NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

AN EFFICIENT DATA ENCRYPTION MECHANISM BY USING OPTIMAL LSB TECHNIQUE

UNDER THE ESTEEMED GUIDANCE OF:

Dr.S.Siva Nageswara Rao *M.Tech.,Ph.D.*

Professor

PRESENTED BY

Sk.Thameem Ansari(19471A0558)

K.Chaitanya(19471A0527)

N.BalaKrishna(19471A0540)

PROJECT AGENDA

- Abstract
- Literature Survey
- System Requirements
- Analysis
- Literature Survey
- Proposed System
- Design Phase
- Implementation Phase
- Testing Phase
- References

Abstract

- In modern era data is heavily gaining importance as information is dependent on the raw fats i.e data. The exchange of information is required to share resources among the distributed users which may be separated by locations. While transferring the data among the users confidentiality and privacy should be maintained.

- This digitally shared data between the users should be converted to some unreadable format which will not be tampered by intruders. Data must be transmitted or sent in a secure way that the intruders could not corrupt it. To meet these requirements the technique Steganography can be used.

- This project report intends to give an overview of image steganography, its uses and techniques. It also attempts to identify the requirements of a good steganography algorithm and briefly reflects on which steganographic techniques are more suitable for which applications.

Literature Survey

- Marghny and Loay introduced a data hiding technique based on simple LSB substitution scheme for gray images. After embedding, they apply an optimal LSBs method to improve the quality of stego image. The experimental results indicate that their method has better performance compared to the corresponding methods, in terms of capacity.
- Kamaldeep and Rajkumar presented a new scheme based on XOR for hiding data into gray images using three bit XOR steganography system.It's time complexity is O(1) and it exceeds over existing methods.
- Khodaei and Faez presented a new adaptive data-hiding approach and it's based on LSB substitution and pixel value differencing methods.This approach is secure against the RS detection attack and steganalysis detector using SPAM features.

PROPOSED SYSTEM

1. The proposed system of steganography provides the user an interface such that one can perform the encryption and decryption operations using that interface with an ease.

2. There is no format barriers for the images that are being used in this process.

3. It means the algorithm is supported for any kind of image formats like .jpg , .jpeg ,.png , .gif , .tiff.

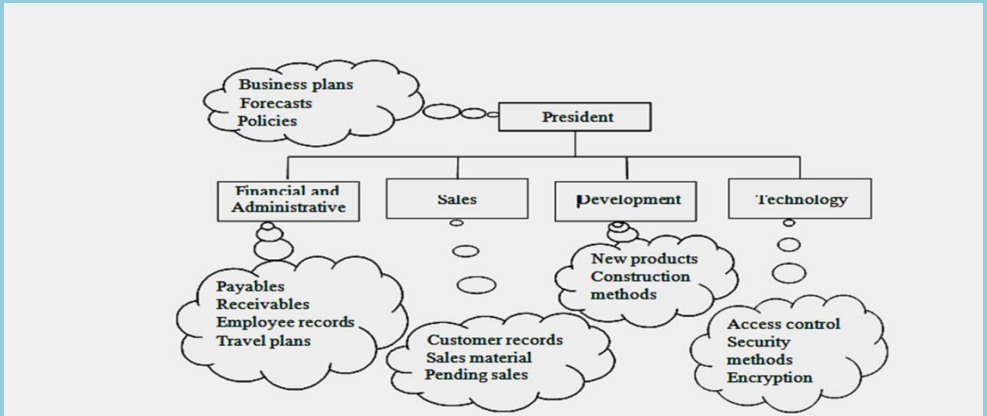
System Requirements

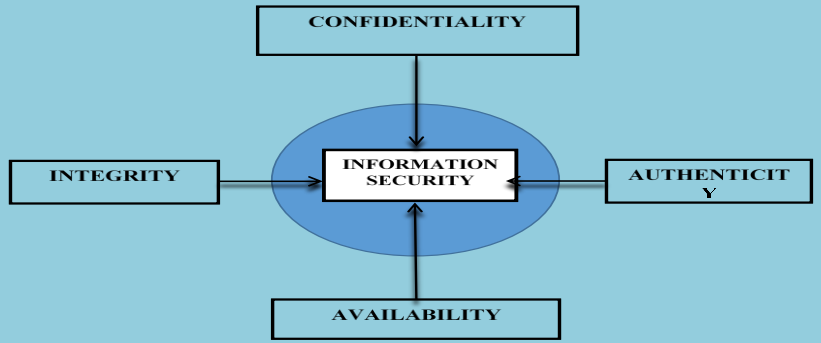
- Operating System : Windows10
- Coding Language : Core Java
- Java Distribution : VS Code, Net Beans

Hardware Requirements

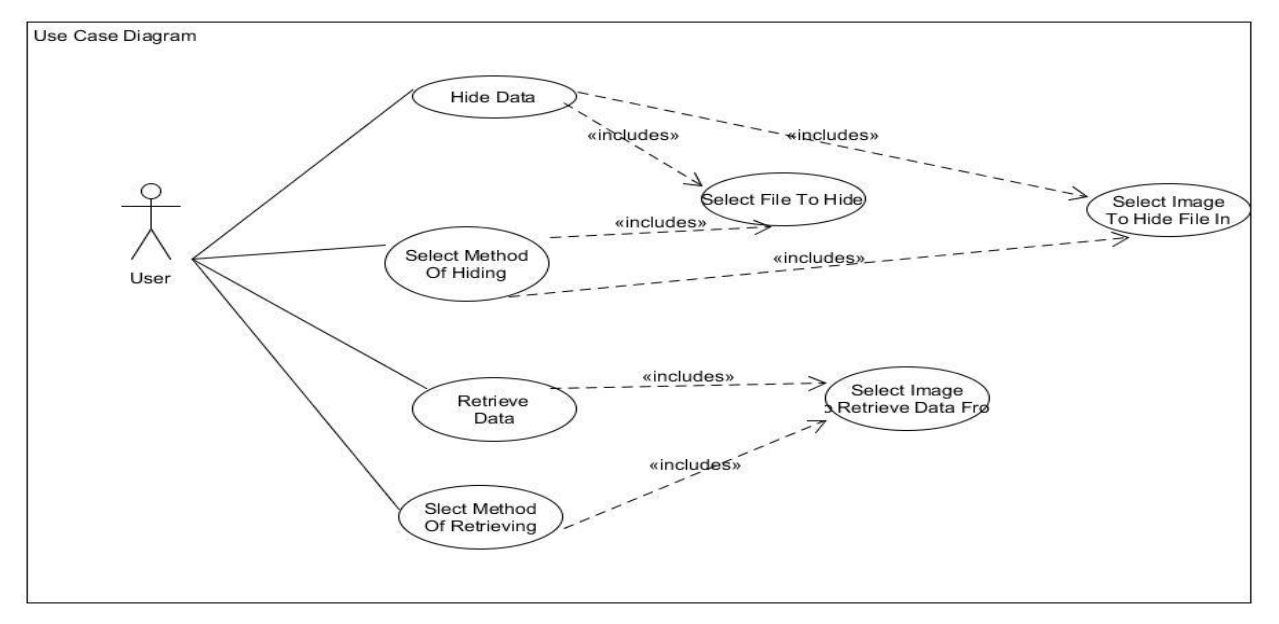
- System Type : Intel core I3
- RAM : 8GB
- Hard Disc : 1TB

Analysis Phase

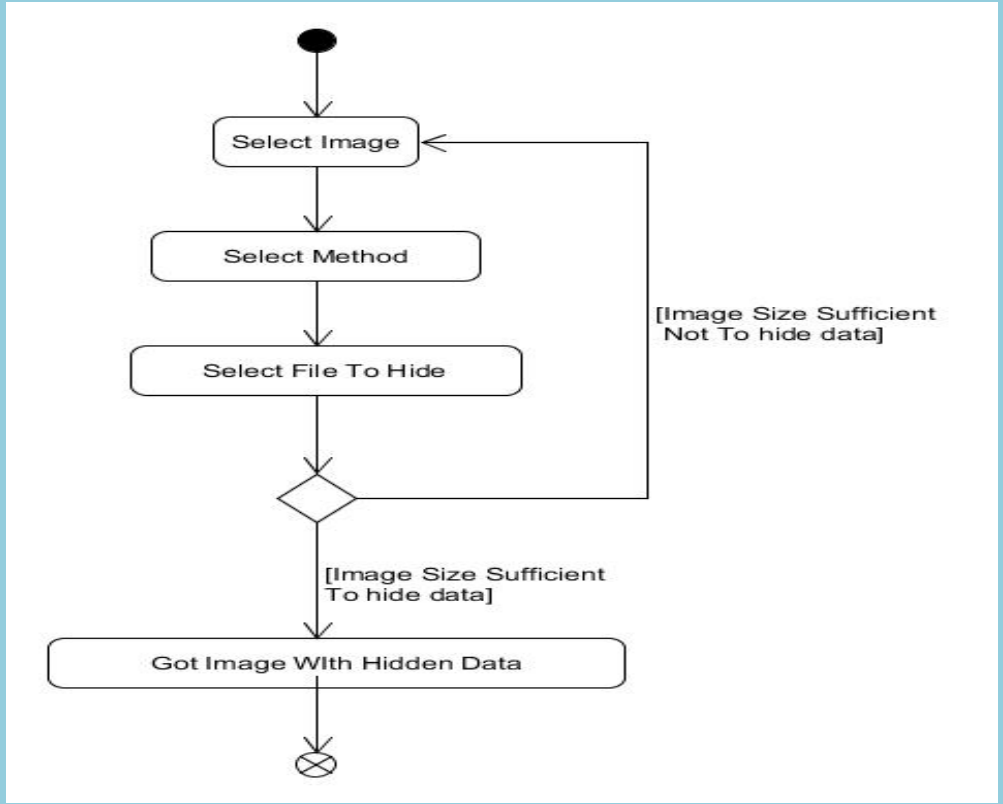




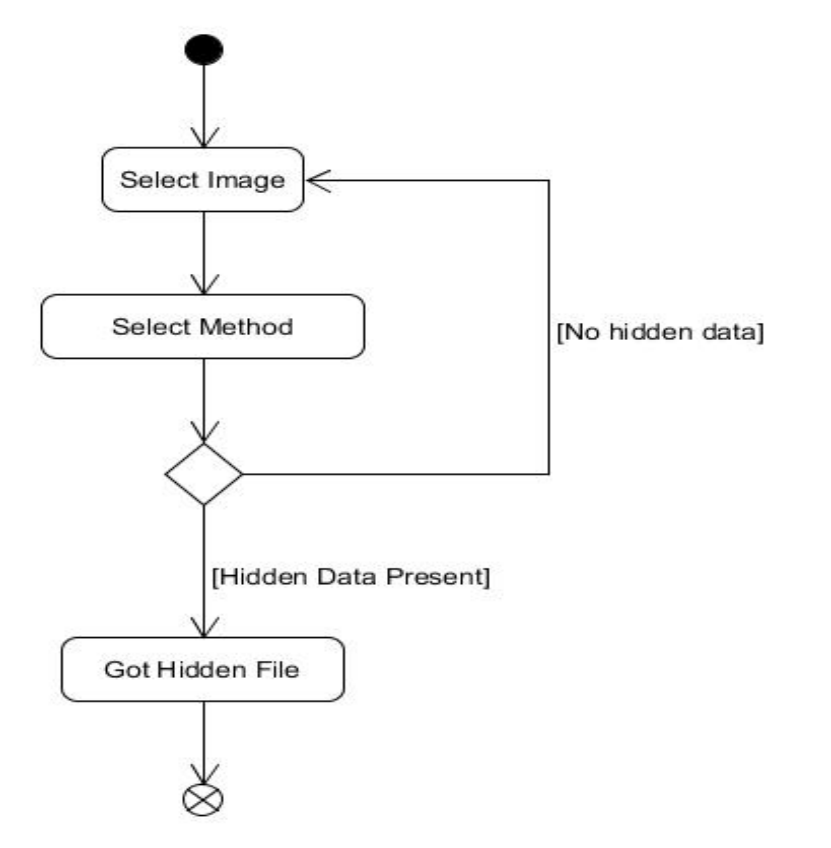
Design Phase



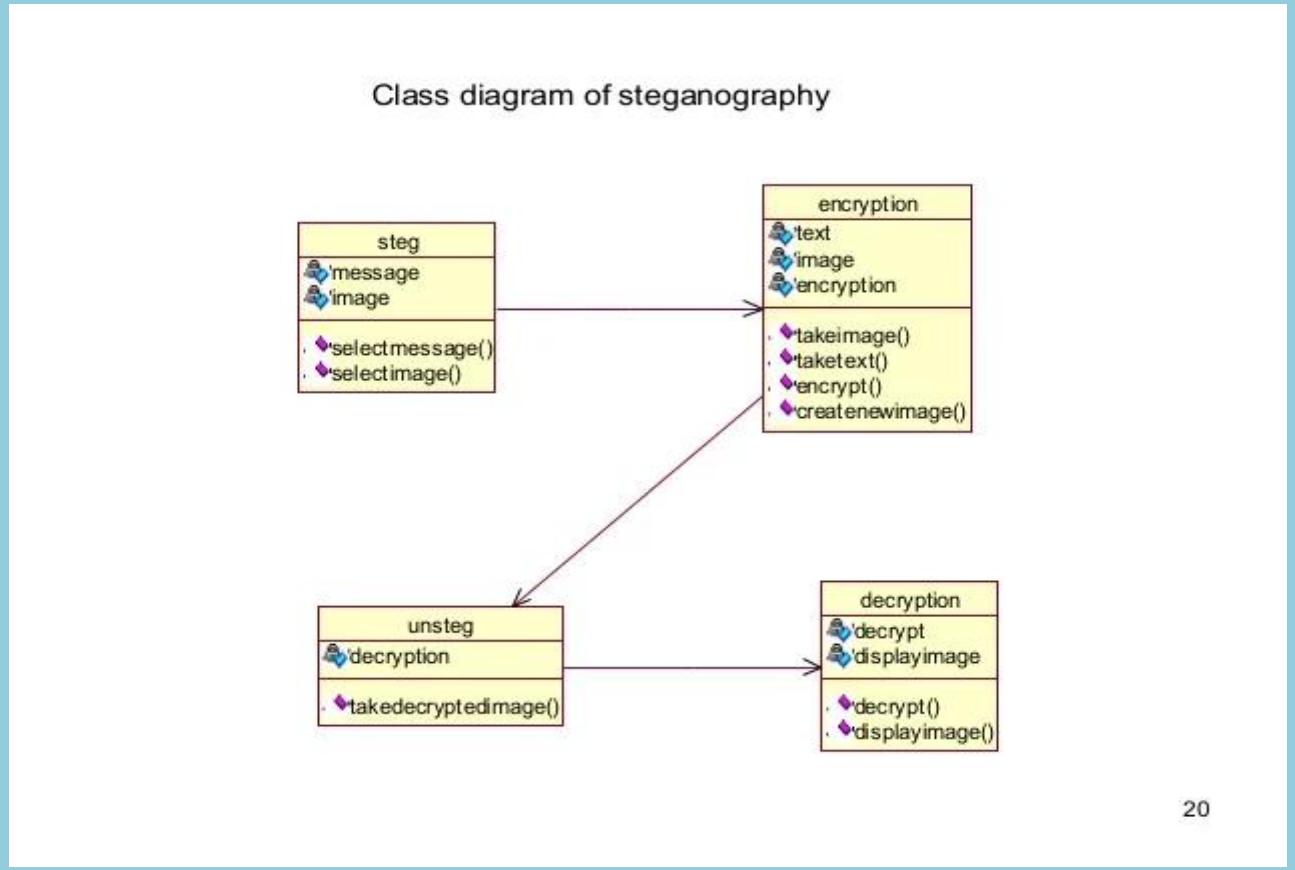
Use Case Diagram for proposed System



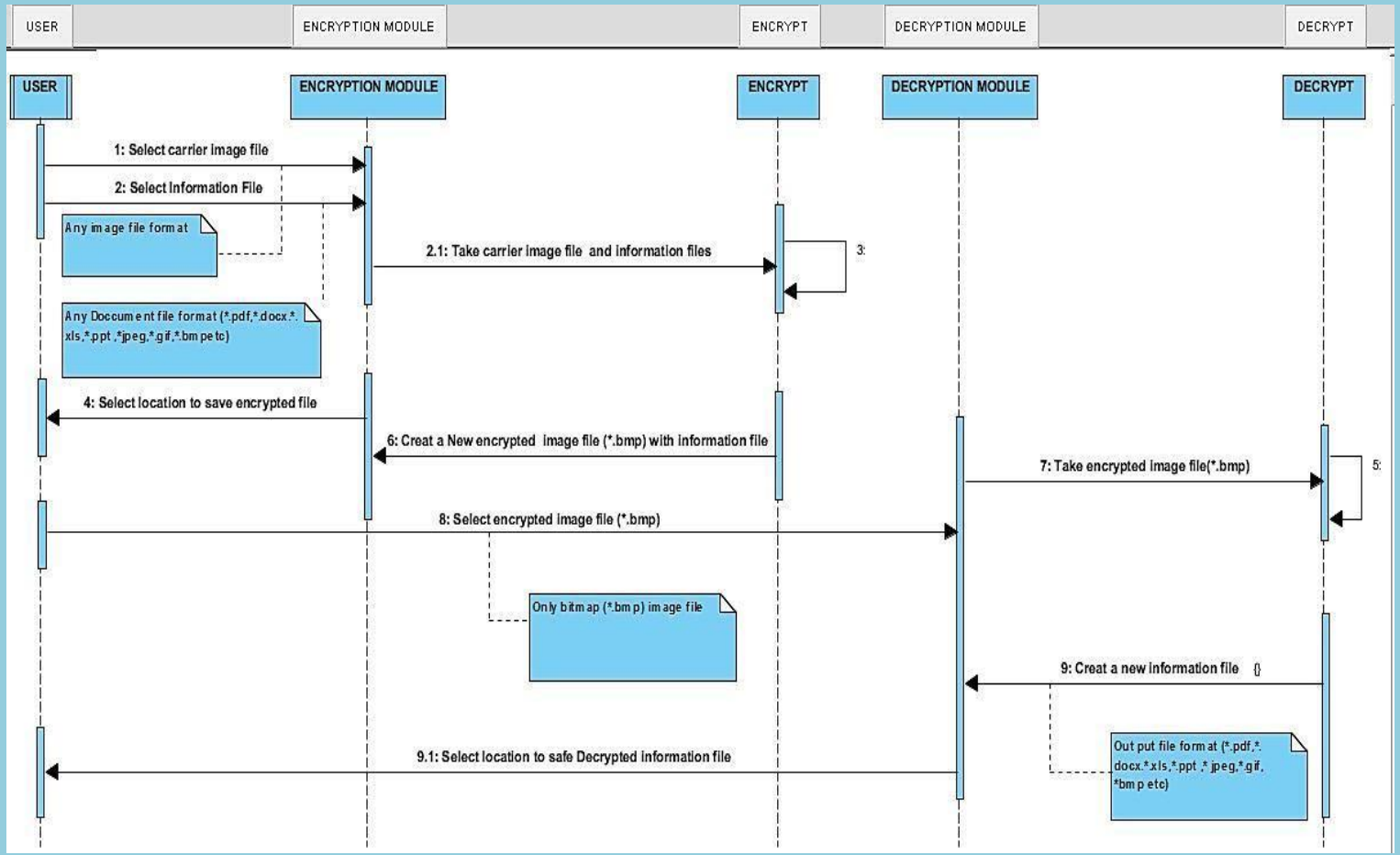
Activity Diagram for Encryption



Activity Diagram for Decryption



Class Diagram



Sequence Diagram

Implementation Phase

LSB ALGORITHM

STEP-1: Convert the pixel value to binary

STEP-2: Get the next bit of the message to be embedded

STEP-3: Create a variable **temp**

STEP-4: If the message bit and the LSB of the pixel are same, set temp = 0

STEP-5: If the message bit and the LSB of the pixel are different, set temp = 1

STEP-6: This setting of temp can be done by taking XOR of message bit and the LSB of the pixel.

STEP-7: Update the pixel of output image to input image pixel value + **temp**

EXAMPLE

Message to be embedded : " **200** "

Binary Form : 11001000

	The proportion of Red (R)	The proportion of Green (G)	The proportion of Blue (B)
Pixel 1	00101101	00011100	11011100
Pixel 2	10100110	11000100	00001100
Pixel 3	11010010	10101101	01100011

	The proportion of Red (R)	The proportion of Green (G)	The proportion of Blue (B)
Pixel 1	0010110**1**	0001110**1**	1101110**0**
Pixel 2	1010011**0**	1100010**1**	0000110**0**
Pixel 3	1101001**0**	1010110**0**	01100011

Optimal LSB Algorithm

- **Step-1**: Choose a cover image and a secret message to embed.
- **Step-2**: Convert the cover image and the secret message to binary format.
- **Step-3**: Determine the number of bits to be embedded per pixel (embedding rate) based on the cover image's size and the secret message's length.
- **Step-4**: Randomly select the pixels where the secret message will be embedded.
- **Step-5**: For each pixel, replace the least significant bits with the bits of the secret message.

Optimal LSB Algorithm

- **Step-6**: Apply error correction codes to the embedded message to ensure its integrity and accuracy.
- **Step-7**: Save the modified cover image with the embedded message.
- **Step-8**: To extract the embedded message, retrieve the least significant bits from the selected pixels of the modified cover image.
- **Step-9**: Decode the extracted message using the error correction codes.
- **Step-10**: Convert the binary message back to its original format.

Testing Phase

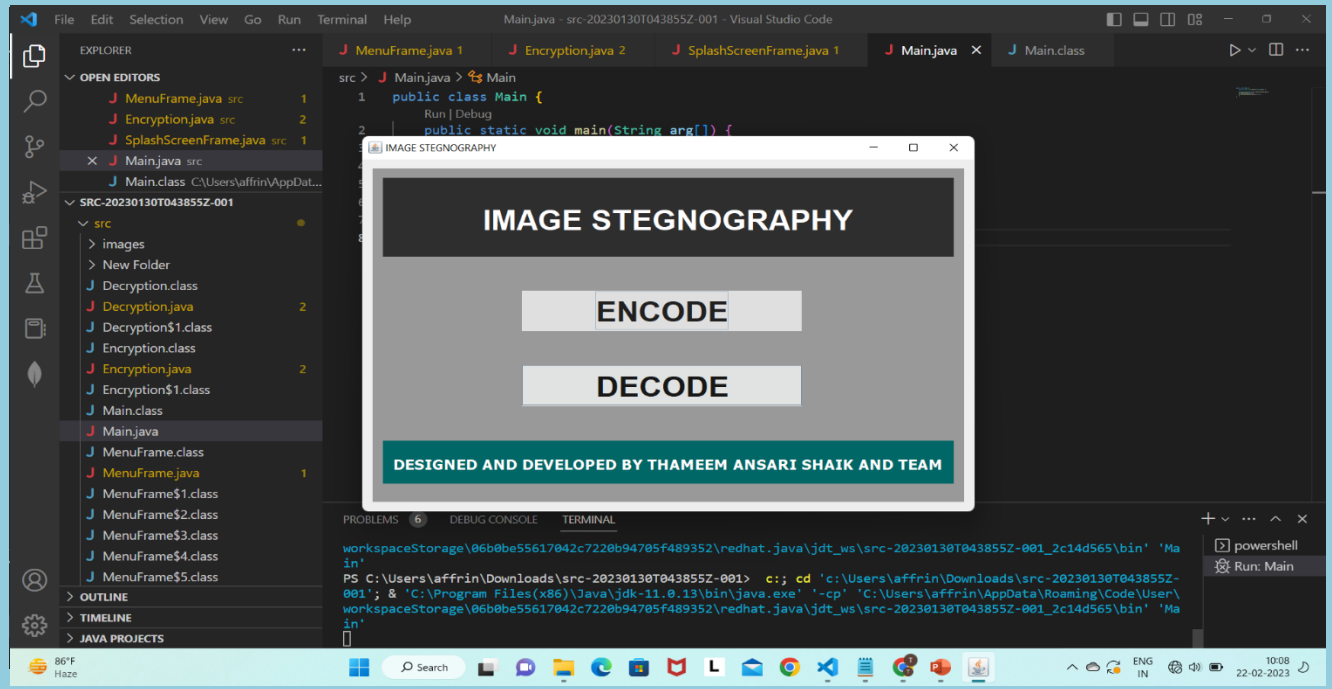


Fig1. Main Frame

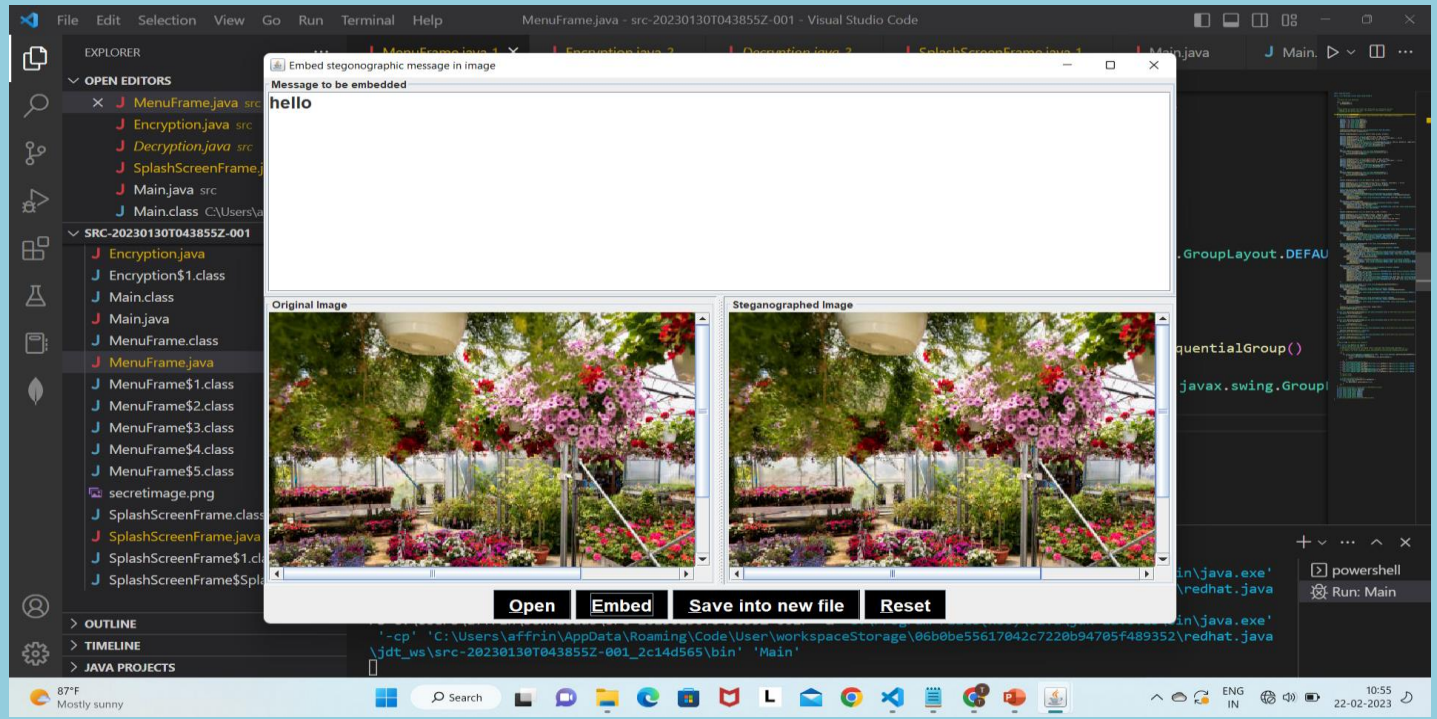


Fig2. Encryption Layout

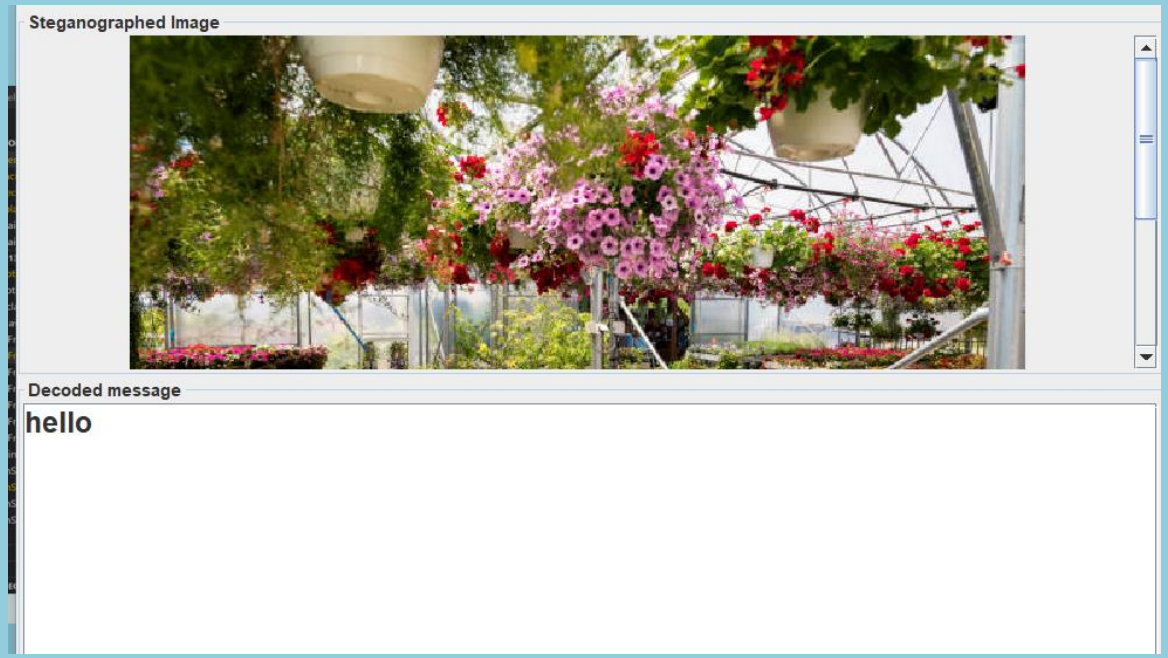
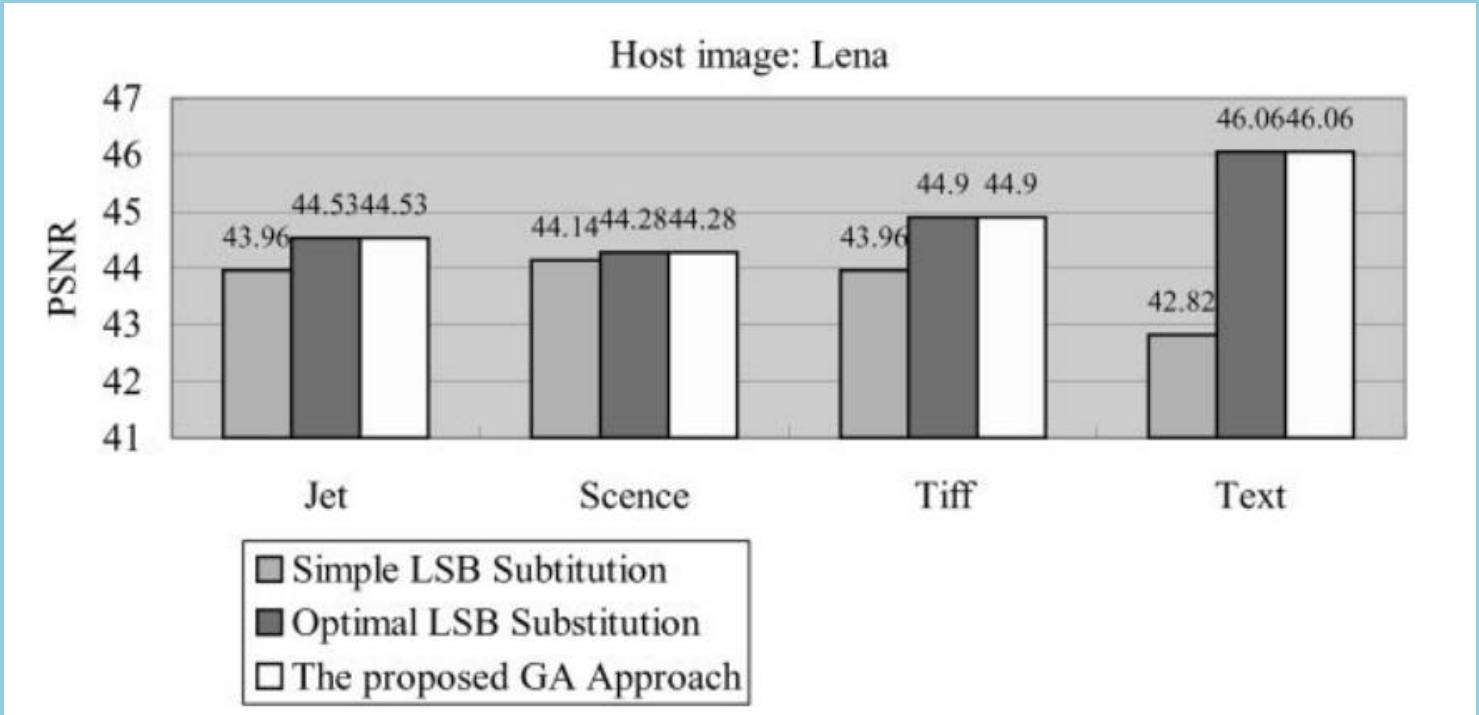


Fig3. Decryption Layout

Experimental Results Of LSB And Optimal LSB



References

- https://www.researchgate.net/publication/358997836_Analytical_Study_on_LSB-Based_Image_Steganography_Approach
- https://ieeexplore.ieee.org/document/9335027
- Arya, A., & Soni, S. (2018). Performance Evaluation of Secrete Image Steganography Techniques Using the Least Significant Bit (LSB) Method. vol, 6, 160-165.
- Rachael, O., Misra, S., Ahuja, R., Adewumi, A., Ayeni, F., & Mmaskeliunas, R. (2020). Image Steganography and Steganalysis Based on Least Significant Bit (LSB).

Thank You