Stegame's behind-the-scenes

This document serves as the initial draft planner for the software development life cycle (SDLC) of Stegame. It details the tools that will be used, the development stages, and the intricacies of its creation.

Development Stages:

1. Planning and Design:(already provided in the previous document)
   - Defined project requirements, including functionality and security.
   - Designed system architecture, user interface, and workflows.

2. Backend Development:
   - Python with Flask or Django for backend development.
   - Encoding, decoding, and encryption logic implementation using OpenCV for image processing and Python Cryptography for encryption.

3. Frontend Development:
   - HTML, CSS, and JavaScript for frontend development.
   - Working on the development of the user interface for uploading images and entering messages.

4. Integration:
   - Integration of frontend and backend components to ensure seamless operation.
   - Testing the integration thoroughly to identify and fix any issues.

5. Testing:
   - Performing unit testing and integration testing to ensure the application functions correctly.
   - Testing for security vulnerabilities and addressing any issues found.

6. Deployment:
   - Deployment of the application to a web server, such as Heroku or AWS, for public access.
   - Configuration of the server for security and scalability.

7. Maintenance:
   - Regularly updating the application to address security vulnerabilities and improve functionality.

- Monitoring the application for performance and security issues and resolving them promptly.

Tools Required:

Programming Languages: Python for backend development, HTML/CSS/JavaScript for frontend development.

Web framework: Flask or Django for developing the web application.

Image Processing Library: OpenCV for image manipulation and processing.

Encryption Library: Python Cryptography or PyCrypto for encrypting and decrypting messages.

Chaos Library: Libraries such as NumPy for generating chaotic random numbers using the logistic map.

Version Control: Git for version control and collaboration.

Deployment: Platforms like Heroku or AWS for deploying the web application.