Software Requirements Specification

for

TruCrypt

Version 1.0 approved

Prepared by Manish Jangid

DR. D.Y. Patil Institute of Management and Research

21/11/2023

Table of Contents

1. Introduction

1.1 Purpose

The primary purpose of this Software Requirements Specification (SRS) document is to establish a comprehensive and detailed understanding of the requirements for the development of an Image Steganography system in Java. The document aims to serve as a foundation for the entire software development life cycle, providing guidance for developers, testers, project managers, and other stakeholders involved in the project.

1.2 Document Conventions

All elements of this document, including code snippets and user documentation, will adhere to the conventions of the Java programming language. Additionally, standard documentation practices such as Javadoc will be employed for code comments to enhance code readability and maintainability.

1.3 Intended Audience and Reading Suggestions

This document is intended for a diverse audience, including developers, testers, and project stakeholders. Developers will utilize this document as a guide for implementation, testers for validation of requirements, and stakeholders for understanding the project scope and functionality.

1.4 Product Scope

The Image Steganography system aims to provide a sophisticated solution for secure information exchange through image files. By leveraging Java's capabilities, the system will offer users a seamless experience for embedding and extracting hidden data within images. The scope extends to the development of a feature-rich standalone application with a strong emphasis on security and ease of use.

1.5 References

1)www.Google.com
2)www.chat.openai.com
3) www.geeksforgeeks.org
4)www.javaworld.com
5)www.researchgate.com
6)www.sliideshare.com
7)www.codecreator.com
8) www.Github.com
9)www.1000projects.org
10)www.youtube.com

2. Overall Description

2.1 Product Perspective

The Image Steganography system will function as a standalone application, independent of external systems. It will manipulate image files to embed and extract data, ensuring a user-friendly experience without dependencies on other software or services.

2.2 Product Functions

The core functions of the system include:

- **Embedding**
  Allow users to embed text data into image files.
  Utilize advanced steganographic techniques to ensure the hidden data remains undetectable. Preserve the visual quality of the original image after embedding.
- **Extraction**
  Enable users to extract hidden data from images manipulated by the system.
  Implement algorithms to accurately retrieve the embedded data without loss or corruption.

2.3 User Classes and Characteristics

The system will cater to the following user classes:

- **End Users**
  Individuals or organizations seeking a secure means of communication through covert image data.
- **Developers**
  Those responsible for maintaining, enhancing, and extending the functionality of the Image Steganography system.

2.4 Operating Environment

The system will be developed using Java and will be compatible with any platform supporting the Java Runtime Environment (JRE). This ensures cross-platform usability, allowing users to run the application on various operating systems.

2.5 Design and Implementation Constraints

The following constraints will be considered during the development process:

- **Programming Language:** The system will be implemented in Java.

- **Compatibility**: The system should be compatible with common image formats, including JPEG, PNG, and BMP.

2.6 User Documentation

Comprehensive user documentation will be provided, including:

- User manuals detailing the steps for embedding and extracting hidden data.
- Technical documentation for developers, including code documentation and architecture diagrams.

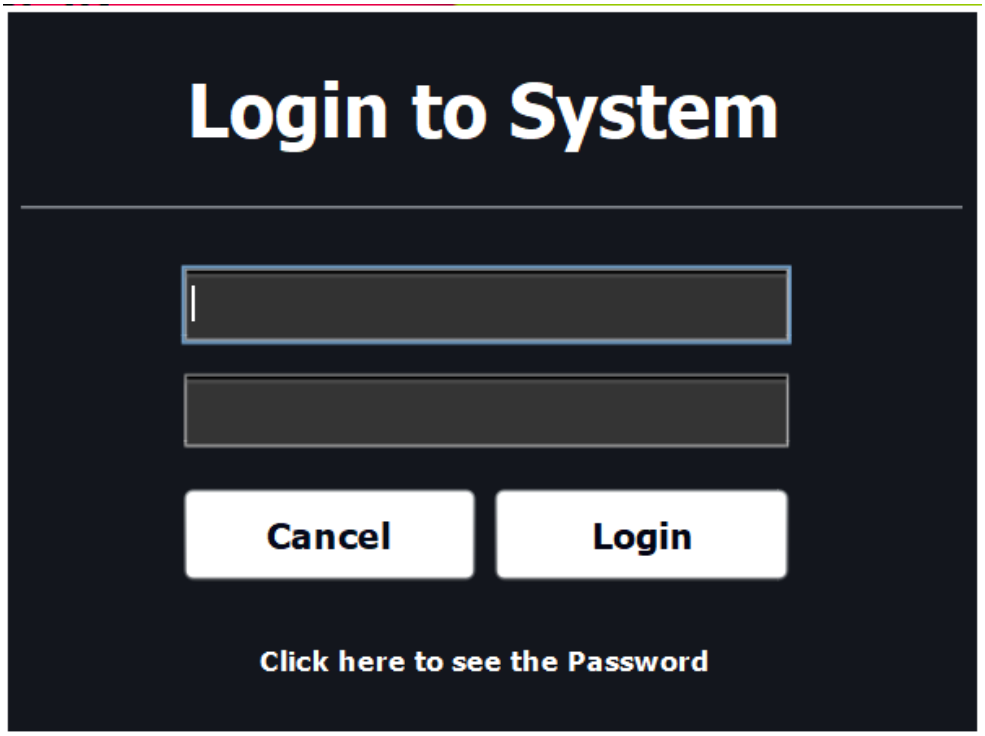
2.7 Assumptions and Dependencies

- **Assumption : Users have a basic understanding of steganography concepts.**
- **Dependency:** The Availability of a Java Runtime Environment on the user's machine for proper execution of the application.

3. External Interface Requirements

3.1 User Interfaces

The system will feature an intuitive Graphical User Interface (GUI) providing users with a seamless and visually appealing experience. The GUI will include controls for embedding, extracting, and configuring steganographic parameters.




TRUCRYPT
DYPIMR
STEGANOGRAPHY
44%

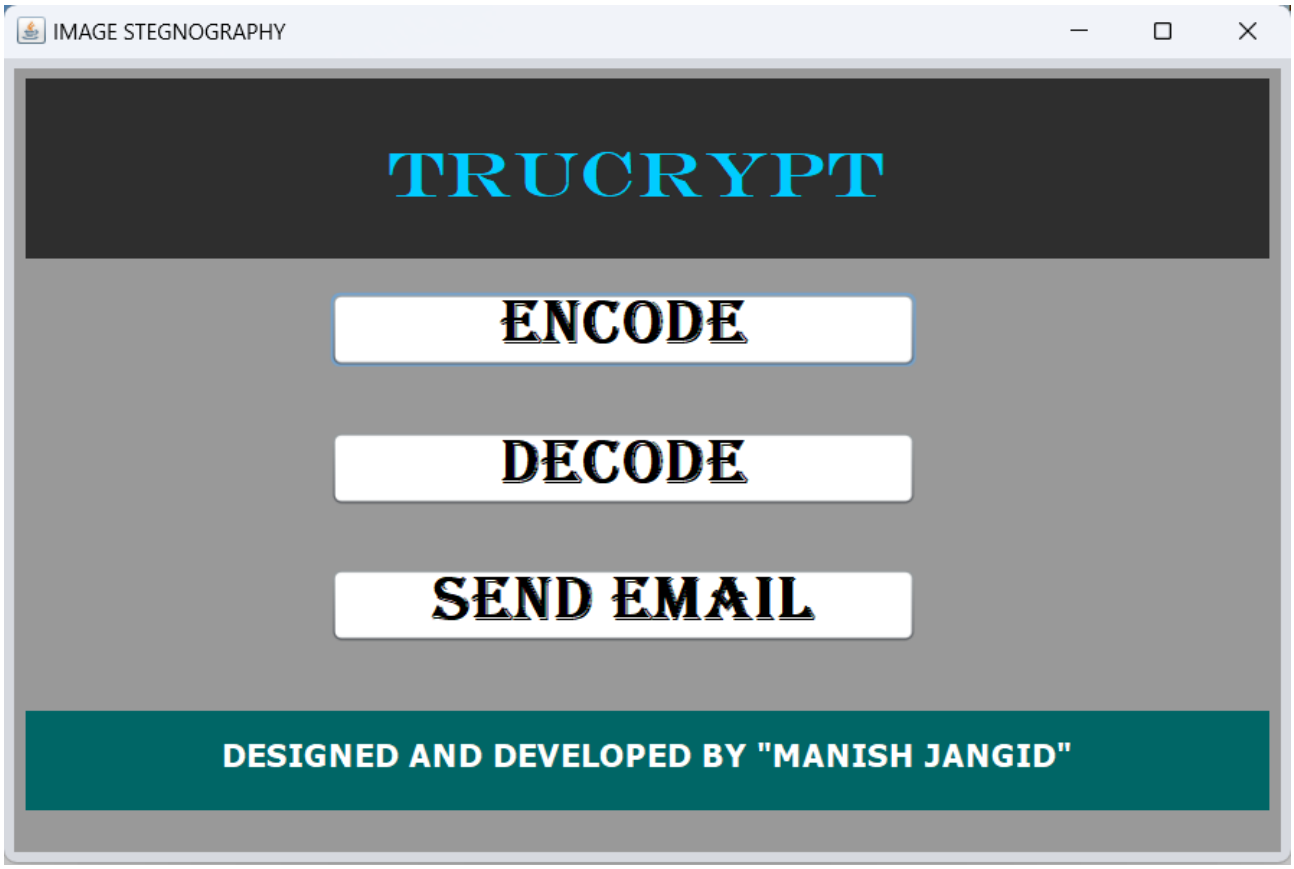


IMAGE STEGNOGRAPHY
TRUCRYPT
ENCODE
DECODE
SEND EMAIL
DESIGNED AND DEVELOPED BY "MANISH JANGID"

3.2 Hardware Interfaces

The system will interact with standard input/output devices, including:

- **Input:** Keyboard and mouse for user interaction.
- **Output:** Display for presenting the GUI and visual feedback.

3.3 Software Interfaces

The system will interface with the following software components:

- **Java Runtime Environment (JRE**): Required for executing the Java-based application.
- **Image Processing Libraries:** Utilized for efficient manipulation of image files.

3.4 Communications Interfaces

No external communication interfaces are required as the system functions as a standalone application.

4. System Features

4.1 System Feature 1 : Advanced Data Embedding

4.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

4.1.2 Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

4.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use "TBD" as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The embedding and extraction processes should be completed within a reasonable time frame for images of varying sizes.

5.2 Safety Requirements

The system should not compromise the visual integrity of the original image during the embedding process.

5.3 Security Requirements

Employ robust cryptographic techniques to secure the hidden data.
Implement measures to prevent unauthorized access to the embedded data.

5.4 Software Quality Attributes

- **Code Maintainability**

Codebase should be well-documented using Javadoc and adhere to established coding standards.

- **User Interface Usability**

The graphical interface should be intuitive, providing clear navigation and feedback to users.

5.5 Business Rules

Compliance with legal regulations regarding the use of steganography for communication.

6. Other Requirements

No Additional Requirements.