IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

# Secure Share: A Web-Based Steganography Platform for Secure Communication

**Neethu K J**
Department of Computer Applications
Don Bosco College, Kozhikode, India

**Abstract:** *Secure Share is a web-based security platform designed to protect confidential communication through digital steganography and encrypted file sharing. The system enables users to securely hide secret text messages inside image files using the Least Significant Bit (LSB) encoding technique, ensuring that the hidden data remains visually undetectable. A PIN-based protection mechanism is integrated to restrict unauthorized message retrieval, providing an additional security layer during decoding.*

*The platform also supports secure user authentication, personal dashboards, file upload management, QR-based sharing, and automated email delivery of encoded images. A unique QR code is generated for each encoded image to simplify secure distribution without exposing the file location. Only authorized users with the correct image and PIN can extract the hidden message.*

*Secure Share is implemented using the Django web framework and Pillow for image processing, ensuring privacy, usability, and accessibility in real-world communication systems..*

**Keywords***:* Steganography, LSB Encoding, Secure Communication, Django, Image Processing, Data Security, QR Code

## I. INTRODUCTION

In the modern digital era, the exchange of information has become faster, easier, and more widespread than ever before. However, as communication technologies evolve, so do the risks associated with data theft, cyber-attacks, identity breaches, unauthorized access, and surveillance. Sensitive data such as passwords, confidential messages, financial information, and private conversations must be transmitted securely to prevent misuse. Traditional security techniques, like encryption, protect the content but also make it apparent that sensitive data is being transmitted, potentially attracting unwanted attention.

To address this limitation, **steganography** is employed—a technique in which confidential information is concealed within seemingly harmless media, such as digital images. This approach allows communication to appear normal while secretly carrying hidden data. **Secure Share** is a web-based platform developed to implement secure and user-friendly steganography. It enables users to hide secret messages inside images and retrieve them safely when required. The system integrates features such as user authentication, secure file uploads, QR-code-based sharing, PIN protection, and a modern user interface, providing a comprehensive solution for confidential digital communication.

## II. IMPLEMENTATION AND METHODS

The Secure Share system is implemented as a web-based application using the Django framework, following the Model–View–Template (MVT) architecture to ensure efficient data handling and user interaction. The backend is developed in Python and manages user authentication, session control, file processing, and request handling, while the frontend is built using HTML, CSS, and JavaScript to provide a responsive and user-friendly interface. Image processing operations are carried out using the Pillow library, enabling reading, modifying, and saving image files without noticeable distortion.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

### 2.1. Least Significant Bit (LSB) Technique

The core method used in Secure Share is the Least Significant Bit (LSB) steganography technique:

- Each image pixel contains RGB (Red, Green, Blue) values.
- The least significant bit of each value is modified to store secret data.
- These changes are minimal and do not affect the visual quality of the image.
- This method ensures that the hidden message remains undetectable to the human eye.

### 2.2. Encoding Methodology

- User uploads a cover image.
- Secret message is entered by the user.
- Message is converted into binary format.
- Optional PIN is assigned for security.
- Length of message is encoded for proper extraction.
- Binary data is embedded into the image using LSB technique.
- Encoded image is generated and stored.
- QR code is created for sharing the encoded file.

### 2.3. Decoding Methodology

- User uploads the encoded image.
- System verifies the PIN (if applied).
- Pixel values are extracted from the image.
- Least significant bits are read sequentially.
- Binary data is reconstructed into text format.
- Hidden message is displayed to the user.

### 2.4. Additional Security Measures

- PIN-Based Security: A user-defined PIN is required during decoding.
- QR Code–Based Sharing: Unique QR code generated for each encoded image.
- Email Integration: Encoded images can be sent securely via email.
- Authentication: User login and access control ensure only authorized access.

### 2.5. Performance Considerations

- Fast encoding and decoding using LSB technique.
- Minimal impact on image quality.
- Efficient for small to medium-sized messages.
- Low computational complexity.

### 2.6. Proposed System

Secure Share uses client–server architecture:

- **Client Side:** Web browser interface for user interaction.
- **Server Side:** Django application handling logic and processing.
- **Database:** Stores user credentials, encoded files, and logs.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

### 2.7. Backend Development

Responsible for:

- User authentication, session, and access management.
- Processing encoding and decoding requests.
- File upload and storage management.

### 2.8. Frontend Development

Designed using HTML, CSS, and JavaScript to provide a responsive and user-friendly interface.

### 2.9. Image Processing Module

Uses Pillow library for:

- Loading and reading image pixel data.
- Modifying pixel values for data embedding.
- Saving encoded images without noticeable distortion.

### 2.10. Database Management

A relational database such as SQLite or MySQL stores:

- User credentials.
- Encoded image metadata.
- Activity logs and history.

## III. RESULTS AND DISCUSSION

The Secure Share system successfully demonstrates secure communication using LSB steganography. Messages are embedded in images without visible distortion. Encoding and decoding are accurate and efficient, with message retrieval possible only with the correct PIN. QR code sharing and email integration enhance usability and security. Performance tests show minimal processing time, low computational overhead, and stable operation for small to medium-sized messages. The system provides a simple and intuitive interface, confirming its effectiveness and practicality for confidential digital communication.

### 3.1. Module Description

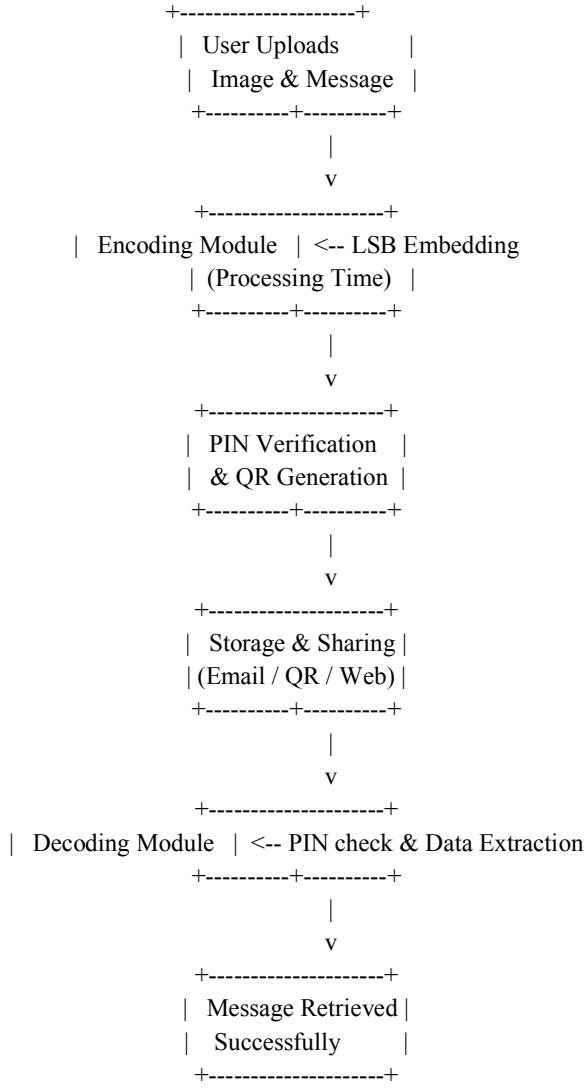
Secure Share is structured into multiple functional modules to ensure clarity, security, and maintainability:

- **User Authentication Module:** Handles registration, login, logout, and session management. Ensures only authorized users can access encoding, decoding, and file management features.
- **Image Encoding Module:** Embeds secret messages in images using LSB steganography. Optional PIN protection ensures only authorized access. Encoded images can be shared via QR code or email.
- **Image Decoding Module:** Retrieves hidden messages from images after verifying user credentials and PIN. Maintains privacy and integrity of extracted messages.
- **QR Code Generation Module:** Generates a secure QR code for each encoded image, allowing safe sharing without revealing file paths.
- **File Management and Dashboard Module:** Provides a centralized interface to view, organize, and track uploaded and encoded files efficiently.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

### 3.2. System performance

**Fig. 1 System Performance Diagram**




## IV. CONCLUSION

Secure Share successfully demonstrates a secure and efficient method for confidential communication using image steganography. The platform integrates authentication, PIN-based protection, QR code sharing, and email delivery to ensure privacy, usability, and accessibility. The modular architecture enhances maintainability and scalability. Testing and security measures confirm reliable performance and data integrity. Secure Share meets its objectives and provides a practical solution for secure digital communication, with scope for future enhancements.

## V. ACKNOWLEDGMENT

The author would like to express sincere gratitude to the Department of Computer Applications, Don Bosco College, Mampetta, Kozhikode, for providing the necessary support and academic environment for this work. The author also

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 6, March 2026 Impact Factor: 8.2

thanks the faculty members for their valuable guidance and encouragement during the development of the Secure share: a web-based steganography platform for secure communication .Finally, appreciation is extended to colleagues, family, and friends for their continuous support and motivation throughout this research.

## REFERENCES

[1] N. Provos and P. Honeyman, "Hide and Seek: An Introduction to Steganography," *IEEE Security & Privacy*, vol. 1, no. 3, pp. 32–44, 2003.
[2] M. Johnson and S. Jajodia, "Exploring Steganography: Seeing the Unseen," *IEEE Computer*, vol. 31, no. 2, pp. 26–34, 1998.
[3] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2nd ed., Morgan Kaufmann, 2007.
[4] Django Software Foundation, "Django Documentation," [Online]. Available: https://docs.djangoproject.com/
[5] A. Clark, "Python Imaging Library (Pillow) Documentation," [Online]. Available: https://pillow.readthedocs.io/
[6] M. Kharate and S. Shinde, "Implementation of LSB Steganography for Secure Data Transmission," *Int. J. Computer Applications*, vol. 69, no. 6, pp. 25–29, 2013.
[7] R. Chandramouli and N. Memon, "Analysis of LSB Based Image Steganography Techniques," *Proc. SPIE – Security and Watermarking of Multimedia Contents II*, vol. 4675, pp. 24–34, 2002.