONTENTS

[ACKNOWLEDGEMENT iii](#_TOC_250008)

[ABSTRACT iv](#_TOC_250007)

Chapter No. Title Page No.

1. INTRODUCTION
   1. [INTRODUCTION 02](#_TOC_250006)
   2. EXISTING METHODS 03
   3. PRESENT WORK 04
2. LITERATURE SURVEY
   1. [RELATED WORK 05](#_TOC_250005)
   2. [SYSTEM STUDY 05](#_TOC_250004)
3. PROJECT IMPLEMENTATION
   1. [DESCRIPTION OF PROPOSED PROJECT 06](#_TOC_250003)
   2. [FLOW CHART 07](#_TOC_250002)
4. IMPLEMENTATION
   1. [MODULES 08-13](#_TOC_250001)
   2. [OVERVIEW TECHNOLOGY 13](#_TOC_250000)
5. RESULT ANALYSIS
   1. TEST CASES 14
6. CONCLUSION & FUTUTRE SCOPE
   1. CONCLUSION 15
   2. FUTURE SCOPE 15

**BIBLIOGRAPHY**

# CHAPTER 1 INTRODUCTION

## INTRODUCTION

In computer systems, the algorithm consists of complex mathematical formulas that dictate.

The rules of conversion process from plain text to cipher text and vice versa combined with the key. However, some of encryption and decryption algorithms use the same key (i.e., sender, and receiver) and in other encryption and decryption algorithms they use different keys but these keys must be related.

The major issue to design any encryption and decryption algorithm is to improve the security level. Therefore, this paper aims to propose a new algorithm to improve the security level and increase the performance by minimizing a significant amount of delay time to maintain the security and makes comparative study. Encryption is the method by which information is converted into secret code that hides the information's true meaning. The science of encrypting and decrypting information is called cryptography. In computing, unencrypted data is also known as [plaintext,](https://www.techtarget.com/searchsecurity/definition/plaintext) and encrypted data is called cipher text.

The formulas used to encode and decode messages are called encryption algorithms, or [ciphers](https://www.techtarget.com/searchsecurity/definition/cipher). To be effective, a cipher includes a variable as part of the algorithm. The variable, which is called a [key](https://www.techtarget.com/searchsecurity/definition/key), is what makes a cipher's output unique. When an encrypted message is intercepted by an unauthorized entity, the intruder has to guess which cipher the sender used to encrypt the message, as well as what keys were used as variables. The time and difficulty of guessing this information is what makes encryption such a valuable security tool. Encryption has been a longstanding way for sensitive information to be protected. Historically, it was used by militaries and governments.

In modern times, encryption is used to protect data stored on computers and storage devices, as well as data in transit over networks. Although their different kinds of security system data encryption and decryption is mostly used in today’s applications. Encryption is not only worth for text messages but also it is used in sending or receiving the images. This method helps to block the third party to hack the information between the sender and the receiver. This logic also used in military so that the information is secured and there will be less chances of hacking it.

# EXISTING SYSTEMS

### Data Security Software System

[Data Privacy Software](https://www.g2.com/categories/data-privacy) **—** Data privacy is a subsection of information security that has emerged from increasing compliance demand. These tools are used to help ensure sensitive information is stored properly. They also simplify the data delivery process for customers or users requesting access to the

information a company has on them. Data privacy solutions are excellent complements to data security tools, but are not sufficient in and of themselves to provide a well-rounded data protection solution.

[Data Center Security Software](https://www.g2.com/categories/data-center-security) **—** Data center security software typically provides two things: firewalls and intrusion management tools. Firewalls are common in many security solutions, protecting everything from networks to endpoint devices. Data center security software will provide a firewall but one specifically designed for data center computing resources and local networks. They will also provide intrusion detection and prevention systems to alert security operations personnel of a breach and expedite the remediation process.

### General Data Security System

Data security software secures and/or encrypts data, allowing only approved parties access to sensitive information about their business, employees, and customers. They allow administrators and management to create access control and perform security tests to audit existing security. Many databases can store encrypted backups in the case of a disaster for easy, secure recover. Mobile data security software provides similar features but adds increased security for mobile devices, communications, and authentication. They may also include mobile device or mobile application management to ensure standards are maintained and malicious software applications are restricted.

### Cloud Security System

Securing cloud data can be a rigorous task. There are many aspects of network, endpoint, and data security that intersect, making the security management process more difficult. Many of these tools will provide various security features such as a firewall, access control, or security monitoring capabilities. That being said, many are designed to simplify end-user security rather than securing data in transit or stored remotely. If you’re an individual concerned with the security of your personal communication data, internal communications software and secure messaging apps might be a cheaper and simpler solution.

* 1. **PROPOSED SYSTEM**

In this section, we will discuss proposed method which combines different hiding techniques, which is Cryptography. In this proposed method first, the message is encrypted by use. So, this technique combines the features of cryptography and provides a high level of security. It is better than either of the technique used separately. There will be an agreement between the sender and the receiver about the key for the concealment algorithm as well as the key for the encryption algorithm or these keys may be exchanged by a secure communication method. Our method starts by encryption first then hide encrypted data.

plain text

cipher text

plaint text

The primary purpose of encryption is to protect the confidentiality of digital data stored on computer systems or transmitted over the internet or any other computer network. In addition to security, the adoption of encryption is often driven by the need to meet compliance regulations. Decryption is the process after cipher text i.e., it shows the data hidden after encryption. The third person or the party cannot hack or enter to see or get the data sent. The encryption and decryption are purely a security-based system.

The encryption and decryption methodology can be used to in many ways. The proposed system shows the working of the data encryption and decryption. This kind of security system helps in maintain the data safely by protecting it from third party. Process of encryption takes place at the sender’s end. Process of decryption takes place at the receiver’s end. A public key or secret key is used in the process of Encryption. A secret key or private key is used in the process of Decryption. In encryption the sender sends the data once it is encrypted. In decryption, the receiver decodes the data once it is received.

# CHAPTER 2 LITERATURESURVEY

## RELATED WORK

One of the earliest encryption techniques is the Caesar Cipher, invented by Julius Caesar more than two thousand years ago to communicate messages to his allies. The Caesar Cipher is a great introduction to encryption, decryption, and code cracking, thanks to its simplicity. Encryption is the process by which a readable message is converted to an unreadable form to prevent unauthorized parties from reading it.

Decryption is the process of converting an encrypted message back to its original (readable) format. The original message is called the plaintext message. The encrypted message is called the cipher text message. Digital encryption algorithms work by manipulating the digital content of a plaintext message mathematically, using an encryption algorithm and a digital key to produce a cipher text version of the message. The sender and recipient can communicate securely if the sender and recipient are the only ones who know the key. As, the data exchange in electronic way is rapidly increasing, it is also equally important to protect the confidentiality of data from unauthorized access. The breaches in security affect user’s privacy and reputation.

## SYSTEM STUDY

The data exchanged can be text, image, audio, video etc. Each type of data has its own features different techniques are used to protect confidential image data from unauthorized access. Hence encryption of data is done to confirm security in open networks such as the internet where the multimedia applications are ever growing. Cryptography is the study of techniques for secure communication in the presence of an adversary. It deals with problems like encryption, authentication, and key distribution to name a few. Plain text is encrypted using an encryption algorithm and an encryption key. This generates an unreadable text which is called as cipher text (encrypted data). Decryption is the inverse of encryption; original form of data can only be viewed by decrypting encrypted data with the correct key. Text Encryption is the method by which information is converted into secret code that hides the information’s true meaning. Cipher is an algorithm which is applied to plain text to get cipher text. It is the unreadable output of an encryption algorithm. The term “cipher” is sometimes used as an alternative term for cipher text.

# CHAPTER 3 PROECT IMPLEMENTATION

## DESCRIPTION OF PROPOSED PROJECT

The proposed new algorithm has been created to encrypt the data and includes efficient and secure. The results Method of applying the proposed this algorithm works to protect data from theft and cannot be decrypted in the text. It is handled very accurately to avoid any penetration to reach the original text. It can be used in companies or any other system, but it takes a long time to encrypt it. To the original text when encryption to ensure the protection of data in full and security. Encrypted text contains a special key, even when stolen. The private key cannot be decrypted by the specialist and licensed by the creator of the code in order to protect the information in an excellent manner the final result we find new method designated for encryption and decryption as shown below. In the first field, regular text is entered hence; the general code of the first key and the special code of the second key were specified through this process, the encoded text shown.



plain text

Encryption

Cipher text



plaint text

Decryption

**Fig.1**.Block diagram of encryption and decryption

The above is the block diagram showing encryption and decryption of the text. As it is a kind of security system this methodology mostly used in social medias and in other aspects. The algorithm works best in protecting the text form the third party and helps to transfer the information between sender and receiver without any disturbances. The characters for the alphabets and numbers are implemented in code itself so whenever the encrypt message is given it is converted to cipher and gives the original text to the receiver after decryption.

* 1. **FLOW CHART**

Start

Enter the encrypt message

The plain text is converted to byte form.

Cipher Text

The cipher text is converted to plain text.

Decrypted message shown

stop

**Fig.2.**Flow chart of encryption and decryption

### Description of flow chart:

The encryption and decryption are a kind of security system. As mentioned before encryption converts the plain text to cipher text and decryption converts the cipher text to plain text. The algorithm works in such a way that when we enter the plain text the computer takes it and converts in to byte form which is nothing but cipher text. All the alphabets and numbers in the plain text are converted to byte forms or the characters in encryption. Till here the encrypted message is shown. When we give the cipher text that is characters or the byte form the decryption process converts in to the plain text that is nothing but the original text which we given. So, decryption is nothing but the reverse process of encryption. Encryption and decryption method is not only used for text bust also images. But the image processing has some complexity but the working algorithm is same for both text and images.

# CHAPTER 4 IMPLEMENTATION

## MODULES

The implementation of code is done in eclipse a java developer software. This software mostly used to the basic applications and helps the developer to give the desired output. The following is the source code

import java.awt.event.ActionEvent; import java.awt.event.ActionListener; import javax.swing.JButton;

import javax.swing.JFrame; import javax.swing.JLabel; import javax.swing.JPanel; import javax.swing.JTextPane;

public class MyFrame extends JFrame {

public MyFrame() { super("Encrypt\_Testing"); setBounds(100, 100, 600, 900);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); getContentPane().setLayout(null);

// PlainText Panel

JPanel plainPanel = new JPanel(); plainPanel.setBounds(0, 0, 584, 287); getContentPane().add(plainPanel); plainPanel.setLayout(null);

// PlainPane -> data setting -> final!

final JTextPane PlainPane = new JTextPane(); PlainPane.setBounds(12, 29, 560, 166); plainPanel.add(PlainPane);

// PlainText Label

JLabel PlaintextLabel = new JLabel("PlainText");

PlaintextLabel.setBounds(12, 10, 57, 15); plainPanel.add(PlaintextLabel);

// CipherText Panel

JPanel cipherPanel = new JPanel(); cipherPanel.setBounds(0, 284, 584, 287); getContentPane().add(cipherPanel); cipherPanel.setLayout(null);

// CipherText Label

JLabel CipherLabel = new JLabel("CipherText"); CipherLabel.setBounds(12, 10, 70, 15); cipherPanel.add(CipherLabel);

// CipherPane -> data setting -> final!

final JTextPane CipherPane = new JTextPane(); CipherPane.setBounds(12, 27, 560, 188); cipherPanel.add(CipherPane);

// Decipher Panel

JPanel decipherPanel = new JPanel(); decipherPanel.setBounds(0, 570, 584, 292); getContentPane().add(decipherPanel); decipherPanel.setLayout(null);

// Decipher Label

JLabel DecipherLabel = new JLabel("DecipherText"); DecipherLabel.setBounds(12, 10, 81, 24); decipherPanel.add(DecipherLabel);

// Decipher TextPane

final JTextPane DecipherPane = new JTextPane(); DecipherPane.setBounds(12, 36, 560, 208); decipherPanel.add(DecipherPane);

// Decryption Button

JButton DecryptedButton = new JButton("Decryption"); DecryptedButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) { String cipher, result\_de;

try {

cipher = CipherPane.getText();

result\_de = FileEncryption.decryptString(cipher); DecipherPane.setText(result\_de);

} catch (Exception e) { e.printStackTrace();

}

}

});

DecryptedButton.setBounds(448, 240, 124, 23); cipherPanel.add(DecryptedButton); setVisible(true);

// Encryption Button

JButton EncryptedButton = new JButton("Encryption"); EncryptedButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

// PlainArea -> GetText! -> Plain

// call encryptString(plain)

// result <- FileEncryption.encryptString(plain)

// CipherPane.setText(result)-> Show! String plain, result\_en;

try {

plain = PlainPane.getText();

result\_en = FileEncryption.encryptString(plain); CipherPane.setText(result\_en);

} catch (Exception e) {

// TODO Auto-generated catch block e.printStackTrace();

}

}

});

EncryptedButton.setBounds(445, 226, 127, 23); plainPanel.add(EncryptedButton);

}

public static void main(String args[]) { new MyFrame();

}

}

The above source code will import the file encryption code which gives the frame for the encryption and decryption process.

import java.io.ByteArrayInputStream; import java.io.ByteArrayOutputStream; import javax.crypto.Cipher;

import javax.crypto.CipherInputStream; import javax.crypto.CipherOutputStream; import javax.crypto.spec.IvParameterSpec; import javax.crypto.spec.SecretKeySpec;

public class FileEncryption {

// Initial Vector

public static final byte[] iv = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };

// EncryptAndDecrypt String -> Input : PlainText + Return :

// CipherText+DecipherText

public static String encryptString(String src) throws Exception { String dst = "";

// Not Input!

if (src == null || src.length() == 0) return "";

// Encryption Setting

byte[] k = "Multimediaproces".getBytes(); SecretKeySpec Key = new SecretKeySpec(k, "AES"); IvParameterSpec ivspec = new IvParameterSpec(iv);

Cipher encryptCipher = Cipher.getInstance("AES/CBC/PKCS5Padding"); encryptCipher.init(Cipher.ENCRYPT\_MODE, Key, ivspec);

ByteArrayOutputStream baos = new ByteArrayOutputStream(); CipherOutputStream cout = new CipherOutputStream(baos, encryptCipher); cout.write(src.getBytes());

cout.flush(); // ByteOutputStream -> Write Encryption Text cout.close();

dst = new String(baos.toByteArray()); return dst;

}

// String src -> EncryptedData

public static String decryptString(String src) throws Exception {

// src value is Encrypted Value!

// So, src value -> Not Byte! String dst = "";

byte[] encryptedBytes = src.getBytes();

// Not Input!

if (src == null || src.length() == 0) return "";

// Decryption Setting

IvParameterSpec ivspec = new IvParameterSpec(iv); byte[] k = "Multimediaproces".getBytes(); SecretKeySpec Key = new SecretKeySpec(k, "AES");

Cipher decryptCipher = Cipher.getInstance("AES/CBC/PKCS5Padding"); decryptCipher.init(Cipher.DECRYPT\_MODE, Key, ivspec);

ByteArrayOutputStream baos = new ByteArrayOutputStream(); ByteArrayInputStream bais = new ByteArrayInputStream(encryptedBytes); CipherInputStream cin = new CipherInputStream(bais, decryptCipher); byte[] buf = new byte[1024];

int read;

while ((read = cin.read(buf)) >= 0) // reading encrypted data!

{

baos.write(buf, 0, read); // writing decrypted data!

}

// closing streams cin.close();

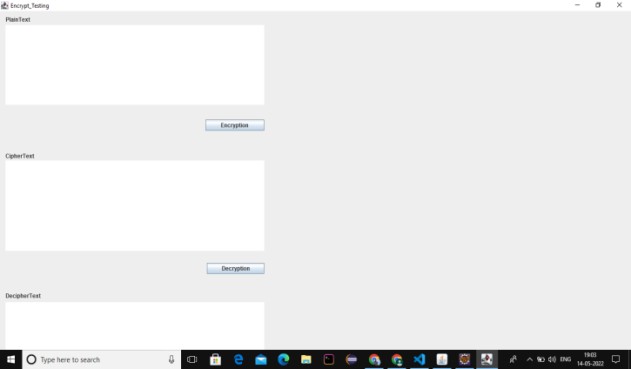
dst = new String(baos.toByteArray()); return dst;

}

}

## OVERVIEW TECHNOLOGY

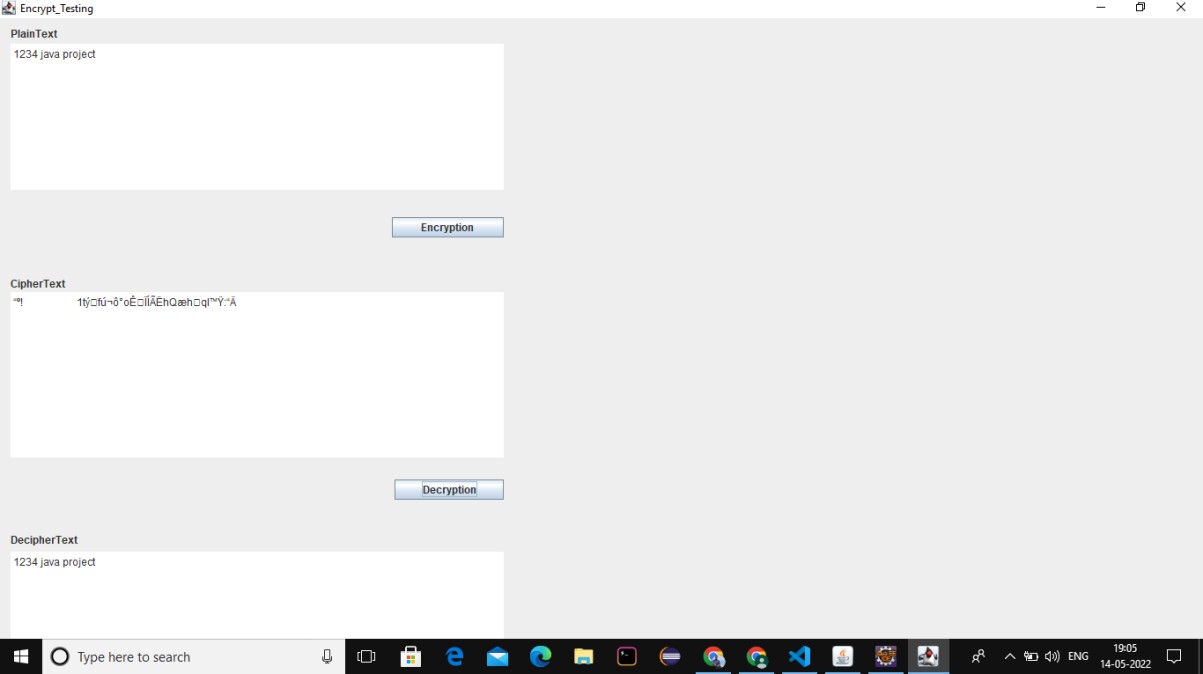
The implementation of the project is done using awt i.e., Abstract Window Toolkit is an API to develop graphical user interface or windows-based application. The java.awt packages provide classes for AWT api such as textfile, label, textarea, button, checkbox, choice etc. The events also used in the code. The easiest control to use is a label. A label contains a string and is an object of type label. Labels are passive controls the do not support any interaction with the user. Whereas a button is a component that contains a label and that generates an event when it is pressed.

The getBytes command is used to convert the alphabets in to character or to byte form to hide the plain text which is nothing but cipher text. The cipher text is then converted to plain textin the process of decryption. The array input stream and output stream are used to store the byte values. The text component class is the super class is the super class of any component that permits the editing of some text. A text component embodies a string of text. Whereas the text field component will allow the user to enter some text. It is used to implement a single line text entry area usually called an edit control. The overall output will be a form of frame.

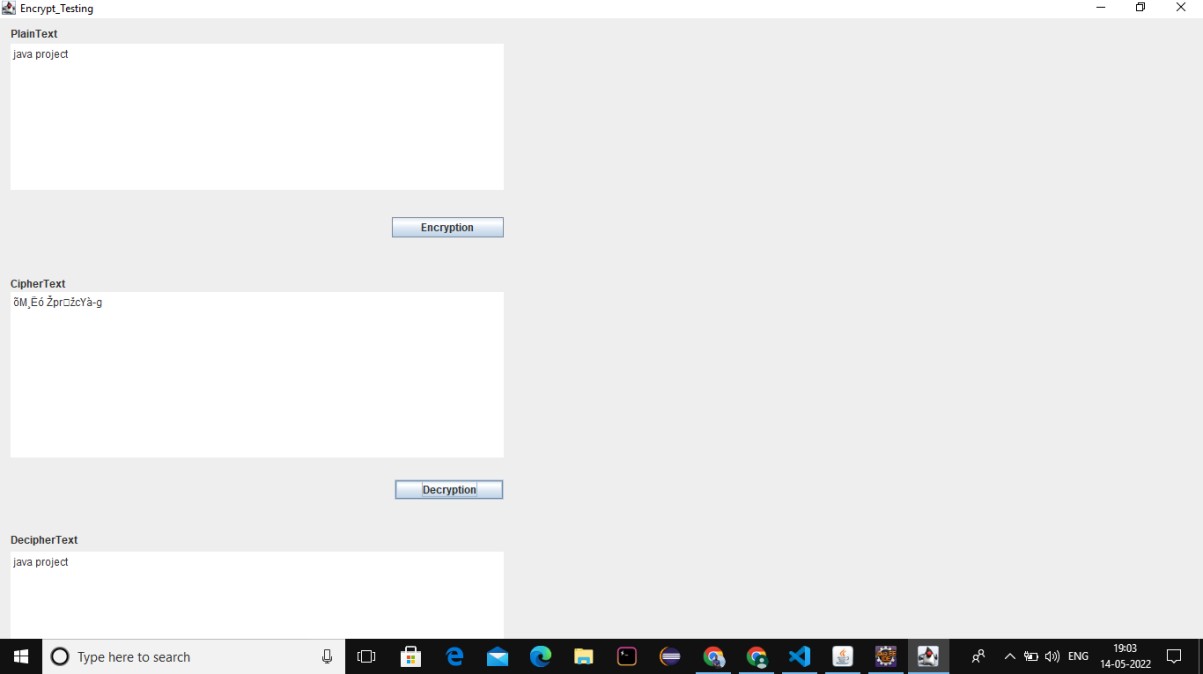
**Fig.3**.Frame of output

# CHAPTER 5 RESULT ANALYSIS

## 5.1 TEST RESULTS



**Fig.4.**Testcase1



**Fig.5.**Testcase

# CHAPTER 6 CONCLUSION AND FUTURE SCOPE

## CONCLUSION

The purpose of this section is to present how encryption and decryption can be used to implement security in the Web. It starts with a list of challenges for protecting information, continues with the presentation of the basic encryption decryption algorithms and protocols, presents the Secure Sockets Layer protocol, and concludes with an example of how cryptography is used in a commercial transaction on the Internet. From a technical point of view, cryptography is the solution to many of the security challenges that are present in the Internet. The technology exists to solve most of the problems. However, there are several issues that have obstructed the widespread use of cryptography in the Internet.

## FUTURE SCOPE

We’re already beginning to witness the foundations being laid for facial encryption. As facial recognition technology advances, we expect to see facial encryption become a fundamental way of securing data and protecting access to confidential information. Also, the encryption and decryption can be used in billing system. It would be move advance if we this methodology in billing system. Encryption and decryption can also be used for image transferring. Officially this methodology can be used even in military base in an advanced way. There is a scope using encryption decryption in many applications in the future.

# BIBLIOGRAPHY

1. P. Zimmerman, "An Introduction to Cryptography", Doubleday & Company, Inc., United State of America, USA, 1999.
2. C. Shannon, “Communication Theory of Secrecy Systems”, Bell Systems Technical Journal, MD Computing, vol. 15, pp. 57-64, 1998.
3. I. Nichols, K. Randall (1999), ICSA Guide to Cryptography, McGraw-Hill, Companies Inc, New York.
4. H. Mohan, and R. Raji. "Performance Analysis of AES and MARS Encryption Algorithms". International Journal of Computer Science Issues (IJCSI), Vol. 8, issue 4. 2011.
5. A. Lee, NIST Special Publication 800-21, Guideline for Implementing Cryptography in the Federal Government, National Institute of Standards and Technology, November 1999.
6. J. Nechvatal, Report on the Development of the Advanced Encryption Standard (AES), National Institute of Standards and Technology, October 2, 2000.
7. M. Wali and M. Rehan, “Effective Coding and Performance Evaluation of the Rijndael Algorithm (AES)”, in the Proceedings of the Engineering Sciences and Technology Conference, vol. 7, pp. 1-7, Karachi, 2005.
8. C. Jie, “Design Alternatives and Implementation of PKI Functionality for VoIP”, Master of Science dissertation, Telecommunication Systems Laboratory, Royal Institute of Technology (KTH), Stockholm, 2006.
9. R. Hunt, “PKI and Digital Certification Infrastructure”, in the Proceedings Ninth IEEE International Conference on Networks, vol. 4, pp. 234 – 239, Bangkok, Thailand, 2001.
10. S. Xenitellis, The Open–Source PKI Book: A Guide to PKIs and Open-Source Implementations, Open CA Team, 2000.
11. W. Emm, “Impact of Multiencryption in Data Security”, International Journal of Computer Theory and Engineering, vol. 1, pp. 571-567, 2009.
12. B.D.C.N.Prasad, P E S N Krishna Prasad, “A Performance Study on AES algorithms”, International Journal of Computer Science and Information Security, Vol. 8, No. 6,September 2010,pp 128-132.
13. S. Tao, W. Ruli, and Y. Yixun, “Clock-Controlled Chaotic Key-Stream Generators”, Institution of Engineering and Technology Electronics Letters, vol. 34, pp. 1932-1934, 1998.
14. A. Masoun,” Cryptography Primitives Based on Piecewise Nonlinear Chaotic Maps”, Master of Science dissertation, UniversitiSains Malaysia (USM), Pineng, Malaysia, 2008.
15. Obaida. Al-Hazaimeh, " Increase the Security Level for Real-Time Application Using New Key Management Solution", International Journal of Computer Science Issues (IJCSI), Vol. 9, Issue 3, 2012.
16. L. Chang-Doo, C. Bong-Jun, P. Kyoo- Seok (2004), “Design and evaluation of a block encryption Algorithm using dynamic-key mechanism”, Future Generation Computer Systems 20, 327–338.