

Configuration Manual

MSc Research Project
MSc In Cybersecurity

Sakshi Suresh Phadtare
Student ID: X22186671

School of Computing
National College of Ireland

Supervisor: Prof. Khadija Hafeez

National College of Ireland
MSc Project Submission Sheet
School of Computing



Student Name: Sakshi Suresh Phadtare
Student ID: x22186671
Programme: MSc in Cybersecurity **Year:** 2023-2024
Module: MSc Internship Project
Lecturer: Khadija Hafeez
Submission Due Date: 12/08/2024
Project Title: Advanced Image Steganography Using Pixel-Value Differencing and AES Encryption
Word Count: 1392 **Page Count:** 13

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Sakshi.S.Phadtare

Date: 12/08/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Sakshi Suresh Phadtare
Student ID: x22186671

1 Introduction

This research focuses on Image Steganography by using PVD and AES and CHACHA algorithm for encryption which is used for secure communication. The project is build using python language and applies a number of libraries for encryption, image manipulation and for creating a web application interface.

This handbook is consisting of all the required information related to instructions and preparation while delivering the model of project. This includes project overview, hardware, libraries and software configuration.

2 Hardware Configuration

- A. Operating System: Windows >=10
- B. Processor: Intel >=i3
- C. System Compatibility: 64-bit
- D. Hard Disk: 500 GB
- E. RAM: 8 GB

3 Software Configurations

3.1 Python 3.10.12

It is also a very generalized and easily manageable programming language which any new programmers as well as the professionals can use. It has a less complex grammar through which the programmer can express a single idea or plan of action in fewer lines of code than it is possible to do in other languages, this makes it more precise and effective. I found that these are procedural, object oriented and functional and it has support a number of libraries and frameworks for Web, data analysis, AI and many more. Because of this flexibility and reliability, the scripts have been sought frequently in the spheres of activity including software development and scientific analysis (Python, 2019).

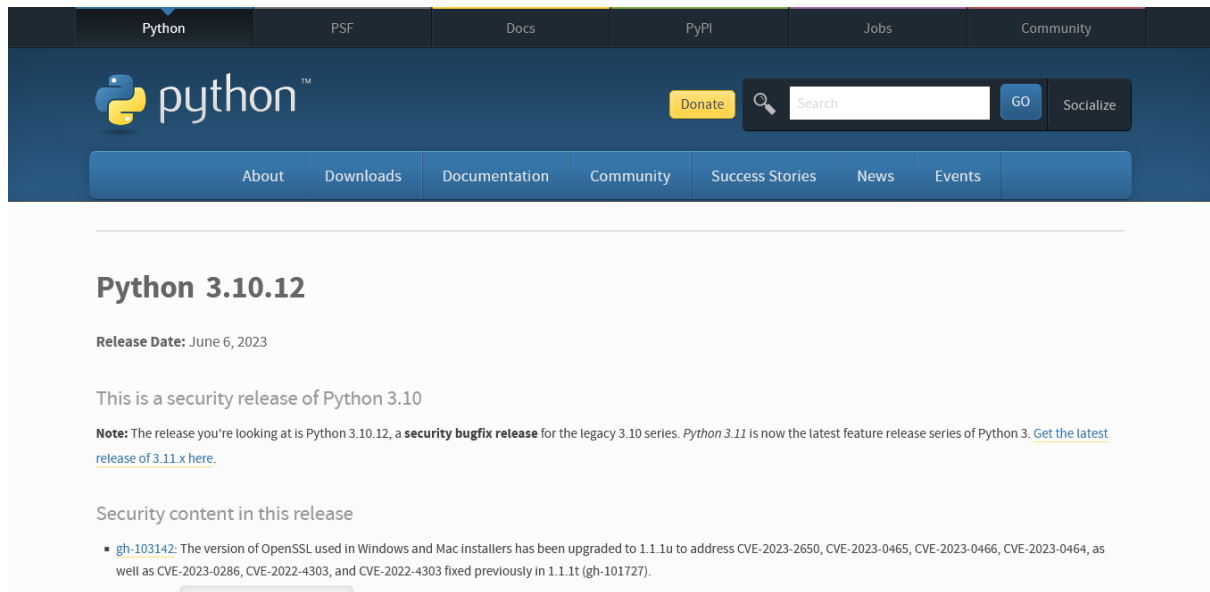


Figure 3.1: Python Application

3.2 Visual Studio Code (VS Code)

Visual Studio Code (VS Code) – is a free code editor that is relatively simple and efficient, which is, in fact, developed by Microsoft. Very often it is used due to the high number of features available, especially the option to work in various programming languages through extensions. It is also stable for code suggestion, various debuggers, the appropriate terminal, and the integration of GitHub, which make the development very convenient in VS Code. General trading platform offers call for a vast number of extensions and themes that improve rendering for developers and organization of work on them. Due to these and other daily updates and users' interactions, VS Code is one of the most useful IDEs that developers can employ while working within the present conditions (Visual Studio Code, 2023).

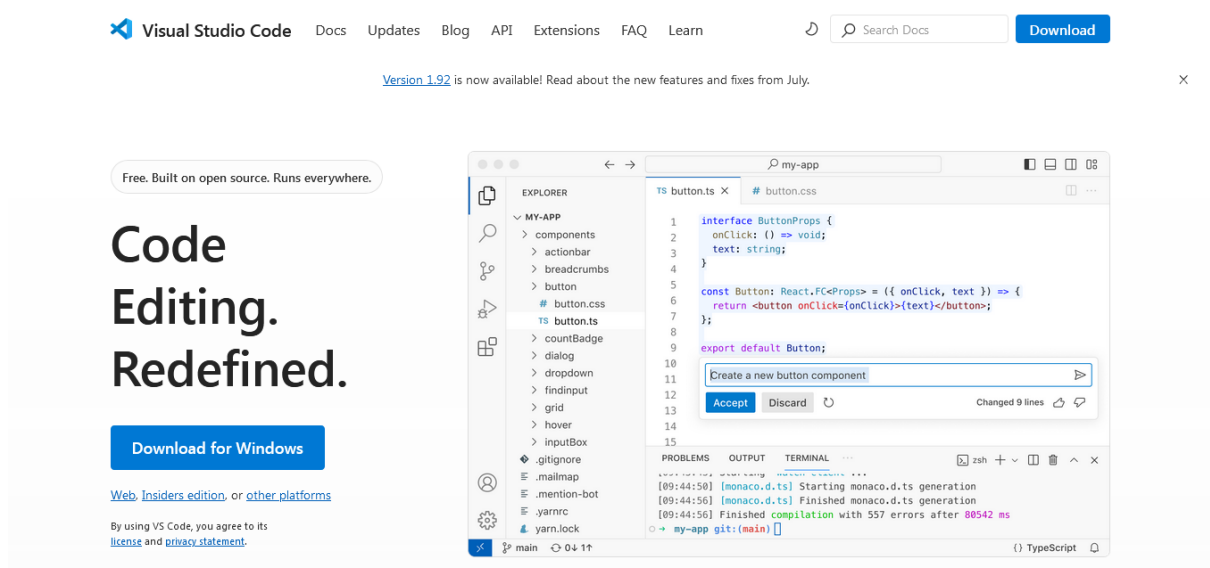


Figure 3.2: Visual Studio Code Application

4 Libraries Configuration

- A. OpenCV (cv2):** Coordinates image handling tasks like reading, modifying, saving an image, which are core to embedding/stealthily pulling information out of images.
- B. Cryptodome (AES, get_random_bytes):** Proper encryption and decryption services are provided through the AES algorithm as well as the generation of secure random bytes for key management to guarantee the privacy of your data.
- C. Cryptography (PKCS7, Cipher, algorithms, modes):** Offers padding schemes and cipher modes which are important when it comes to secure encryption methods such as cipher ChaCha and integrity.
- D. Flask:** Allows for the setup of a project's web front-end, which helps users interact with the steganography tool and lets users use the steganography tool through a web based application.

5 Usage

5.1 Encrypting a Message

1. Navigate to the /serviceencrypt page.
2. Upload the image to be used for encoding.
3. Enter the secret message you want to hide.
4. Enter the recipient's email address where the encoded image and AES OR CHACHA key will be sent.
5. Submit the form.

The application will:

- Encrypt the message using AES OR CHACHA.
- Embed the encrypted message into the image using PVD.
- Save the resulting image and AES key.
- Send an email with the image and key to the specified recipient.

5.2 Decrypting a Message

1. Navigate to the /servicedecrypt page.
2. Upload the encoded image.
3. Enter the AES OR CHACHA key that was used for encryption.
4. Submit the form.

The application will:

- Extract the encrypted message from the image using PVD.

- Decrypt the message using AES OR CHACHA.
- Display the decrypted message on the page.

6 Project Implementation

This is the interface created for the project. This interface demonstrates the image steganography system. In which we have two options one for steganography encryption and decryption.

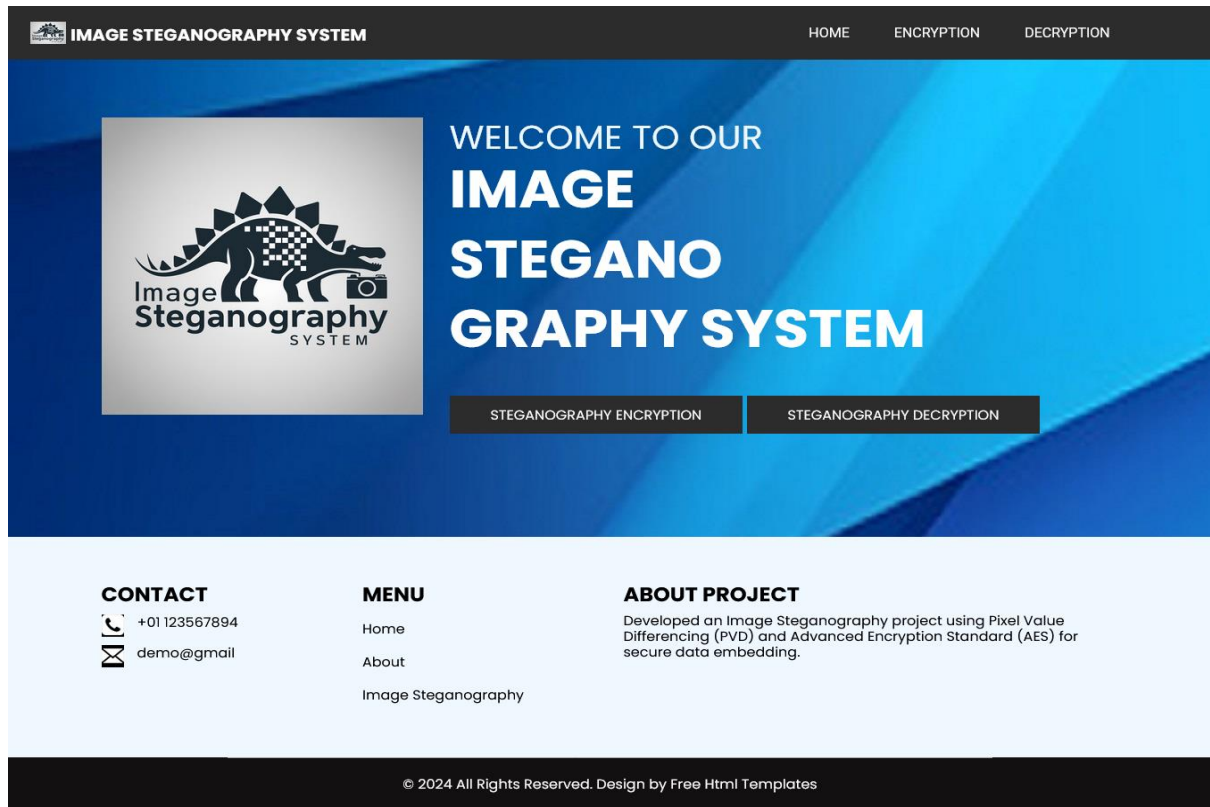


Figure 6.1: Web Application Interface

6.1 OUTPUT SCREEN IMAGE STEGANOGRAPHY SYSTEM USING AES

6.1.1 Encryption Process

The figure 6.2 demonstrates the web interface for generating PVD steganography image with AES. Where user has to upload the image behind which he wants to hide the text. The other box is where user has to enter the text data and email ID of the recipient. And then he can generate steganography image with AES.

IMAGE STEGANOGRAPHY SYSTEM

HOMEENCRYPTIONDECRYPTION

SELECT IMAGE AND HIDE INFORMATION IN IMAGE USING IMAGE STEGANOGRAPHY PVD ENCRYPTION

Select/Upload Image File

ENTER MESSAGE TO HIDE IN IMAGE STEGANOGRAPHY

Tonight I am sad.

ENTER EMAIL ID

Enter Email ID

GENERATE PVD STEGANOGRAPHY IMAGE WITH AES

CONTACT

+01 123567894

demo@gmail

MENU

Home

About

Image Steganography


ABOUT PROJECT

Developed an Image Steganography project using Pixel Value Differencing (PVD) and Advanced Encryption Standard (AES) for secure data embedding.

© 2024 All Rights Reserved. Design by Free Html Templates

Figure 6.2: Interface for Image Encryption with AES

Figure 6.3 demonstrates the interface of steganography image generated. After image generation user get the ciphertext, encryption time, and steganography image. At the same time, recipient get the email of stegano image and key for decryption.


IMAGE STEGANOGRAPHY SYSTEM

[HOME](#)
[ENCRYPTION](#)
[DECRYPTION](#)

SELECT IMAGE AND HIDE INFORMATION IN IMAGE USING IMAGE STEGANOGRAPHY PVD ENCRYPTION

Select/Upload Image File

ENTER MESSAGE TO HIDE IN IMAGE STEGANOGRAPHY

Enter Message

ENTER EMAIL ID

Enter Email ID

GENERATE PVD STEGANOGRAPHY IMAGE WITH AES

IMAGE STEGANOGRAPHY GENERATED SUCCESSFULLY



b'+O \ xfa \ x0b \ x9c \ xa6 \ x9b \ xff \ x1b_m \ xaf \ x7fG* \ xd6 \ xa7'

Encryption Process Time: 9331.29



Download Steganography Image

Original Image

Stegano Image

CONTACT

 +01 123567894
  demo@gmail

MENU

[Home](#)
[About](#)
[Image Steganography](#)

ABOUT PROJECT

Developed an Image Steganography project using Pixel Value Differencing (PVD) and Advanced Encryption Standard (AES) for secure data embedding.

© 2024 All Rights Reserved. Design by Free Html Templates

Figure 6.3: interface for uploading image, text data and mail id

6.1.2 Decryption Process

Figure 6.4 demonstrates the web application interface for retrieving information from steganography images. Here, the recipient has to upload the stegano image and has to enter the AES key to decrypt the message. Then recipient has to click on the retrieve information.

6

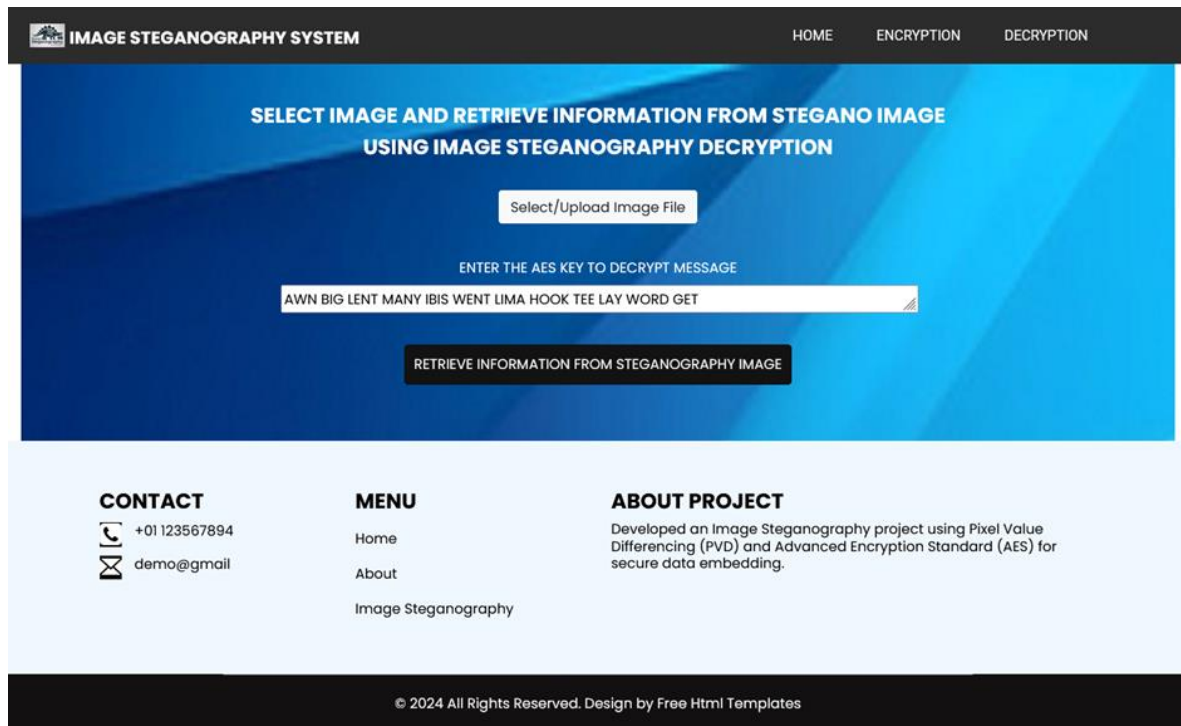


Figure 6.4: interface for retrieving process with key

Figure 6.5 demonstrates the interface of decryption. By entering required information recipient get the retrieved message and decryption process time.

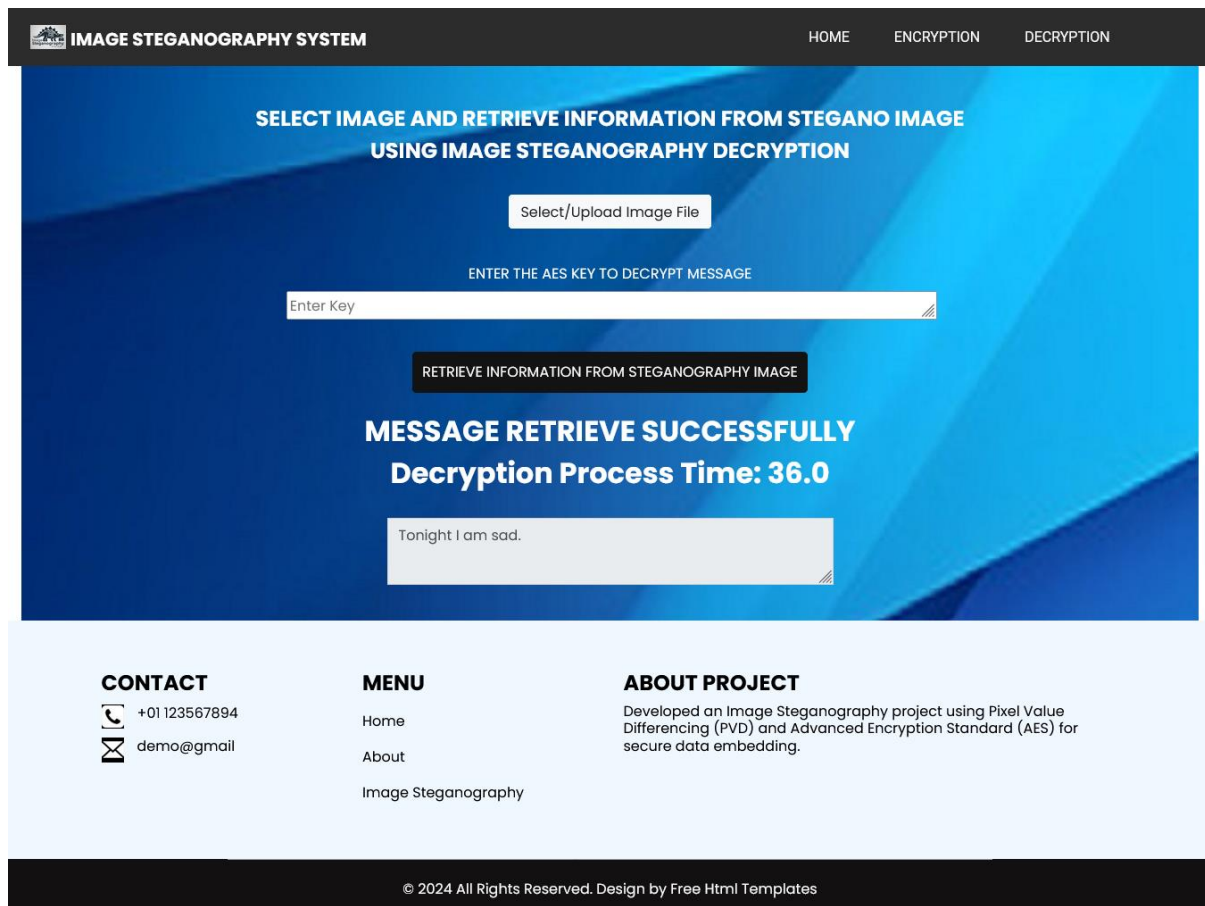
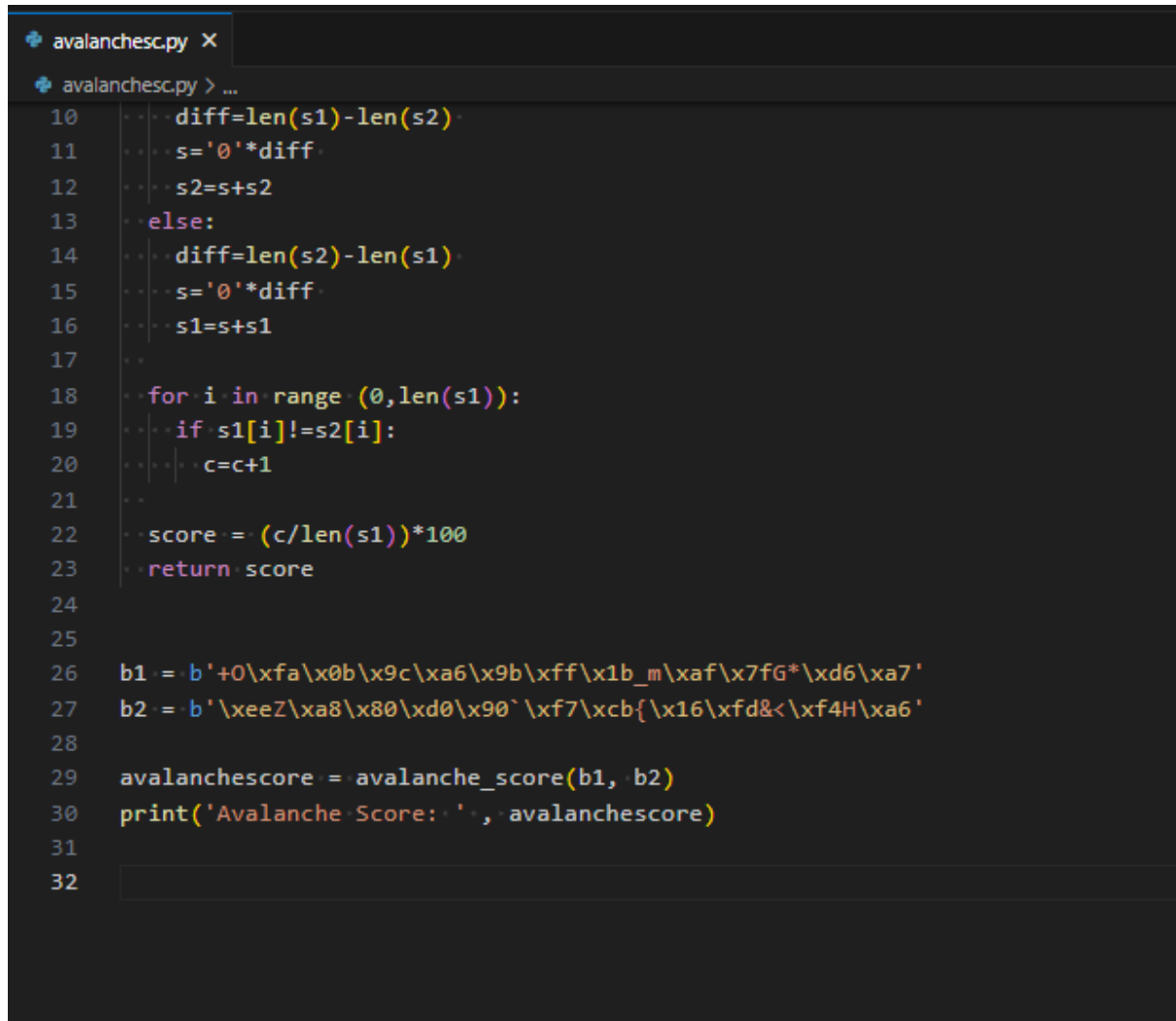


Figure 6.5: interface for decryption process

6.1.3 Avalanche Score

Figure 6.6 demonstrates a python script which calculates the Avalanche score between two byte sequences. This is the Avalanche score calculated for AES algorithm. It is calculated on the basis of difference between encryption and decryption time. Calculated avalanche score is 49.86.



```
10     diff=len(s1)-len(s2)
11     s='0'*diff
12     s2=s+s2
13     else:
14         diff=len(s2)-len(s1)
15         s='0'*diff
16         s1=s+s1
17
18     for i in range(0,len(s1)):
19         if s1[i]!=s2[i]:
20             c=c+1
21
22     score = (c/len(s1))*100
23     return score
24
25
26 b1 = b'+0\xfa\x0b\x9c\xa6\x9b\xff\x1b_m\xaf\x7fG*\xd6\xa7'
27 b2 = b'\xeeZ\xa8\x80\xd0\x90'\xf7\xcb{\x16\xfd&<\xf4H\xa6'
28
29 avalanchescore = avalanche_score(b1, b2)
30 print('Avalanche Score: ', avalanchescore)
31
32
```

Figure 6.6: Avalanche Score Calculation for AES Algorithm

Avalanche Score: 49.86072423398329

6.2 OUTPUT SCREEN IMAGE STEGANOGRAPHY SYSTEM USING CHACHA

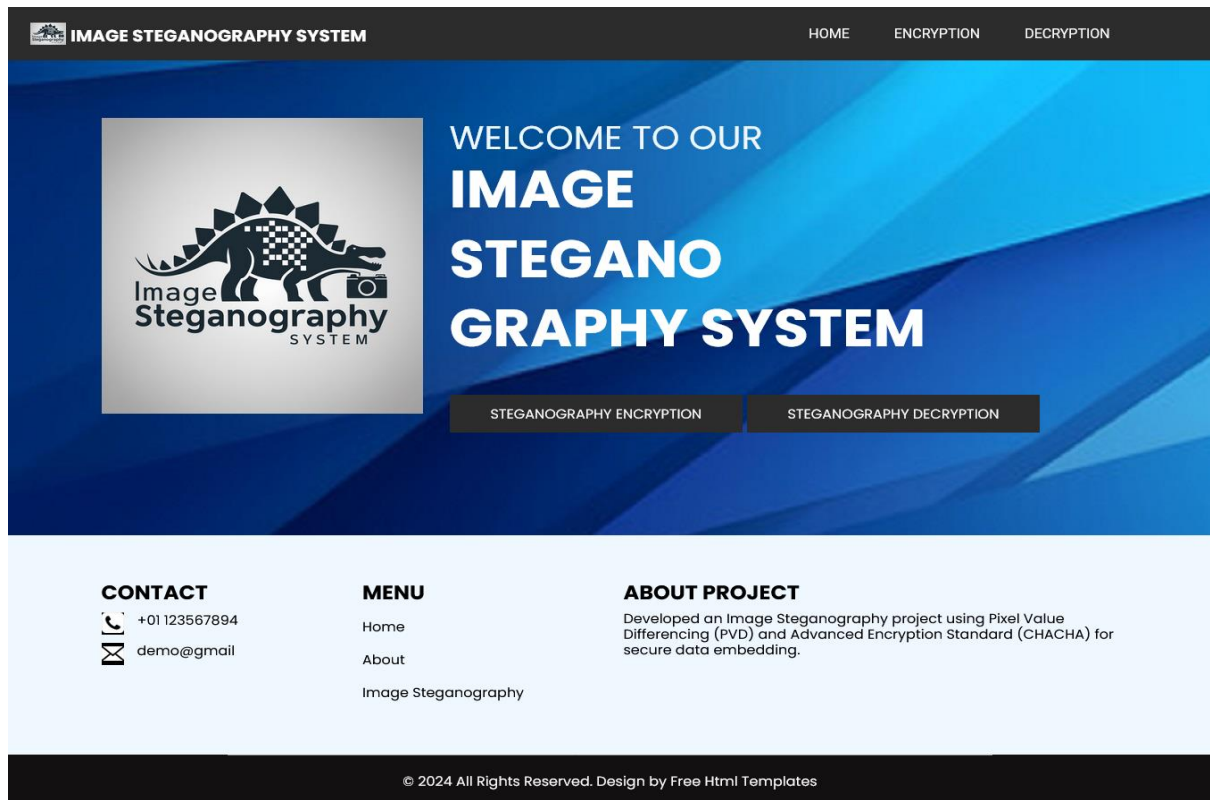


Figure 6.7: Web App Interface for ChaCha Algorithm

6.2.1 Encryption Process

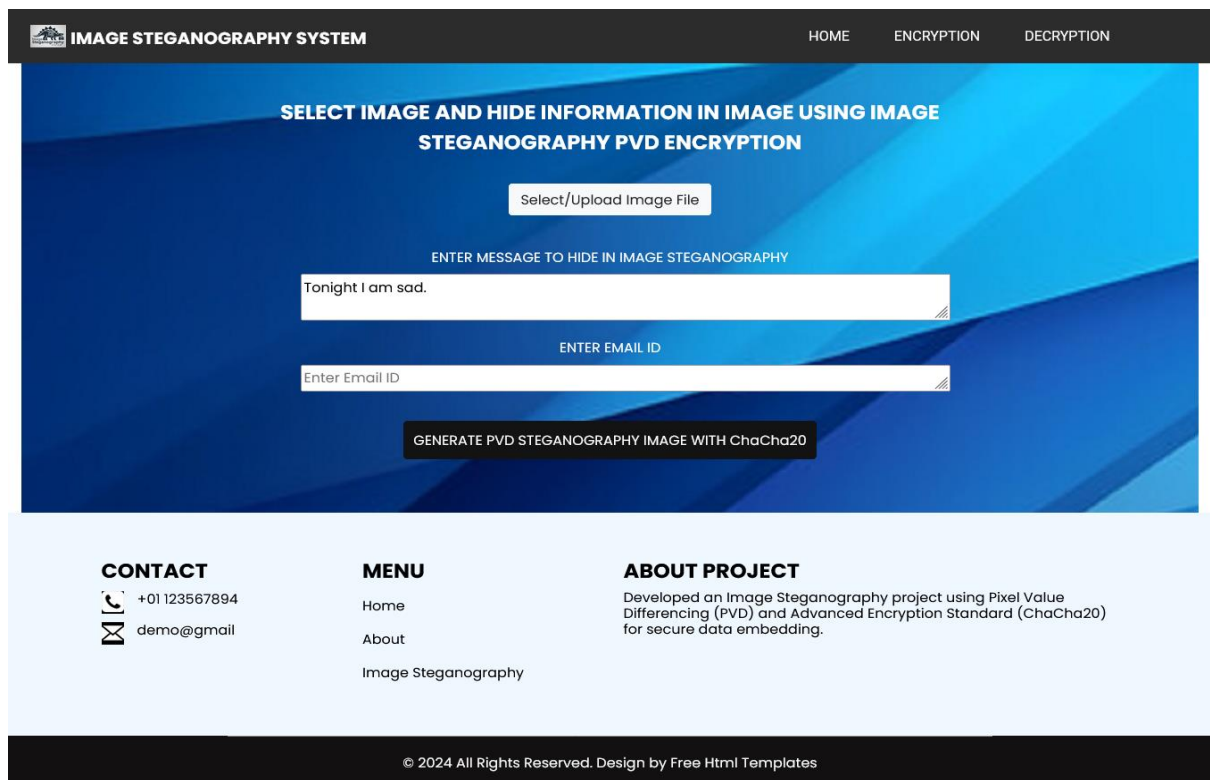


Figure 6.8: Interface for Uploading data for encryption

Figure 6.8 demonstrates the process of encryption by ChaCha algorithm. Where user has to upload image, text and email ID.

Figure 6.9 shows the interface of stegano image generated with ciphertext and encryption time

IMAGE STEGANOGRAPHY SYSTEM

HOMEENCRYPTIONDECRIPTION

SELECT IMAGE AND HIDE INFORMATION IN IMAGE USING IMAGE STEGANOGRAPHY PVD ENCRYPTION

Select/Upload Image File

ENTER MESSAGE TO HIDE IN IMAGE STEGANOGRAPHY

Enter Message

ENTER EMAIL ID

Enter Email ID

GENERATE PVD STEGANOGRAPHY IMAGE WITH ChaCha20

IMAGE STEGANOGRAPHY GENERATED SUCCESSFULLY

b' \ x02P \ xce \ xa4k \ xeb \ xdb \ x1dv" \ x03 \ x1e \ xff \ x9cG \ x97|'

Encryption Process Time: 8637.92

Download Steganography Image

Original Image

Stegano Image

CONTACT

+01 123567894

demo@gmail

MENU

Home

About

Image Steganography

ABOUT PROJECT

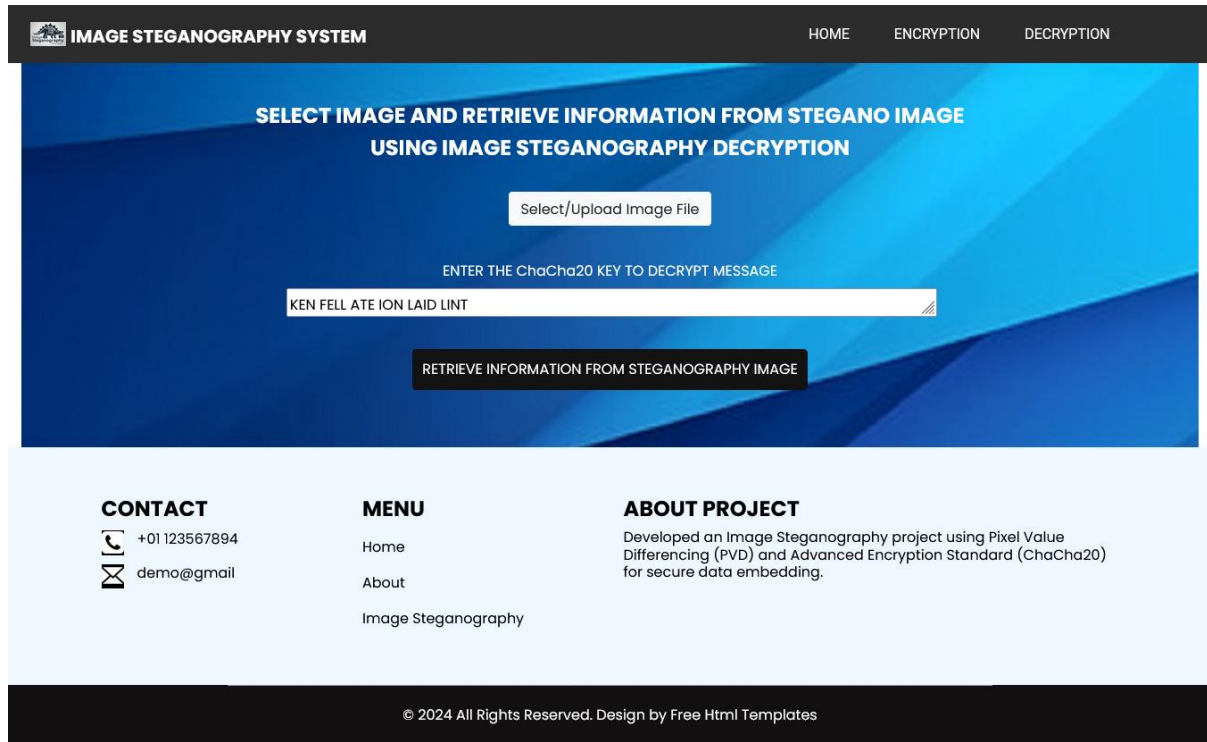
Developed an Image Steganography project using Pixel Value Differencing (PVD) and Advanced Encryption Standard (ChaCha20) for secure data embedding.

© 2024 All Rights Reserved. Design by Free Html Templates

Figure 6.9: Encrypted text Interface

6.2.2 Decryption Process

Figure 6.10 is the interface for decryption process. In which recipient has to upload the stegano image and key received on the mail.



The screenshot displays the 'IMAGE STEGANOGRAPHY SYSTEM' interface for decryption. The header includes a logo and navigation links for HOME, ENCRYPTION, and DECRYPTION. The main section, titled 'SELECT IMAGE AND RETRIEVE INFORMATION FROM STEGANO IMAGE USING IMAGE STEGANOGRAPHY DECRYPTION', features a 'Select/Upload Image File' button, a text input field for the 'ChaCha20 KEY TO DECRYPT MESSAGE' (containing 'KEN FELL ATE ION LAID LINT'), and a 'RETRIEVE INFORMATION FROM STEGANOGRAPHY IMAGE' button. The footer contains three columns: 'CONTACT' with phone and email details, 'MENU' with links to Home, About, and Image Steganography, and 'ABOUT PROJECT' describing the use of Pixel Value Differencing (PVD) and Advanced Encryption Standard (ChaCha20). A copyright notice for 2024 is at the bottom.

Figure 6.10: Interface for decryption process

Figure 6.11 demonstrates the interface of decryption. After uploading required information like stegano image and key of encryption. The recipient will get the original message and the decryption process time.

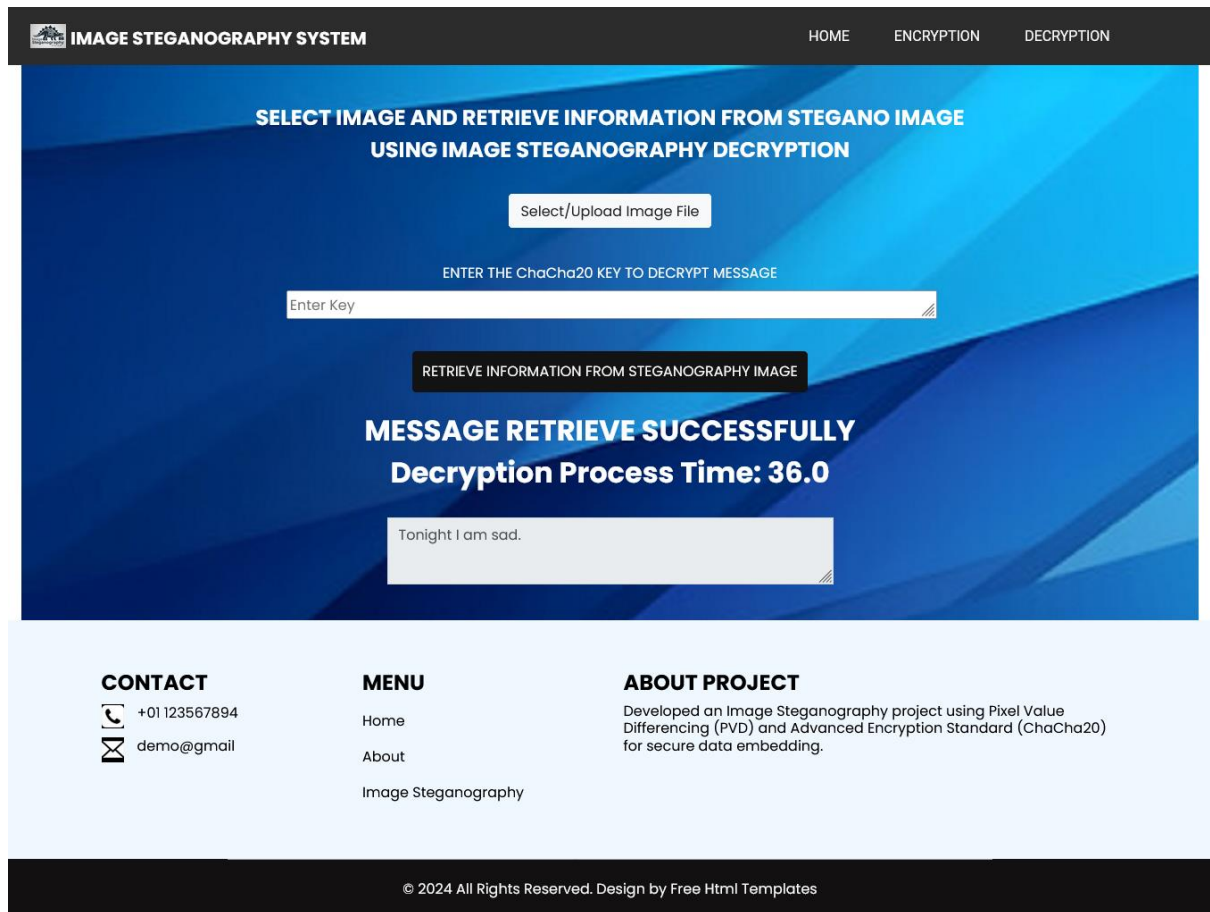


Figure 6.11: Interface for Uploading data for Decryption Process

6.2.3 Avalanche Score

Figure 6.12 demonstrates the python script for calculating avalanche score. The avalanche score for ChaCha algorithm is 49.52.

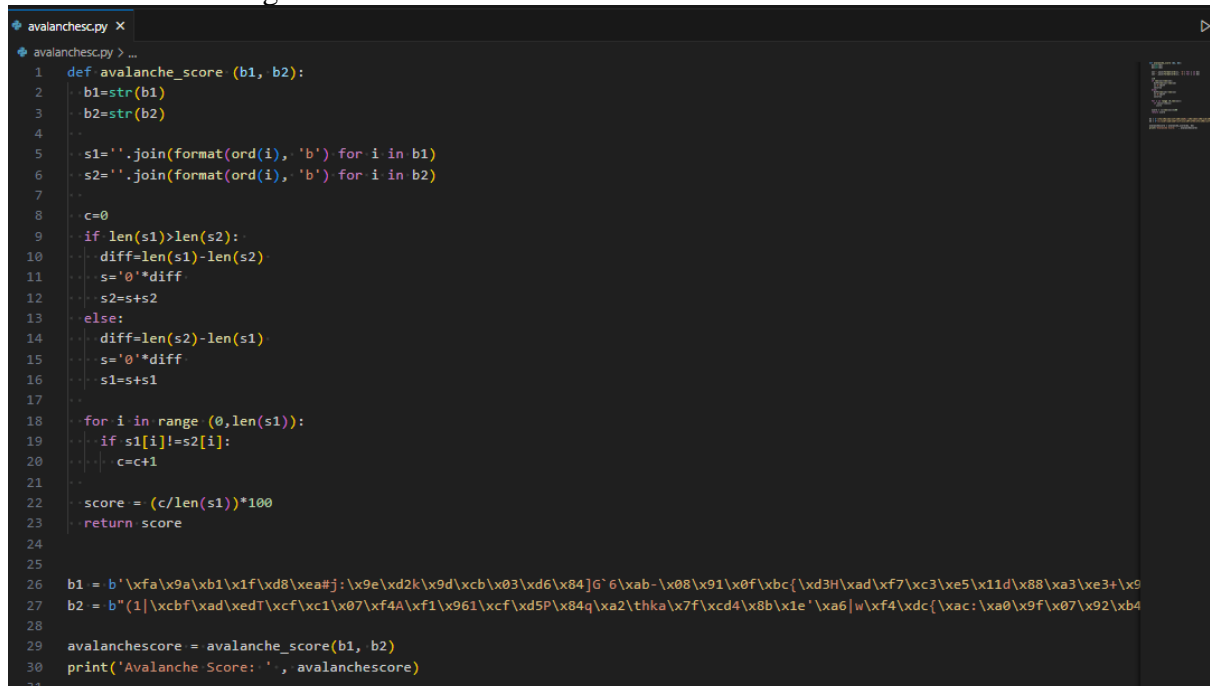


Figure 6.12: Avalanche Score Calculation for ChaCha Algorithm

Avalanche Score: 49.523809523809526

References

1. Python (2019). Python. [online] Python.org. Available at: <https://www.python.org/>.
2. Visual Studio Code (2023). Documentation for Visual Studio Code. [online] code.visualstudio.com. Available at: <https://code.visualstudio.com/docs>.