A

Project Report

On

Image Steganography embedded with Advance Encryption Standard (AES) securing with SHA-256

By

Yash Kumar Shukla

DEPARTMENT OF COMPUTER APPLICATIONS

JSS ACADEMY OF TECHNICAL EDUCATION, NOIDA

May’2020

A
Project Report
On

Image Steganography embedded with Advance
Encryption Standard (AES) securing with SHA-256

In partial fulfillment of requirements for the degree of

Master of Computer Applications

SUBMITTED BY:

Yash Kumar Shukla

Under the Guidance of

DR. Shivani Dubey

DEPARTMENT OF COMPUTER APPLICATIONS
JSS Academy of Technical Education, Noida

May’2020

CERTIFICATE

This is to certify that the project entitled **“Image Steganography embedded with Advance Encryption Standard (AES) securing with SHA-256”** has been carried out by Shivani Jain under my guidance in partial fulfillment of the degree of **Master of Computer Applications** of **APJ Abdul Kalam Technical University, Lucknow** during the period Jan,2020 - May,2020.

Name of the Student**:**

Yash Kumar Shukla

1809114910

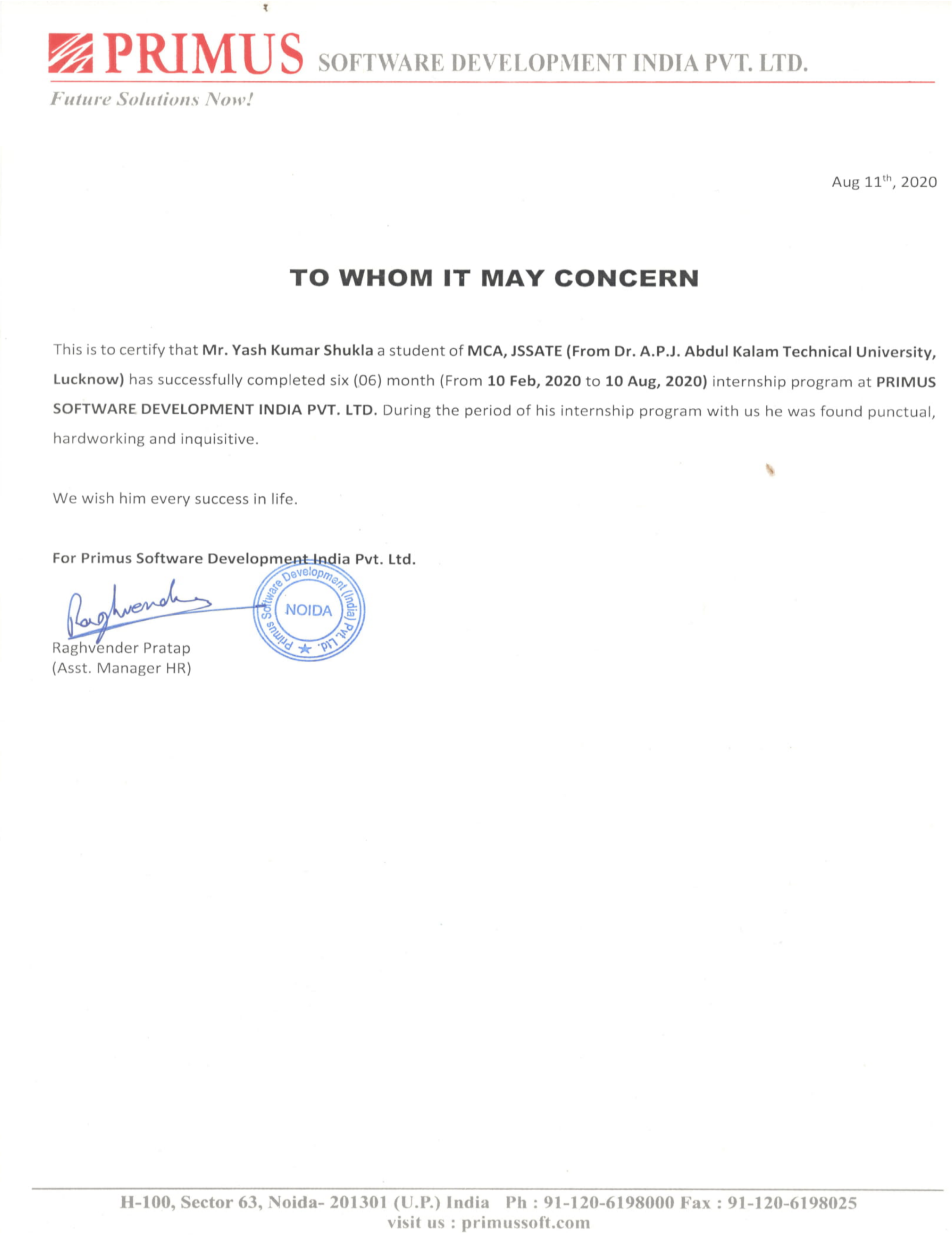
Date:

Place: Noida, U.P

Internal Guide

Ms. Shivani Dubey

COMPANY CERTIFICATE



PRIMUS SOFTWARE DEVELOPMENT INDIA PVT. LTD.

Future Solutions Now!

Aug 11th, 2020

TO WHOM IT MAY CONCERN

This is to certify that **Mr. Yash Kumar Shukla** a student of **MCA, JSSATE (From Dr. A.P.J. Abdul Kalam Technical University, Lucknow)** has successfully completed six (06) month (From **10 Feb, 2020** to **10 Aug, 2020)** internship program at **PRIMUS SOFTWARE DEVELOPMENT INDIA PVT. LTD.** During the period of his internship program with us he was found punctual, hardworking and inquisitive.

We wish him every success in life.

For Primus Software Development India Pvt. Ltd.

Primus Software Development (India) Pvt. Ltd.
NOIDA

Raghvender Pratap
(Asst. Manager HR)

H-100, Sector 63, Noida- 201301 (U.P.) India Ph : 91-120-6198000 Fax : 91-120-6198025
visit us : primussoft.com

ACKNOWLEDGMENT

I owe my deep gratitude to our project guide **DR. Shivani Dubey** & **Prof. Vikas Singhal** who took keen interest on our project work "**Image Steganography embedded with Advance Encryption Standard (AES) securing with SHA-256**" and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

We are also grateful to **Mr. Vikas Singhal** for his valuable help and guidance and to our respected principal **Prof. Gurulingappa M. Patil**, JSSATE NOIDA for permitting us to utilize all the necessary facilities of the institution and bring this project to existence.

Lastly, we would like to express our deep appreciation towards our friends and our to our parents for providing us the moral support and encouragement.

Yash Kumar Shukla

1809114910

PAGE INDEX

Contents

ABSTRACT

The proposed project "**Image Steganography embedded with Advance Encryption Standard (AES) securing with SHA-256"** works upon the idea of securing the classified information. This is achieved by using steganography which is an approach to hide classified information into some other file while maintaining its visual aids and secondly is cryptography which works upon textual data and transform it in a way that no one can comprehend it. The proposed method secures the weaker section which is the key in Advance Encryption Standard using hashing technique. The proposed work enhances the level of concealment of information from unauthorized access and for covert information exchange by encrypting the data and hiding it into a multimedia file known as image. The Secure Hash Algorithm 256 generates a hash key of 256 bits which is an unbreakable hashing technique after that the key is used in the process of encrypting the text with Advance Encryption Standard 256 which is an unbreakable encryption technique till this time and a cipher text is obtained. The cipher text is embedded into a target image using Least Significant Bit method which make changes in image that cannot be understand by naked eyes. The change in byte is 0.000002%. It ensures the visual quality of an image remains intact. The distortion or change in the image remains intermittent to human eyes. The major issue concerned for the government and security agencies such as were to exchange highly classified information in a secure and undetectable manner and abide the notion of hacker to comprehend any such information. So, this project secure the classified information in an undetectable manner and can be protected from the hackers also.

CHAPTER 1

1. INTRODUCTION

Steganography is a methodology of covert classified data within an ordinary or non-secret, file or message in order to avoid detection; the secret data is then extracted at its destination. The steganography can be blended with encryption as a step for shielding of data.

Sender
Intruder will not get to know of the existence of secret message
Receiver

Steganography extracted from the Greek term steganos, meaning covered or secret, and graphy (writing or drawing). In simple words, steganography is covert writing, whether it consists of invisible ink on paper or information hidden in a file like audio, image,etc. Cryptography scrabbles a message into a code to nullify its meaning, steganography hides the message entirely. These two covert communication technologies can be used individually or together—for example, by first encrypting a message, then hiding it in another file for transmission. steganography's role is gaining prominence. Steganography basically does is to exploit human behaviour, human senses are not trained to peep into files that have information hidden inside of them, although there are programs available that can do what is called Steganalysis (Detecting use of Steganography.) Steganography hides the secrete message within the host data set and presence imperceptible and is to be reliably communicated to a receiver. The host data set is purposely corrupted, but in a covert way, designed to be invisible to information analysis. It is one of the methods employed to guard secret or classified data from malicious attacks.

Cryptography and steganography are techniques used to hide and protect secret data. However, cryptography makes the data unreadable, or hides the meaning of the data, while steganography hides the existence of the data.

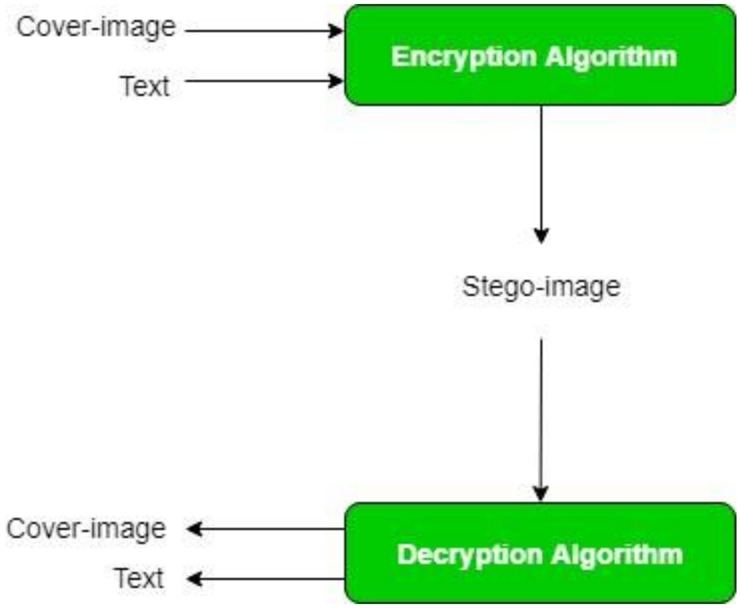
In layman's terms, cryptography is similar to writing a letter in a secret language: people can read it but won't understand what it means. However, the existence of a (probably secret) message would be obvious to anyone who sees the letter, and if someone either knows or figures out your secret language, then your message can easily be read.

If you were to use steganography in the same situation, you would hide the letter inside a pair of socks that you would be gifting the intended recipient of the letter. To those who don't know about the message, it would look like there was nothing more to your gift than the socks. But the intended recipient knows what to look for, and finds the message hidden in them.

Similarly, if sender and receiver exchange media files over the internet, it would be more difficult to determine whether these files contain hidden messages than if they were communicating using cryptography.

Cryptography is often used to supplement the security offered by steganography. Cryptography algorithms are used to encrypt secret data before embedding it into cover files.

It is viable to alter graphic and sound files without losing their overall viability for the viewer and listener. With audio, one can use bits of the file that contain sound which is not audible to the human ear. With graphic images, one can remove redundant bits of colour from the image and still produce a picture that looks intact to the human eye and is difficult to discern from its original. These are those bits that stego conceal its data. A stego program uses an algorithm, to embed data in a file that could be an image, audio, etc , and a password scheme to allow you to retrieve information. In image steganography, covertness is achieved by embedding data into a cover image and publishing a stego-image. There are different types of steganography techniques each has its strengths and weaknesses. In this paper, we review the different security and data hiding techniques that are used to implement steganography such as LSB, ISB, MLSB etc.



In this project, the algorithm that will be used is LSB (Least Significant Bit). A digital image is described using a 2-D matrix of the colour intestines at each grid point (i.e. pixel). Typically, grey images use 8 bits, whereas colored utilizes 24 bits to describe the color model, such as the RGB model. The Steganography system which uses a file as the cover, there are several techniques to conceal information inside cover-image. The techniques manipulate the cover-image pixel bit values to embed the secret information. The last bits are altered directly to the cover image pixel bytes. Consequently, the techniques are simple and easy to implement.

The **Least Significant Bit (LSB)** is one of the main techniques in image Steganography. The concept of LSB Embedding is simple. It works upon the level of precision in different image formats is far greater than that perceivable by average human vision. Therefore, a stego image with slight variations in its colour intensity will be indistinguishable from the original by a human being, just by looking at it.

For making our data more secure we will be using a cryptographic algorithm because if the intruders uses **Steganalysis (Detecting use of Steganography)** and they gets to know that there is a data present inside the image then also the intruder can’t get his hand to the classified data as it will be encrypted by the most secure cryptographic algorithm know as Advance Encryption Standard (AES-256).

The **Advance Encryption Standard (AES)** is based on substitution-permutation network. AES is a sub-set of Rijndael. It is a family of ciphers with different key and block sizes. In AES, the block size is 128 bits or 16 characters which means 16 characters can be encrypted at a time. It has three different key size variation: 128 bits, 192 bits and 256 bits.

When hackers want to access a system, they will aim for the weakest point, which is not the encryption, but the key. AES-256 is most secure encryption technique and till now there is no report of its cracking. But the key is the weakest point in order to secure that SHA-256 (Secure Hash Algorithm) will be used that will give a hash value. Hash is “not encryption” because it can never be decrypted back to the initial text, it a one-way street. This makes the system robust and patches the weakest link.

The Secure Hash Algorithm (SHA-256) is a part of the family. The cryptographic hash functions was produced by the “National Institute of Standards and Technology” as a **U.S. Federal Information Processing Standard**. SHA-256 is patented in US patents 6829355. Hash is not “encryption” because it can never be decrypted back to the initial text. It is of fixed length for any varying length of the source text. SHA-256 is achieved by the following steps like padding, append length, divide the input into 512-bit blocks, add chaining variable, process block respectively. SHA-256 generates a unique 256-bit or we can say 32-byte of hash code. It does 64 rounds in order to reach to the final hash value.

Any text length will also be transformed into a digest size of 256 bits (32 bytes). 16.80 cycles per byte on 64-bit processor x86 architecture. It has 64 iterations in one cycle. Which makes it more safe and secure.

1.1 Purpose

The goal of steganography is covert communication. So, an elementary requirement of this steganography system is that the covert message carried by stego-media should not be sensible to human beings. The secondary target of steganography is to avoid drawing suspicion to the existence of a hidden message. This approach of information hiding technique has recently become important in several application areas This project has the following objectives:

- To provide security tool based on image steganography techniques.

- To explore techniques of hiding data using hide module of this project
- To extract techniques of getting secret data using the retrieve module.

Steganography comes into play where encryption is not permitted. Or, more precisely, steganography is used to addendum encryption. An encrypted file can be concealed using steganography, so even if the encrypted file is deciphered, the hidden message is not seen or vice versa is also possible. Steganography hides the data inside the image no one able to notice that this image will be containing some crucial information but even if someone detects that the image contains the data inside it, the data will be protected with the encryption. The image will only be having the cipher text embedded with it. For decrypting that cipher text the user will be needing the key without it, it is impossible to get the information.

1.2 Scope

The system hides the classified information into an image in order to secure the data and make communication of data undetectable and enhancing its security by using Advance Encryption Standard (AES-256) and even securing the weak link that is the key with SHA-256. It will make the system impenetrable would having rightful access. This system could help the Government agencies and National Security agencies in order to secure the classified information.

1.3 Definition, Acronym and Abbreviation

The project is the digital information security system used for securing the classified text. The modules such a hide, retrieve and key creation module depicts the functionality of the application. The Earlier form of linguistic or language of hidden writing. Then later in series, such as invisible ink are used to hide messages physically. One drawback of linguistic steganography is that users must have skilled themselves to have a good knowledge of linguistry. In recent years, everything is trending toward digitization. And with the development of internet technology, digital media can be transmitted conveniently over the network. Therefore, messages can be secretly carried by digital media by using steganography techniques and then be transmitted through the internet rapidly.

Steganography is a technique of hiding the fact that communication is taking place, by hiding information in other information. Many different cover file formats can be used, but digital images are trending because of their frequency on the internet. For concealing classified data into images, there are various steganography techniques some are more complex than others and all of them have respective strong and weak points. So, we create this application, to make the data hiding simpler and user friendly.

Least Significant Bit (LSB) is a steganography technique embeds the data into the cover image or target image. This technique is implemented by making changes in the least significant bit (LSB) of a byte. LSB provides the least change in the image which is far from persistent of a human naked eye. The change in a byte is 0.000002% which is almost negligible.

Secure Hash Algorithm (SHA-256) is a part of the family. The family of cryptographic hash functions which was declared by the National Institute of Standards and Technology as a U.S. Federal Information Processing Standard. SHA-256 is patented in US patents 6829355. Hash is “not encryption” because it can never be decrypted back to the initial text. It is of fixed length for any varying length of the source text. SHA-256 is achieved by the following steps like padding, append length, divide the input into 512bit blocks, add chaining variable, process block respectively. SHA-256 generates a unique 256-bit or we can say 32-byte of hash code. It does 64 rounds in order to reach to the final hash value.

Advance Encryption Standard (AES-256) is a symmetric key encryption technique, which uses only one secret key to cipher(encrypt) and decipher(decrypt) data. AES-256 uses a key length of 256 bits, it uses the largest bit length and is practically adamantine by brute force attack on the basis of current computing power, which makes it the strongest encryption standard. The AES encryption method first performs substitution of data using a substitution table; which is followed by shifting of data rows, then after that mixing of the columns takes place, and finally the last transformation with a simple exclusive (XOR) operation performed on each column using a different part of the encryption key.

CHAPTER 2

2. THE OVERALL DESCRIPTION

The overall description of project can be depicted as a secure system designed to guard the classified information, primarily without getting noticed and enabling shields that could protect the data from unauthorized access. The embedding of data into a file and extracting data from a file is the major task prosecuting the security levels and shields in order to protect classified information.

2.1 Product perspective: -

The product provides mainly three modules hide module, retrieve module and key Creation module with the facility easily covering of the classified data into a simple image. The problem that the project resolves is of hiding sensitive data that should not be perceived by others. The transferring of data over network that is private and sensitive needs to be protected. The data may contain sensitive information that should not be altered and read by others. Sharing secret information without being noticed is an issue in the current scenario. For this issue, the project is designed to hide the textual classified data into an image without intervening its visual aid and providing unbreakable security.

Python

Python is a growing language of our digital lifestyle. Python is a general-purpose programming language that is class-based, object-oriented, and designed to have as few implementation dependencies as possible. It follows the modularity approach write once, run anywhere. It provides extremely fast processing and dynamically typed language. It is used in growing filed like Machine Learning & AI, Web Development, Data Science & Data Analysis.

2.1.1 User interface: -

Front-end: Command Prompt or Terminal
Back-end: Python, Python Libraries.

2.1.2 Hardware interface: -

Processor: Intel Dual Core
RAM: 2GB OR Above.

HDD: 80 GB

2.1.3 Software interface: -

Operating System: Windows.

Language: Python 3.7

Tool: Pycharm

2.1.4 Memory constraints: -

Hard-disk Space: 60 GB Minimum.
RAM: 2 GB minimum.
Processor: Intel, AMD (Min DUAL CORE).

2.2 Product Functions: -

The steganographic system performs some major functions to accomplish the required tasks.

These functions constitute a basis for the whole system. These functions can be stated as:

2.2.1 Key Creation

Key Creation method generates a key with is of 256 bits. A user gives a password which is a string of varying length, which gets passed under the SHA-256 an cryptographic hash function and it generates a inversible hashed 256 bits key which can never be decrypted or its the original form cannot be perceived by any manner.

2.2.2 Hide

Hide method embeds the classified information into an image via LSB (Least Significant Bit) which is a steganographic technique. The user inputs the simple textual data, it gets automatically converted into a cipher text using a most secure encryption AES-256 and an cover image is selected by the user in order to hide the data in it while ensuring its visual aids.

2.2.3 Retrieve

Retrieve method is used to extract the classified information from a file (Stego-image) which is not being visual from human eyes and stored in a encrypted format. This method extract the data from in an encrypted format then by entering a right key it gets decrypted into an readable format. And at last shown to the user.

2.3 User Characteristics: -

All the end users can use the product, all they need to do is setup the python environment and install the dependencies related to it and simply run the script. They can perform both the tasks of generating a stego image as well as retrieving classified information from the stego image provided they enter correct security key.

2.4 Constraints: -

2.4.1 Regularity policies: -

They should always provide an correct password else the user can never access the classified information. And the size of the stego image should always be greater than size of the textual information.

2.4.2 Hardware limitations: -

There is no limitation in the operating system in which application will work. However, it will be requiring the python libraries or dependencies and python environment together. Users can access the application without any internet connection as it is a system application.

CHAPTER 3

3. SPECIFIC REQUIREMENTS

3.1 External Interfaces:-

3.1.1 User Interface:-

- Front-end software: Pycharm IDE
- Back-end software: Python, Python Libraries.

3.1.2 Hardware interface: -

- Windows.

3.1.3 Software interface: -

- Windows or Linux (Any version)
- Pycryptodome (version 3.9.7)
- Stegano (version 0.9.8)

3.2 Software product features: -

- **Edit, build, and debug with ease:**

PyCharm IDE features a lightning-fast source code editor, perfect for day-to-day use. It supports many programming languages, PyCharm IDE helps you be instantly productive with syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more. It contains keyboard shortcuts, which ease customization and keyboard shortcut mappings let you navigate your code with ease.

3.2.1 Type of Development model: -

The model applicable to this project is iterative model. The iterative model is a implementation of a software development life cycle (SDLC) that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. Iterative model has come to be a generally accepted evolution over the traditional waterfall of the past, it turns out that iterative methods were used in projects as early as the 1950s. During this period, the United States Air Force and NASA worked together to develop the X-15 hypersonic aircraft, using a largely iterative design process throughout. While this was not directly a software development project, the success of this process led NASA to use an iterative model for the software development of Project Mercury, which was the first manned spaceflight for the United States. Iterative model is best for a cyclical process. After an initial planning phase, and some of the stages are repeated over and over again, with each completion of the cycle incrementally improving and iterating on the software. Enhancements can quickly be made and implemented throughout each iteration, allowing the next iteration to be at least marginally better than the last.

Process-

-Planning and requirements

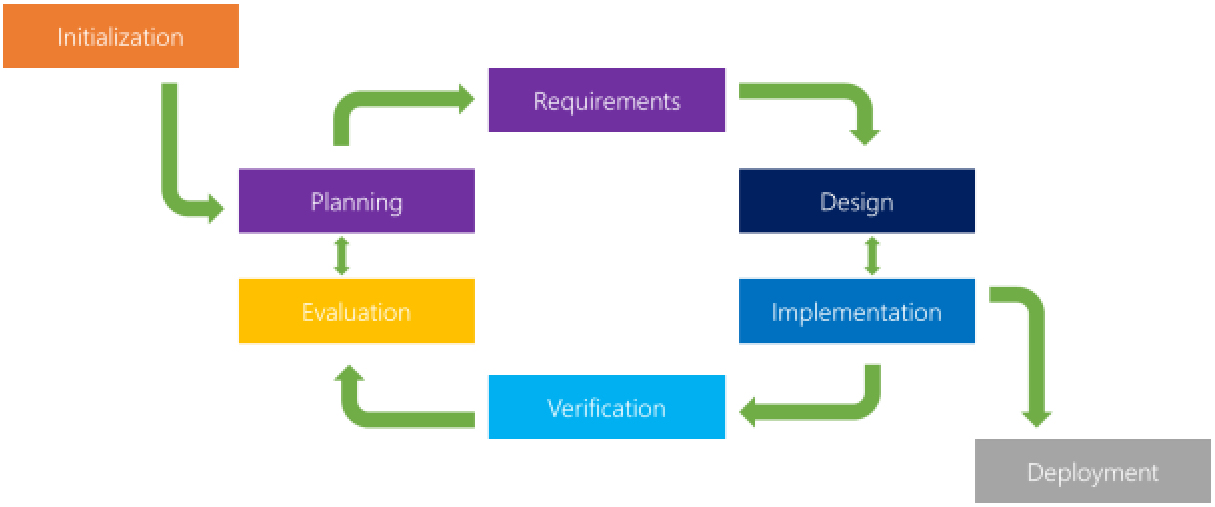
-Analysis and design

-Implementation

-Testing

-Evaluation

The crux of the entire iterative model, whereby the most recently built iteration of the software, as well as all feedback from the evaluation process, is brought back to the planning and development stage at the top of the list, and the process repeats itself all over again.



Why iterative model?

Most software development life cycles will include different type of versioning, indicating the release stage of the software at any particular stage. However, the iterative model makes this even easier by ensuring that newer iterations are incrementally improved versions of previous iterations. A previous iteration can quickly and easily be implemented or "rolled back," with minimal losses. The iterative model really starts to shine when it's in the hands of a smaller, more agile team. Another upper hand of the iterative model is the ability to rapidly adapt to the ever-changing needs of both the project or the whims of the client.

3.3 Software System Attributes: -

Feasibility study generally determines the need and solutions considered to accomplish the requirements are practically implementable in the software or not, information such as availability of the resource, estimation of cost for the development of the project and the cost which would be incurred on maintenance of the project is carried out in feasibility study.

There are different types of feasibility:

- Technical Feasibility

- Operational Feasibility
- Economic Feasibility.

3.3.1 Technical Feasibility: -

Project is technical feasible due to following reasons:

- This site is technical feasible because in this application, technology which is used to develop the site is efficient and is easily upgraded time to time and separated module makes it easy to implement and maintenance.
- Technical feasibility give assurance in respect to accuracy, reliability, ease of access and the data security.

3.3.2 Economic Feasibility: -

Project is economical feasible due to following reasons:

- The system is economically feasible and based on all freely licensed software. It does not require any additional software or hardware installation. There is nominal expenditure and economically feasible certainly.

3.3.3 Operational Feasibility: -

- This application is operational feasible because in this all users can easily operate access the facilities and module meant for according to the type of user.
- The well-planned architecture assures the optimal utilization of the resources and will be fully secure from threats. Thus, provides easy access to all the users with their registered mail Id and password.

3.3.3.1 Reliability: -

Reliability defines software to work without any failure for a given period of time. Reliability decreases because of bugs in the code, hardware failures, or problems with other system components. In order to estimate software reliability, you can count the percentage of operations that are completed correctly or track the average period of time the system runs before failing.
As Digital Security Model is a windows-based service provided to the user it is exceptionally reliable.

3.3.3.2 Availability: -

Availability is gauged by the period of time that the system's functionality and services are available for use with all operations. So, scheduled maintenance periods directly influence this parameter. And it's important to define how the impact of maintenance can be minimized. When drafting the availability requirements, the team has to state the most critical components of the system that must be available at all time in all critical states. The availability of this software depends on the availability of the system having hardware requirements.

3.3.3.3 Security: -

Digital Security Model is highly secured application. It uses Authentication functionality which make it more secure at both database level. The database server should be protected from hacking, virus etc.

3.3.3.4 Maintainability: -

The Digital Security Model is very easy to maintain new modules can be added easily without harming the functionality of other modules.

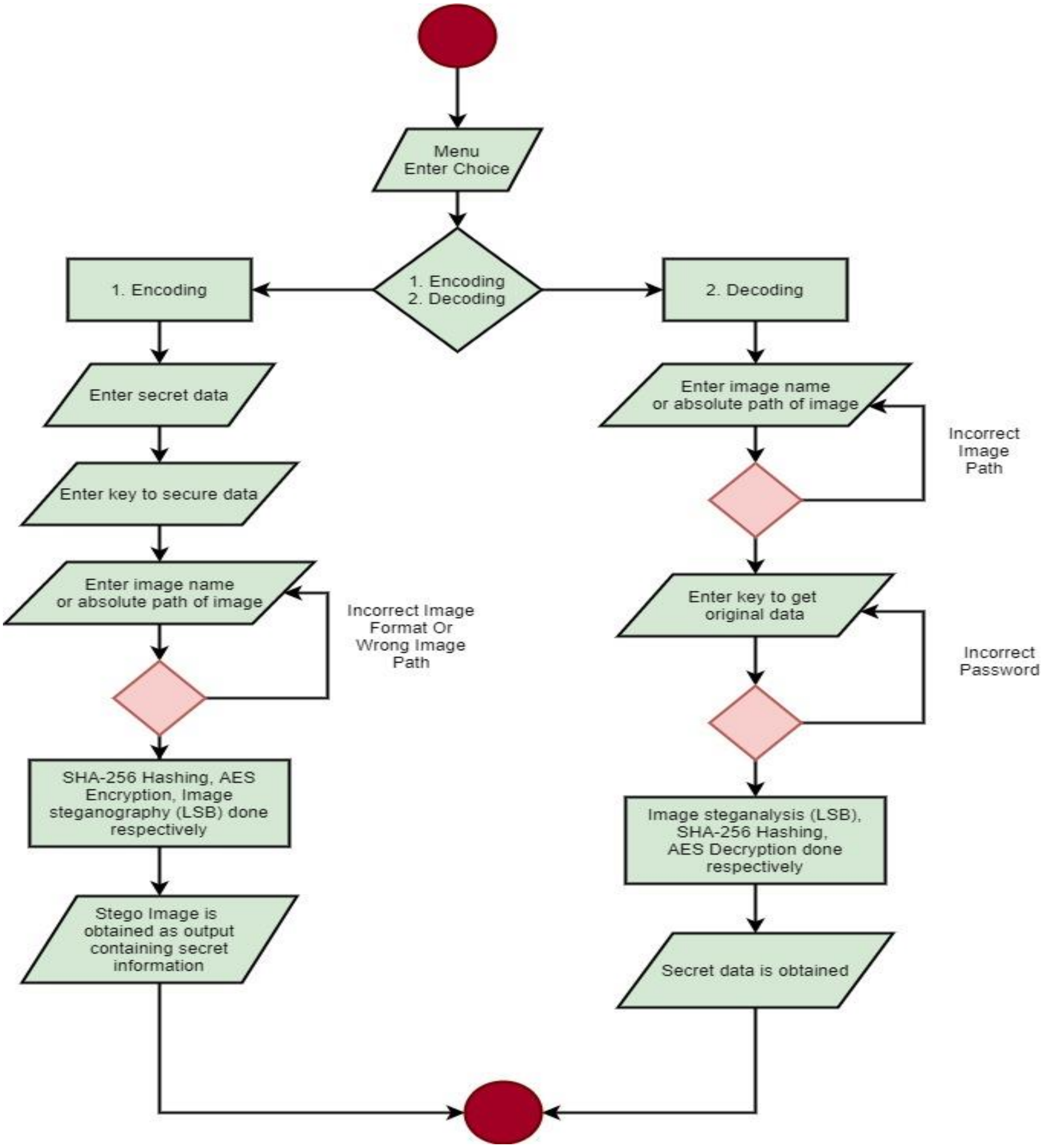
3.3.3.5 Portability: -

Portability requirements describe how the system must grow without negative influence on its performance. It means that serving more users, processing more data, and doing more transactions. Scalability has both hardware and software implications. For instance, you can increase scalability by adding memory, servers, or disk space.

- This application is very easy to be installed in the system.
- The application is developed using standard open-source software like Python, PyCharm IDE. This software will work upon all system operating system. Hence, portability problems will not arise.

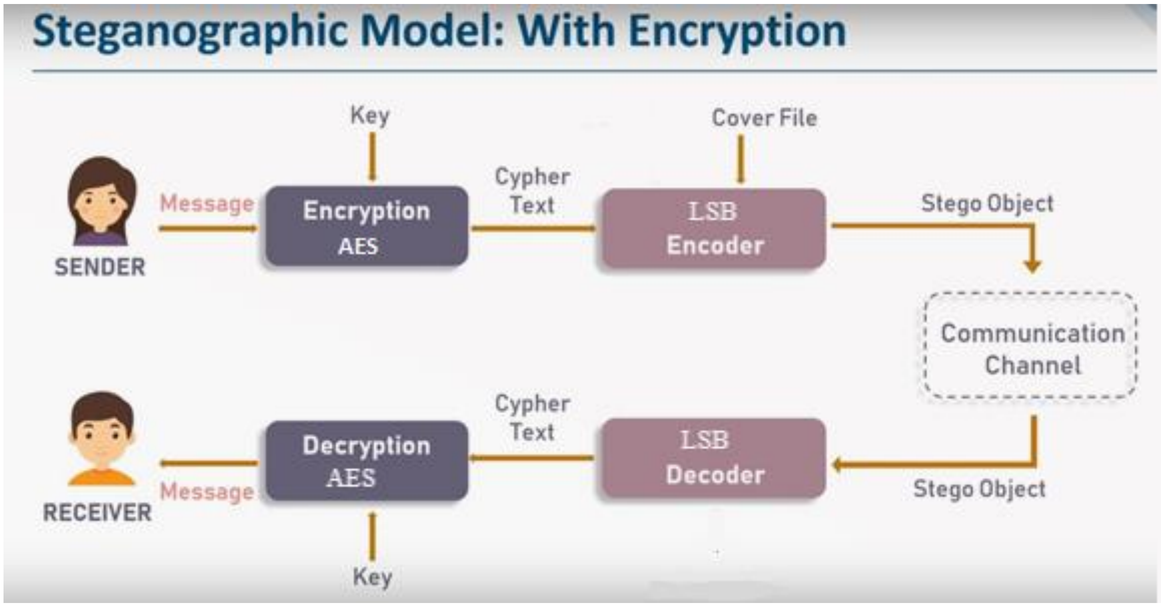
3.4 List of diagrams: -

3.4.1 Activity diagram: -



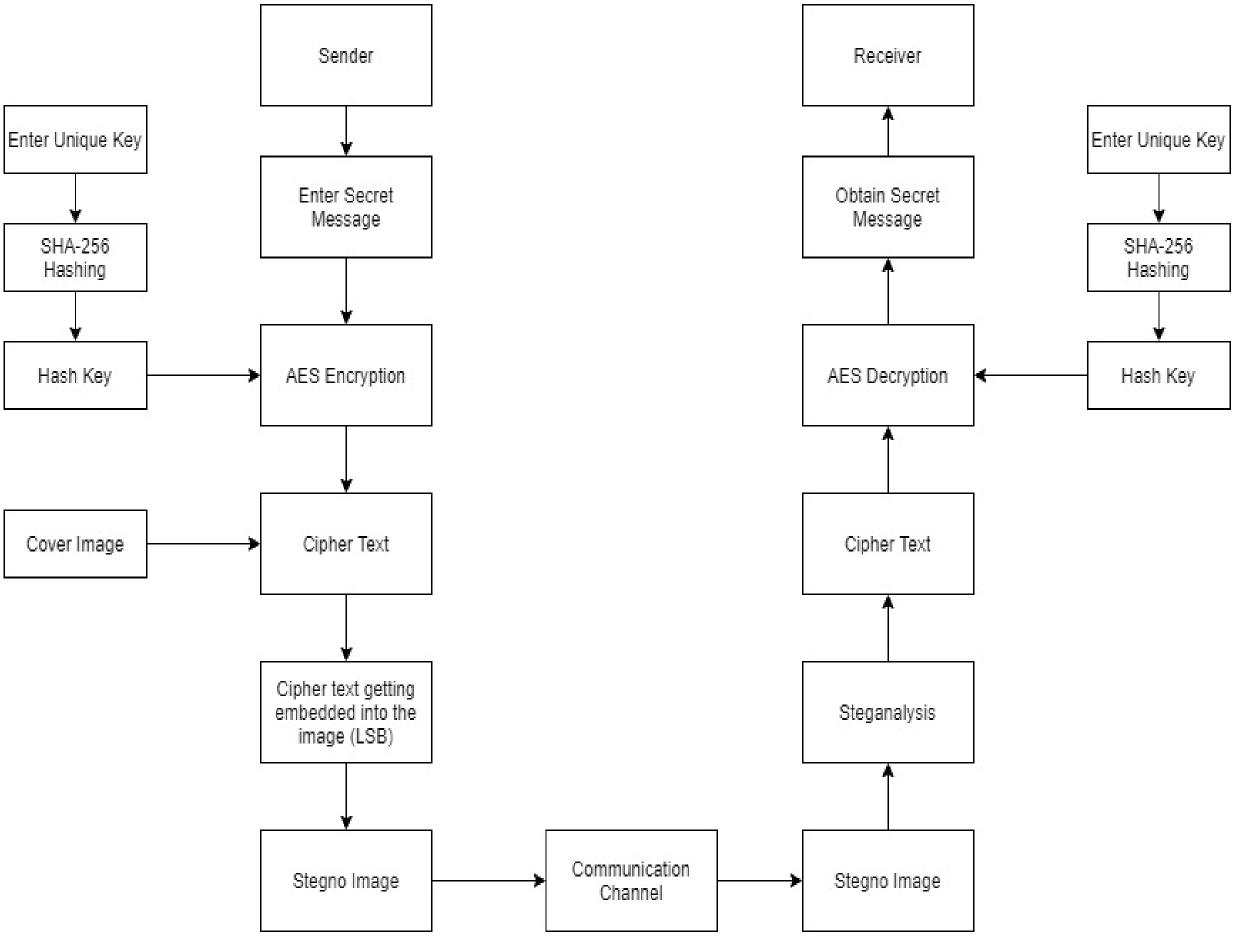
3.4.2 Data Flow diagram: -

It stands for Data flow diagram, it is a way to representing data diagrammatically and the data objects of the system. Basically, DFD is a way to show how the data is processed in the system, it shows how data moves at different stages in the system. DFD depicts information flow diagrammatically & the transformations that are applied as data moves from input to output.



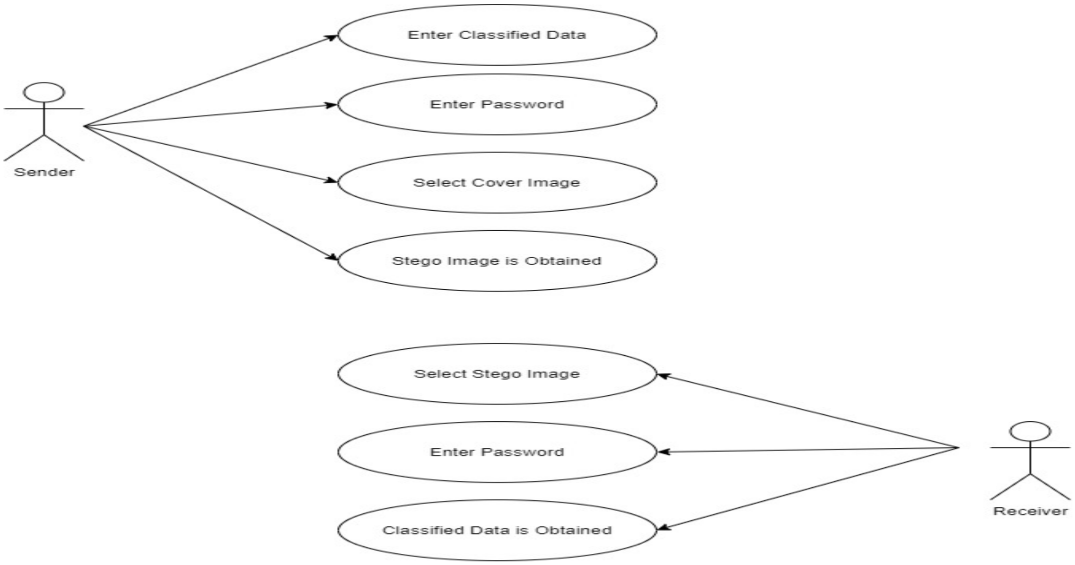
Steganographic Model: With Encryption

0 Level DFD

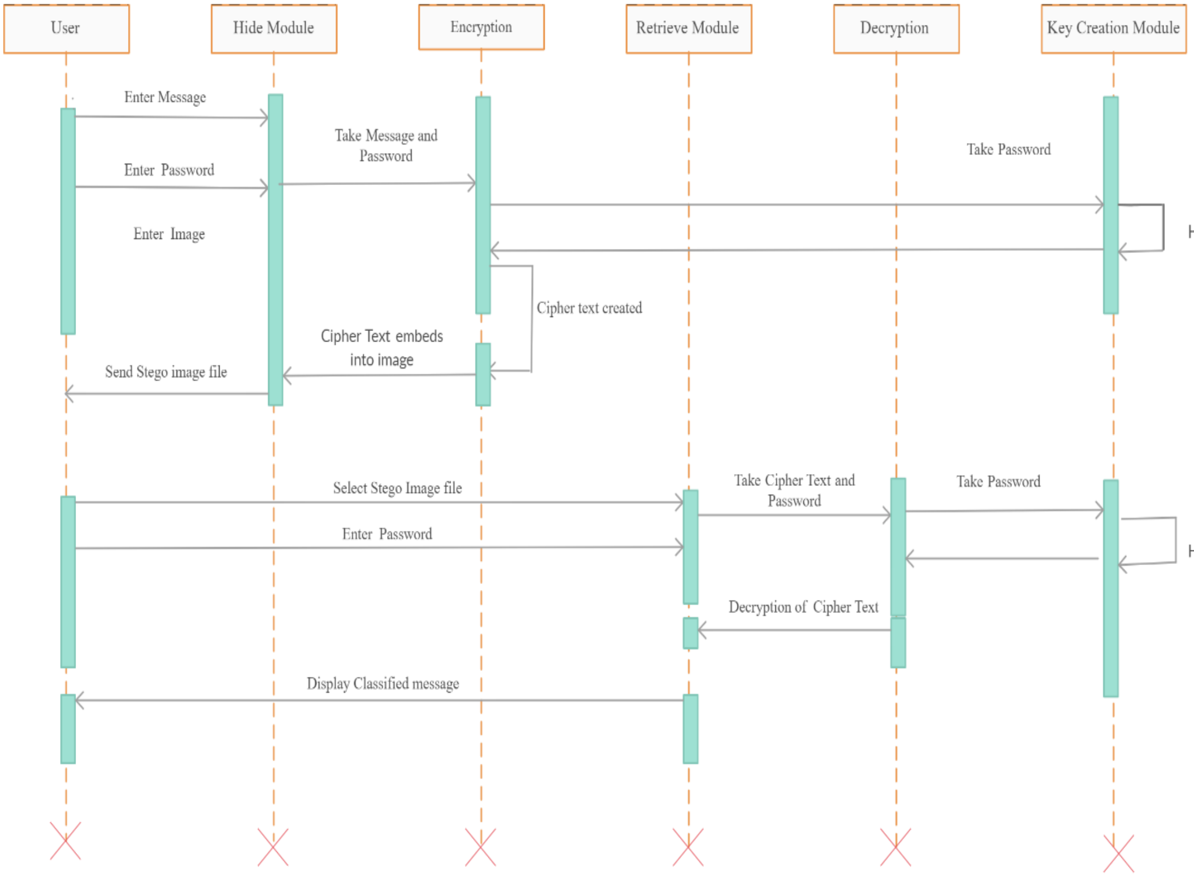


1-Level DFD

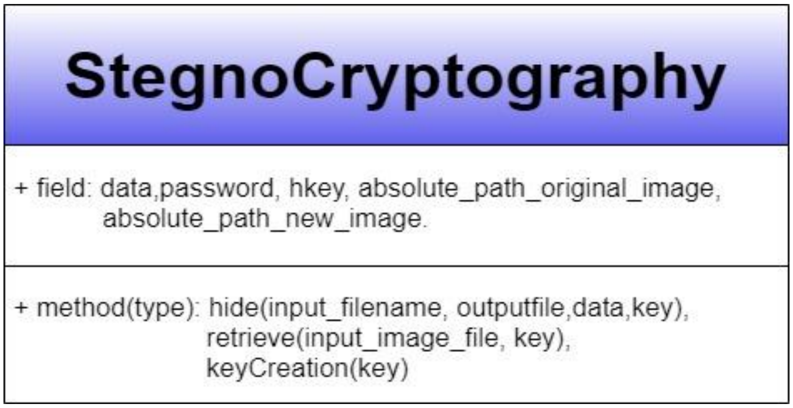
3.4.3 Use Case diagram: -



3.4.4 Sequence Diagram: -

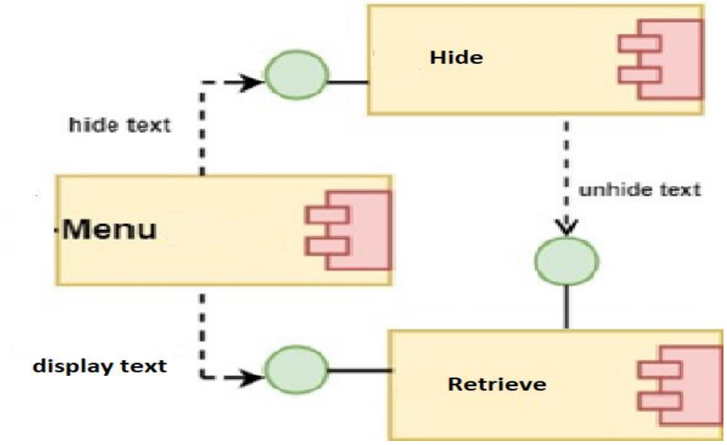


3.4.5 Class Diagram: -



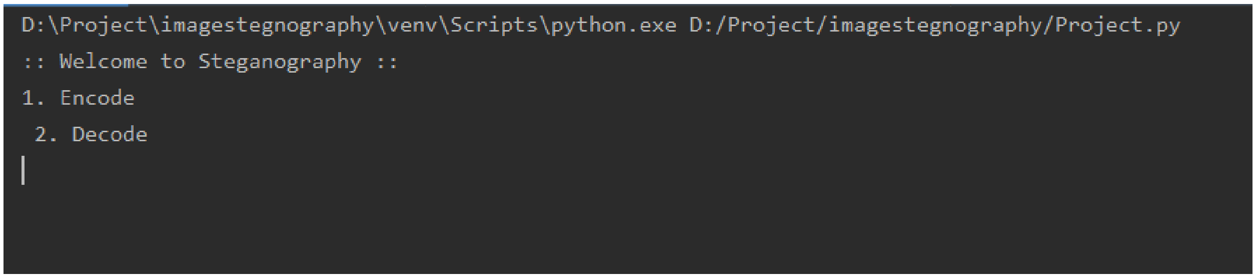
StegnoCryptography
+ field: data,password, hkey, absolute_path_original_image, absolute_path_new_image.
+ method(type): hide(input_filename, outputfile,data,key), retrieve(input_image_file, key), keyCreation(key)

3.4.6 Component Diagram: -



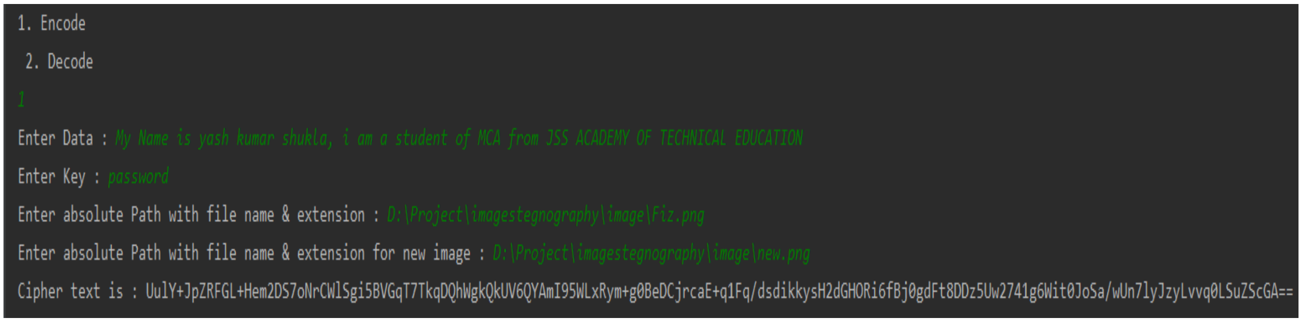
CHAPTER 4

4. IMPLEMENTATION RESULTS



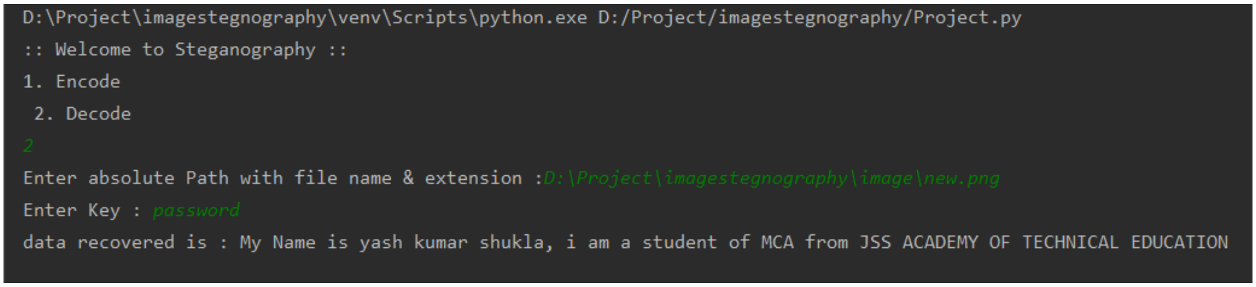
```
D:\Project\imagestegnography\venv\Scripts\python.exe D:/Project/imagestegnography/Project.py
:: Welcome to Steganography ::
1. Encode
 2. Decode
|
```

Starting Menu



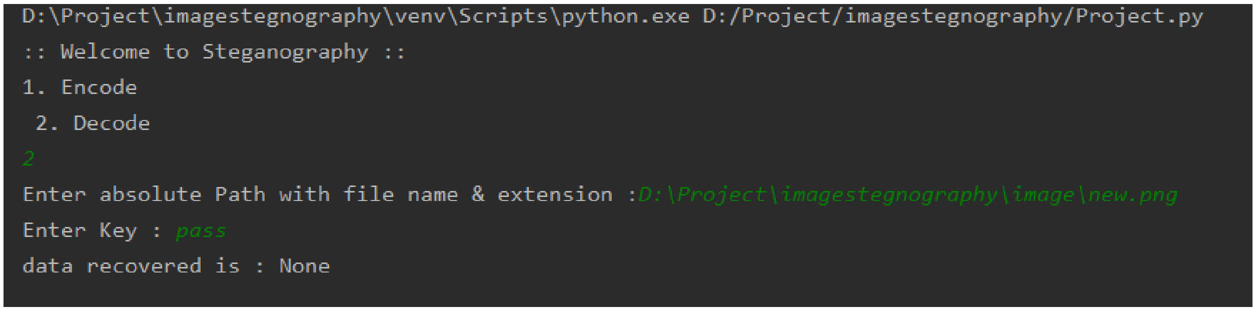
```
1. Encode
 2. Decode
1
Enter Data : My Name is yash kumar shukla, i am a student of MCA from JSS ACADEMY OF TECHNICAL EDUCATION
Enter Key : password
Enter absolute Path with file name & extension : D:\Project\imagestegnography\image\Fiz.png
Enter absolute Path with file name & extension for new image : D:\Project\imagestegnography\image\new.png
Cipher text is : UulY+JpZRFGL+Hem2DS7oNrCWlSgi5BVGqT7TkqDQhWgkQkUV6QYAmI95WLxRym+g0BeDCjrcaE+q1Fq/dsdikkysH2dGHORi6fBj0gdFt8DDz5Uw2741g6Wit0JoSa/wUn7lyJzyLvvq0LSuZScGA==
```

Encoding



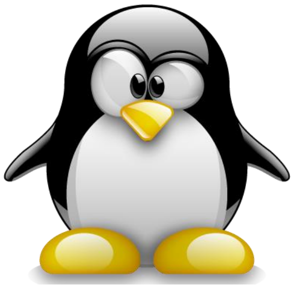
```
D:\Project\imagestegnography\venv\Scripts\python.exe D:/Project/imagestegnography/Project.py
:: Welcome to Steganography ::
1. Encode
 2. Decode
2
Enter absolute Path with file name & extension :D:\Project\imagestegnography\image\new.png
Enter Key : password
data recovered is : My Name is yash kumar shukla, i am a student of MCA from JSS ACADEMY OF TECHNICAL EDUCATION
```

Decoding

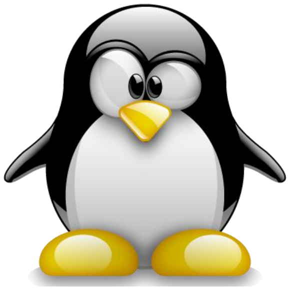


```
D:\Project\imagestegnography\venv\Scripts\python.exe D:/Project/imagestegnography/Project.py
:: Welcome to Steganography ::
1. Encode
 2. Decode
2
Enter absolute Path with file name & extension :D:\Project\imagestegnography\image\new.png
Enter Key : pass
data recovered is : None
```

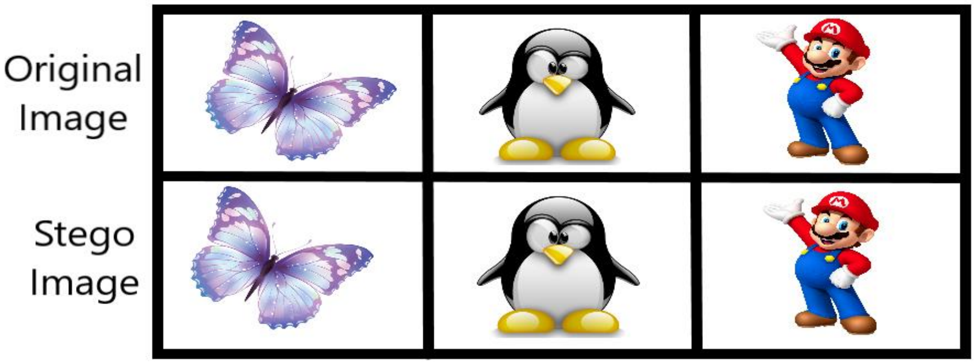
Decoding (Wrong key)



Original image (Fiz.png)



Encoded image (new.png)



Original and Stego Image

Table 6: Result after performing the experiment

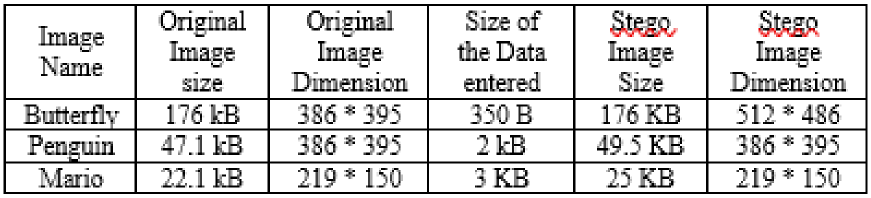
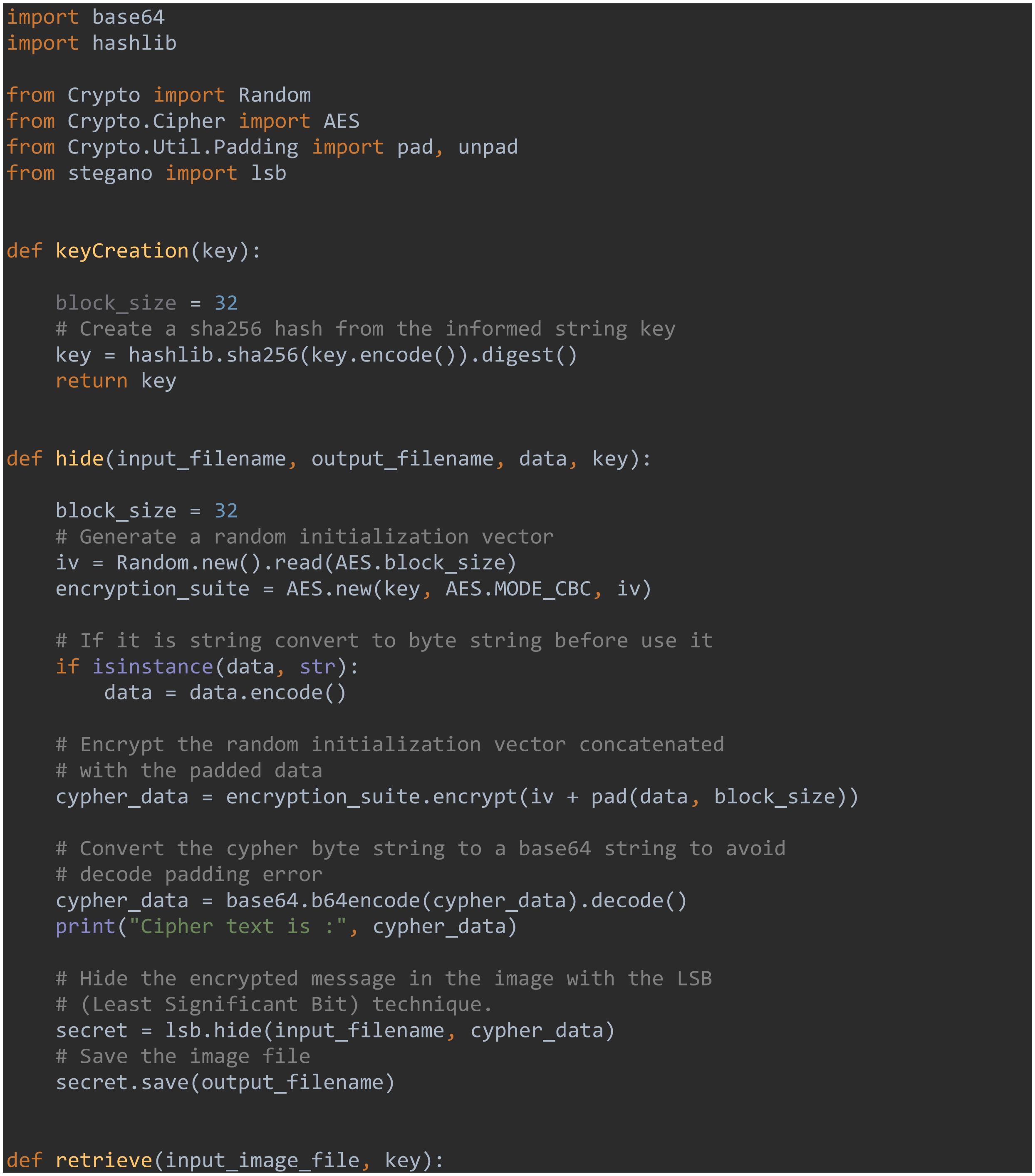


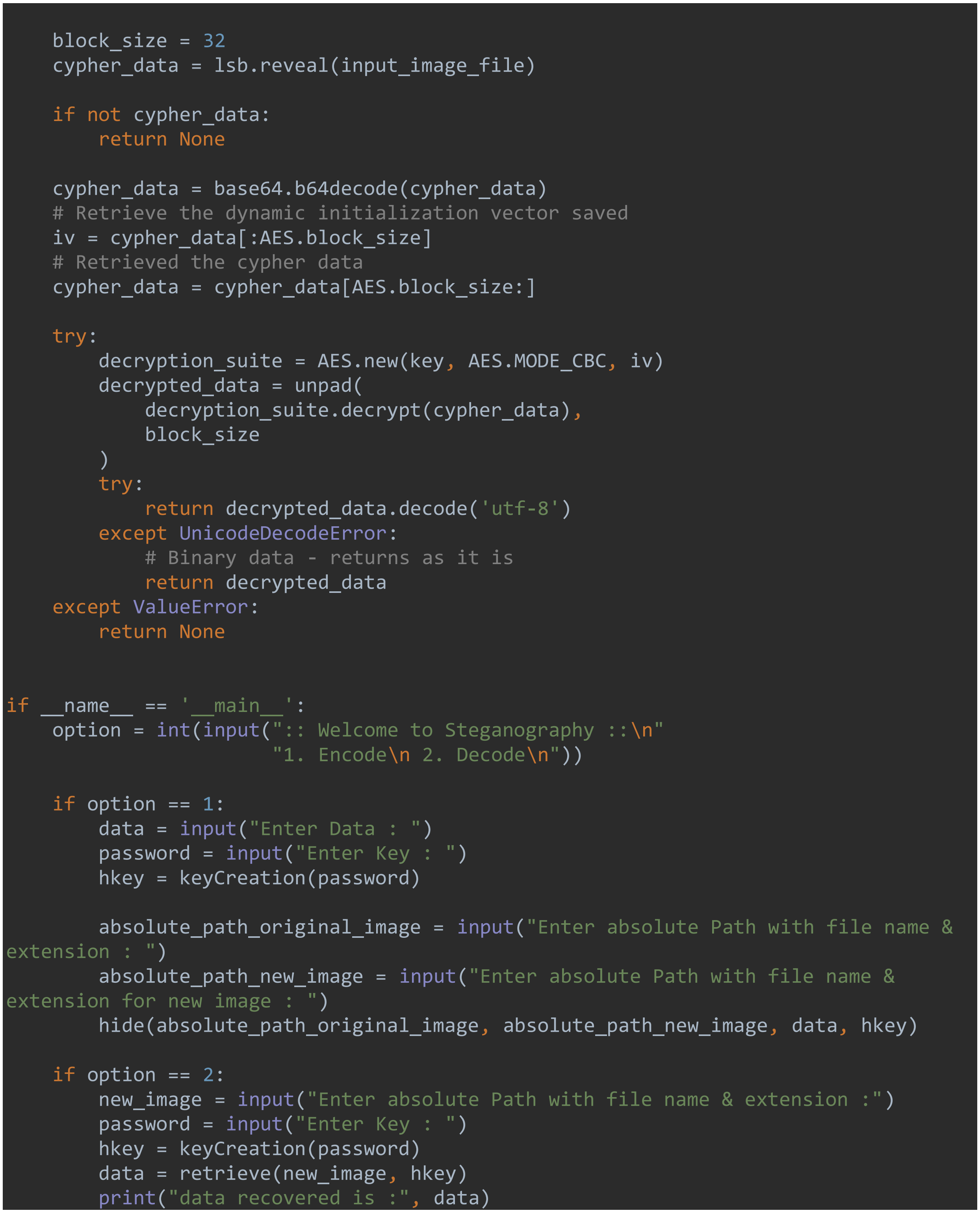
Image Name	Original Image size	Original Image Dimension	Size of the Data entered	Stego Image Size	Stego Image Dimension
Butterfly	176 kB	386 * 395	350 B	176 KB	512 * 486
Penguin	47.1 kB	386 * 395	2 kB	49.5 KB	386 * 395
Mario	22.1 kB	219 * 150	3 KB	25 KB	219 * 150

CHAPTER 5

Source Code



```
import base64
import hashlib

from Crypto import Random
from Crypto.Cipher import AES
from Crypto.Util.Padding import pad, unpad
from stegano import lsb


def keyCreation(key):

    block_size = 32
    # Create a sha256 hash from the informed string key
    key = hashlib.sha256(key.encode()).digest()
    return key


def hide(input_filename, output_filename, data, key):

    block_size = 32
    # Generate a random initialization vector
    iv = Random.new().read(AES.block_size)
    encryption_suite = AES.new(key, AES.MODE_CBC, iv)

    # If it is string convert to byte string before use it
    if isinstance(data, str):
        data = data.encode()

    # Encrypt the random initialization vector concatenated
    # with the padded data
    cypher_data = encryption_suite.encrypt(iv + pad(data, block_size))

    # Convert the cypher byte string to a base64 string to avoid
    # decode padding error
    cypher_data = base64.b64encode(cypher_data).decode()
    print("Cipher text is :", cypher_data)

    # Hide the encrypted message in the image with the LSB
    # (Least Significant Bit) technique.
    secret = lsb.hide(input_filename, cypher_data)
    # Save the image file
    secret.save(output_filename)


def retrieve(input_image_file, key):
```



```
    block_size = 32
    cypher_data = lsb.reveal(input_image_file)

    if not cypher_data:
        return None

    cypher_data = base64.b64decode(cypher_data)
    # Retrieve the dynamic initialization vector saved
    iv = cypher_data[:AES.block_size]
    # Retrieved the cypher data
    cypher_data = cypher_data[AES.block_size:]

    try:
        decryption_suite = AES.new(key, AES.MODE_CBC, iv)
        decrypted_data = unpad(
            decryption_suite.decrypt(cypher_data),
            block_size
        )
        try:
            return decrypted_data.decode('utf-8')
        except UnicodeDecodeError:
            # Binary data - returns as it is
            return decrypted_data
    except ValueError:
        return None


if __name__ == '__main__':
    option = int(input(":: Welcome to Steganography ::\n"
                       "1. Encode\n 2. Decode\n"))

    if option == 1:
        data = input("Enter Data : ")
        password = input("Enter Key : ")
        hkey = keyCreation(password)

        absolute_path_original_image = input("Enter absolute Path with file name &
extension : ")
        absolute_path_new_image = input("Enter absolute Path with file name &
extension for new image : ")
        hide(absolute_path_original_image, absolute_path_new_image, data, hkey)

    if option == 2:
        new_image = input("Enter absolute Path with file name & extension :")
        password = input("Enter Key : ")
        hkey = keyCreation(password)
        data = retrieve(new_image, hkey)
        print("data recovered is :", data)
```

CHAPTER 6

6. TESTING

Testing Objectives:

My web application doesn't have to be perfect; it just needs to meet intended customer's requirements and expectations.
Software Testing is done to find errors, bug and faults in the software.

Testing performed for testing this website, are as follows:

6.1 Usability Testing-

In usability testing, we looked at aspects of my application that affect the user's experience, such as:

- How easy is it to navigate through my application?
- Is the look-and-feel of your application consistent from page to page, including font sizes and colours

6.2 Unit Testing: -

Unit testing is a testing of software at individual units/ components of a software are tested. The purpose is to validate that each unit of the software works accordingly. A unit is the smallest testable part of any software which could be called a module. It takes a few inputs parameters and usually a single output.

For example, if you are testing a function or method, whether loop or any statement in a program is working efficiently or not then this is called as unit testing.

6.3 Integration Testing: -

Integration testing is a phase in software testing in which each unique software modules are combined and tested as a group. This phase of testing is done after unit testing and before validation testing.

- Top-down Integration Testing
- Sandwich Integration Testing
- Bottom-UP Integration Testing.

6.4 User Acceptance Testing: -

User acceptance testing (UAT) - also called beta testing, application testing, and end-user testing it is a phase of software development in which the software is tested in the "real world" by the intended audience or user group.

6.5 Performance Testing: -

Performance testing, it is a testing technique which is a non-functional requirement performed to determine the system parameters in terms of responsiveness and stability under the various workload. Performance testing estimates the quality attributes of the system, such as scalability, reliability and resource usage.

- **Load testing -** It is a simple form of testing which is performed in order to understand the behavior of the system under a specific load. Load testing concludes in measuring important business-critical transactions and load on the database, application server, etc., are also monitored.
- **Stress testing -** It is conducted to find the maximum capacity of the system and to determine how the system performs if the current load goes well above the expected maximum.
- **Soak testing -** Soak Testing is an endurance test, and it is executed in order to determine the system parameters under the continuous expected load. During soak tests, the parameters such as memory utilization are monitored to detect memory leaks or other performance issues. The main aim is to find the system's performance under desirable use.

- **Spike testing -** Spike testing is executed by increasing the number of users instantaneously by a very large amount and measuring the performance of the system. The main aim is to determine whether the system will be able to sustain the workload.

Attribute Tested:

- Speed
- Scalability
- Stability
- Reliability

CHAPTER 7

7. LIMITATIONS AND FUTURE EHANCEMENTS

7.1 Limitations: -

- The limitation is it does not allow to transfer from application itself.
- It works upon the lossy compression image format that is BMP, PNG, etc.

7.2 Future enhancements: -

- Transfer images using the application itself.
- Other mediums can also be used for the cover as audio, video.
- Adding more advance Cryptographic and Hashing algorithm.

CHAPTER 8

8. References

http://www.ijitee.org/wp-content/uploads/papers/v9i8/H6442069820.pdf

https://computerresearch.org/index.php/computer/article/view/912/911

https://csrc.nist.gov/csrc/media/publications/fips/197/final/documents/fips-197.pdf

https://medium.com/hackernoon/simple-image-steganography-in-python-18c7b534854f

https://www.geeksforgeeks.org/lsb-based-image-steganography-using-matlab/

https://www.geeksforgeeks.org/image-based-steganography-using-python/

https://pypi.org/project/stegano/

https://pypi.org/project/pycryptodome/

http://www.ijmer.com/papers/Vol2_Issue6/EN2646344638.pdf

https://shodhganga.inflibnet.ac.in/bitstream/10603/97221/10/th-1816_ch8.pdf