

ANDROID SECURE COMMUNICATION SYSTEM USING STEGANOGRAPHY



Kleona Binjaku
Elinda Kajo Meçe

DAAD Workshop "Cooperation at Academic Informatics Education across Balkan Countries and Beyond"
Primošten, Croatia, 2nd - 8th September 2018

Content

- Introduction
- Application Development Phases
- Demonstration
- Conclusions

Introduction

Why secure application for mobiles?

- we mostly use mobiles for communication
- information we share is no more secure

Why steganography?

- the third party doesn't even know if something is hidden
- interesting way of hiding information.

Why Android?

- 87%** of users prefer Android operating system in their mobiles
- easier to program

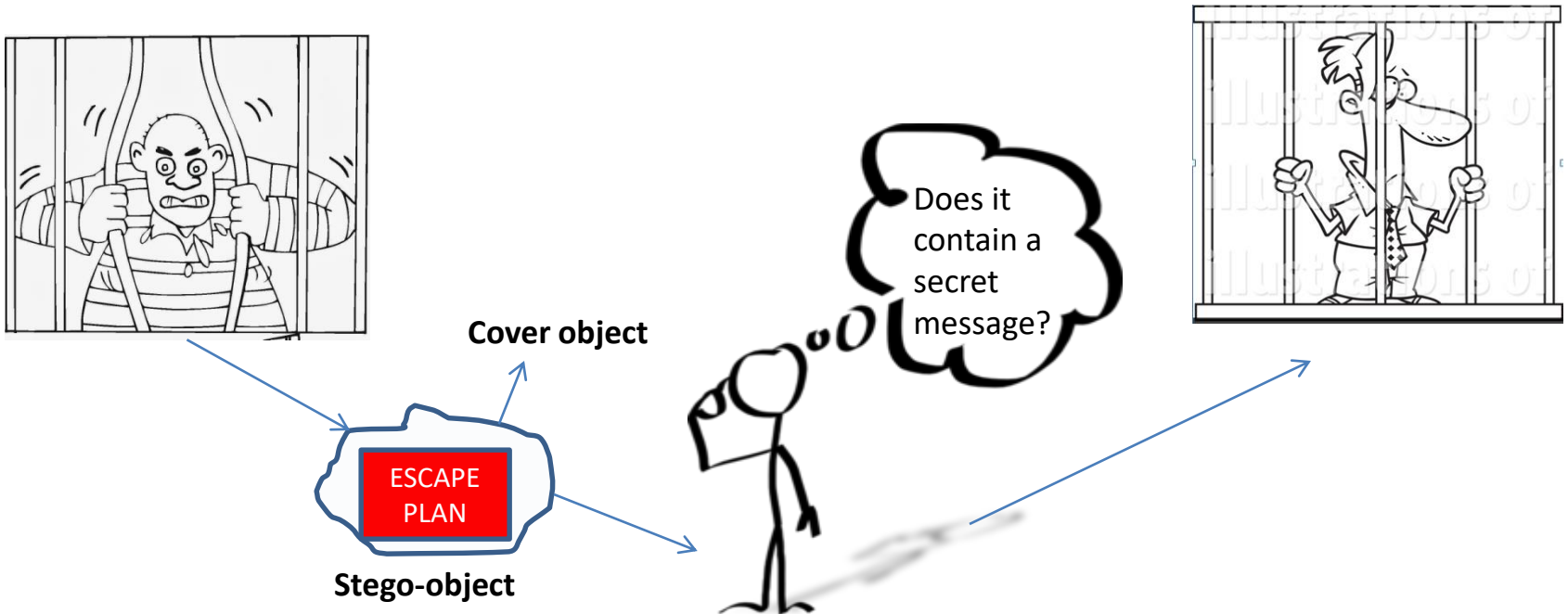
Steganography-Theoretical background

Facts

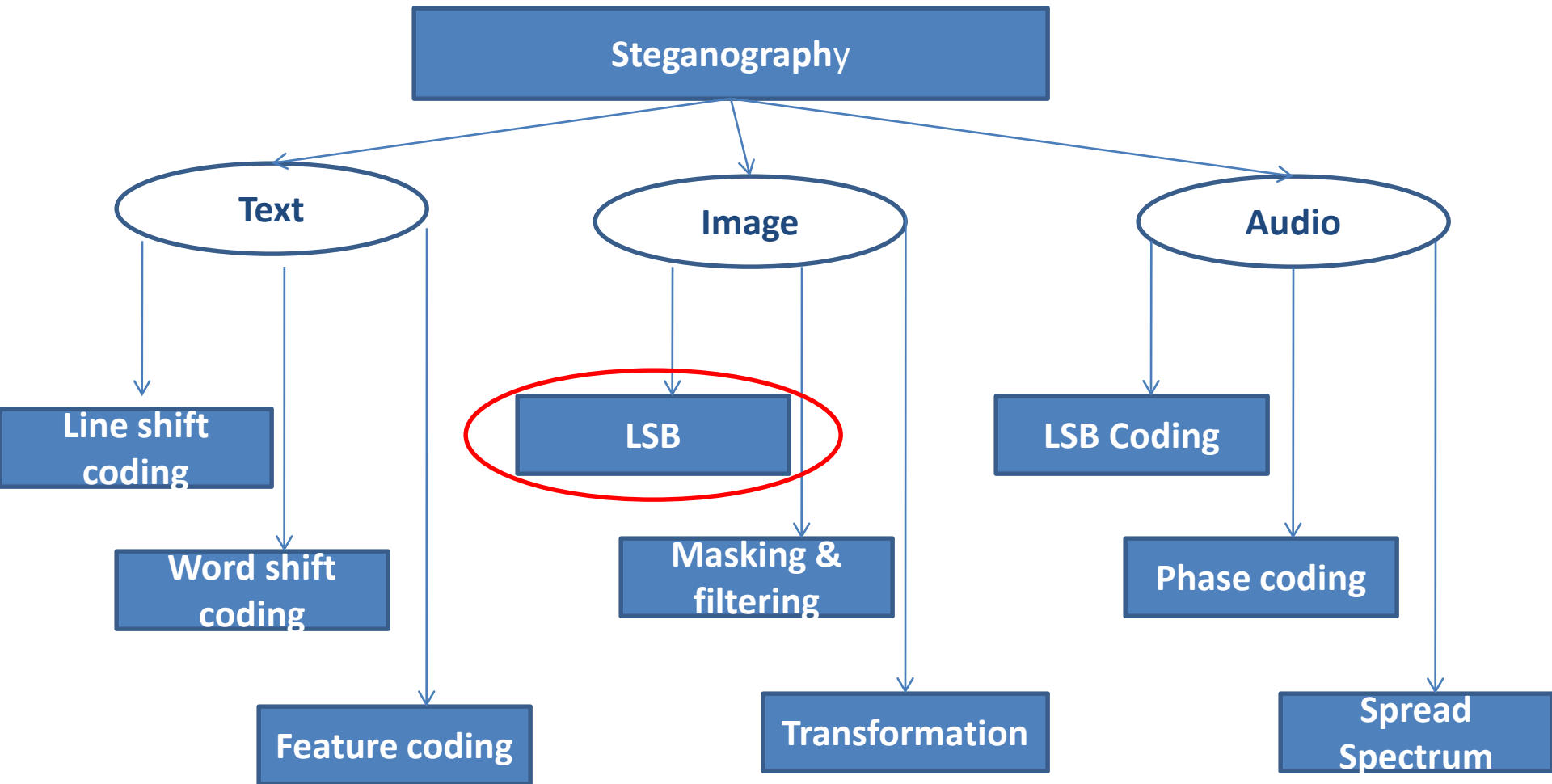
Since 1499
Used in World War

Goal

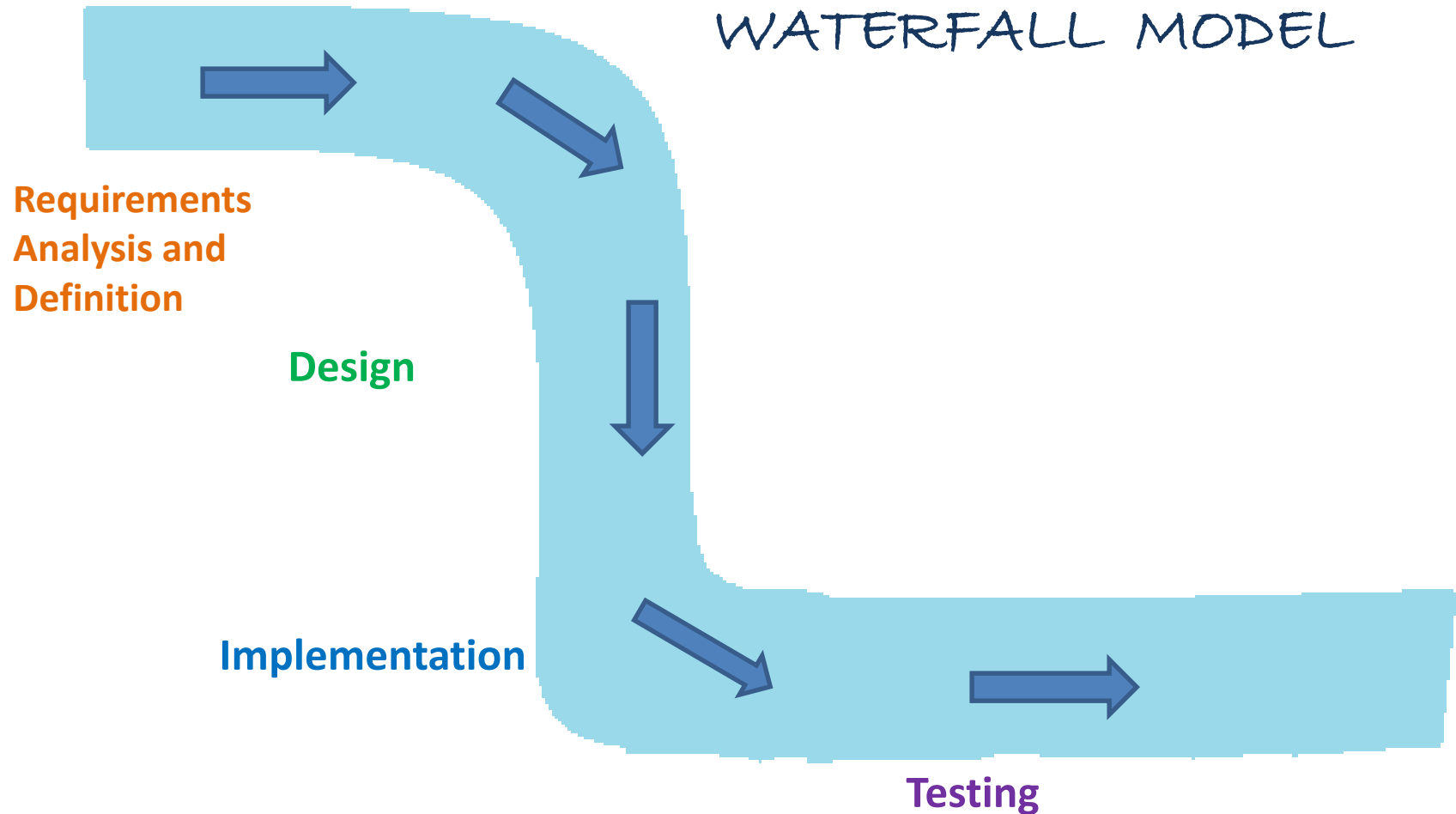
to hide the information in a digital carrier, so that it is undetectable by third party



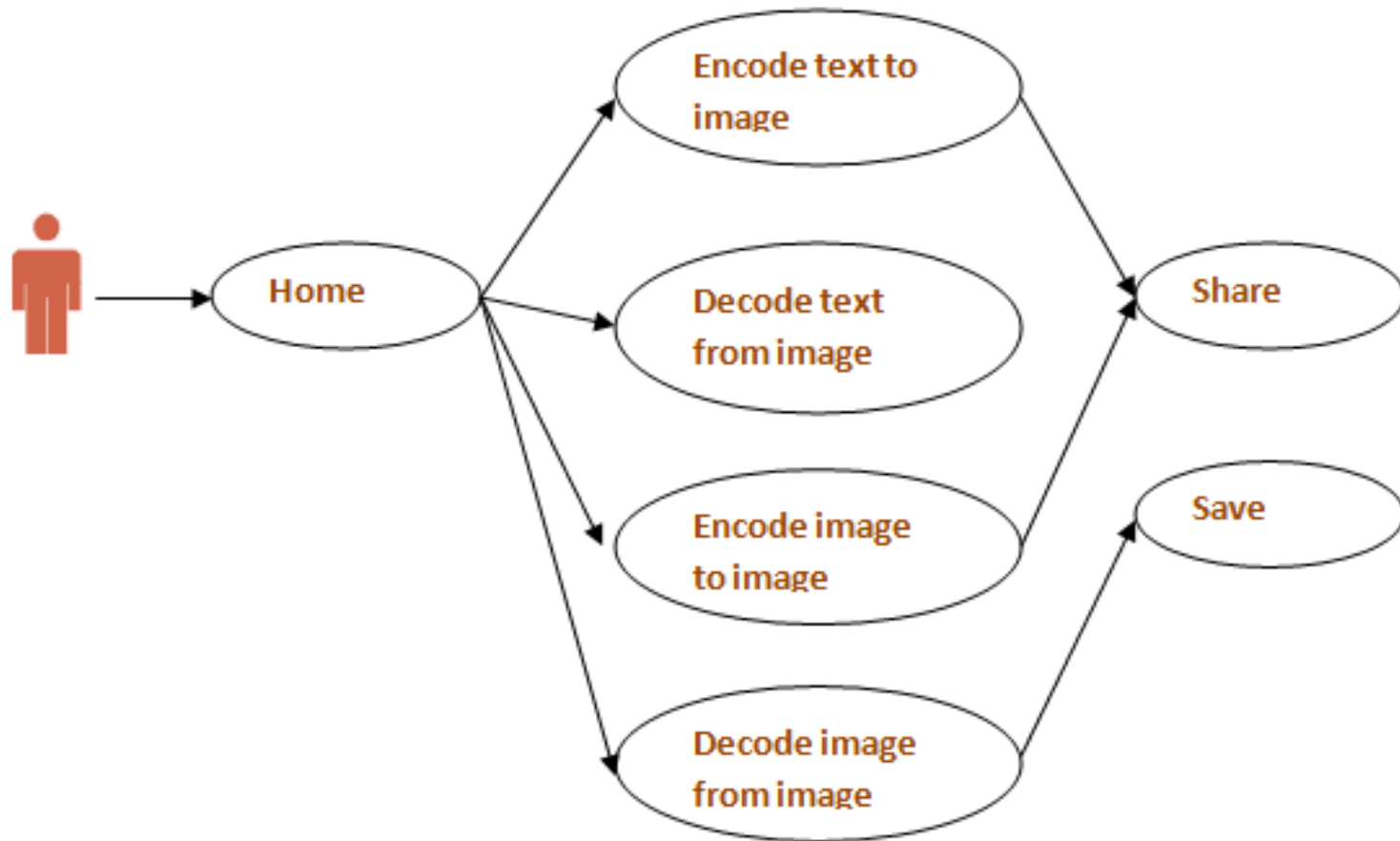
Types and methods for steganography



Software development phases



use case diagram



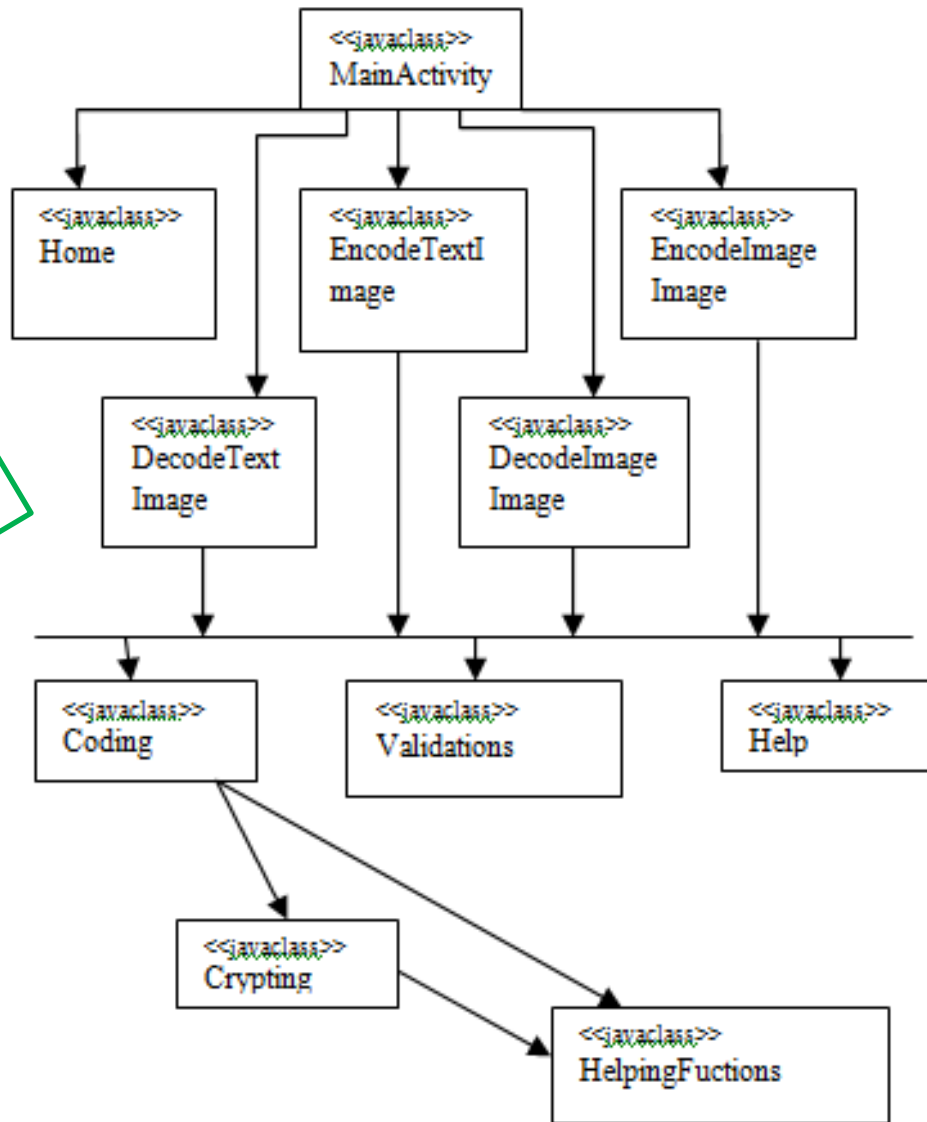
Requirements
Analysis and Definition

Design

Implementation

Testing

class diagram



One package
11 Java classes

Requirements
Analysis and Definition

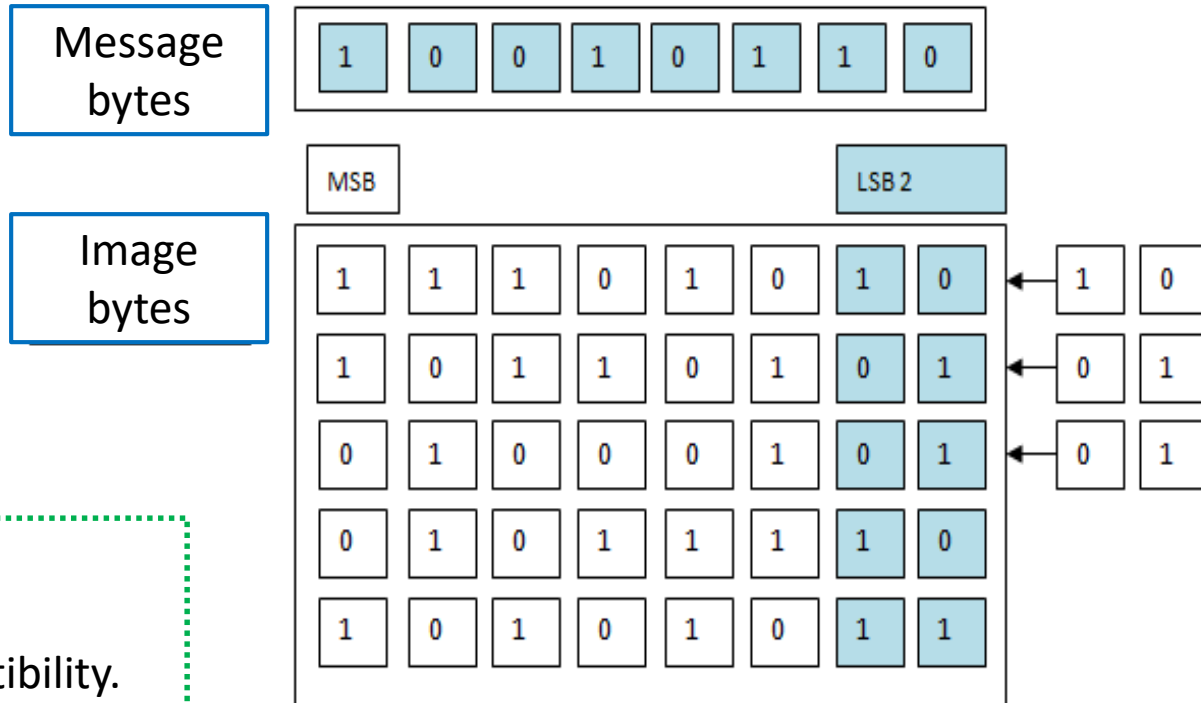
Design

Implementation

Testing

Steganography technique

LSB2



Why LSB2:

- high capacity
- high imperceptibility.

Requirements
Analysis and Definition

Design

Implementation

Testing

Project specification

ANDROID

STUDIO



Application name: **CryptoStego**

Minimum Required SDK: **API 23**

Target SDK: **API 25**

Screen size: **Adaptive**

Screen mode: **Portrait**

Online/Offline work: App works perfectly in offline mode. But, internet is needed to share images.

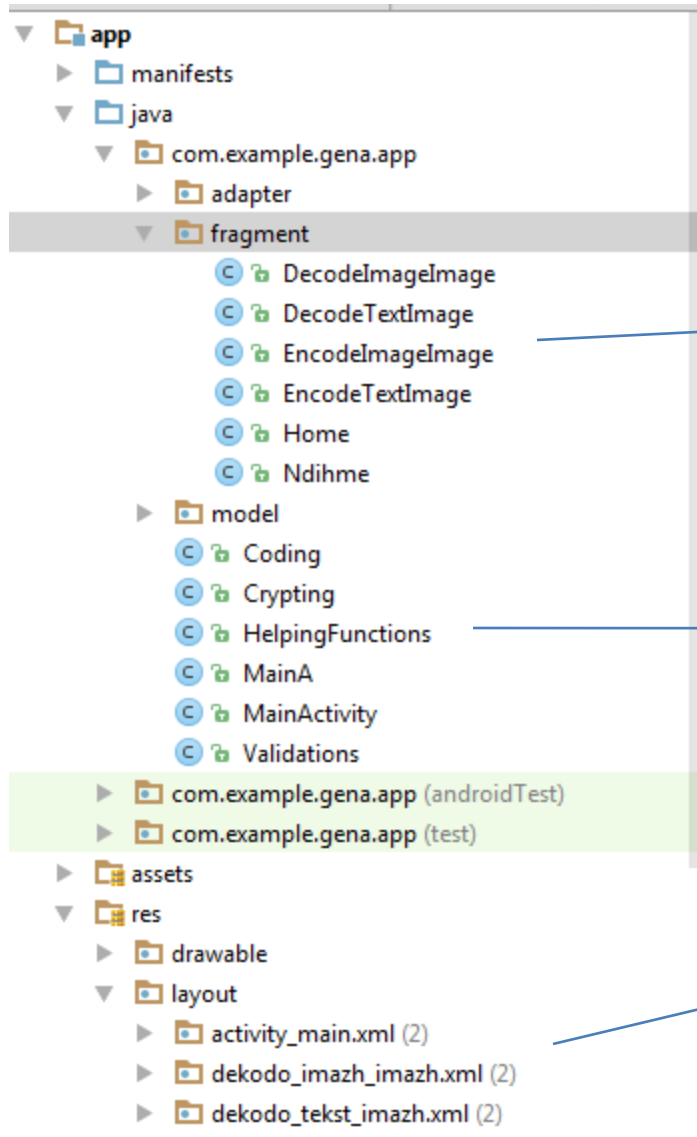
Requirements
Analysis and Definition

Design

Implementation

Testing

Application structure in Android Studio



Fragments for
different screens

Java classes

UI Layouts

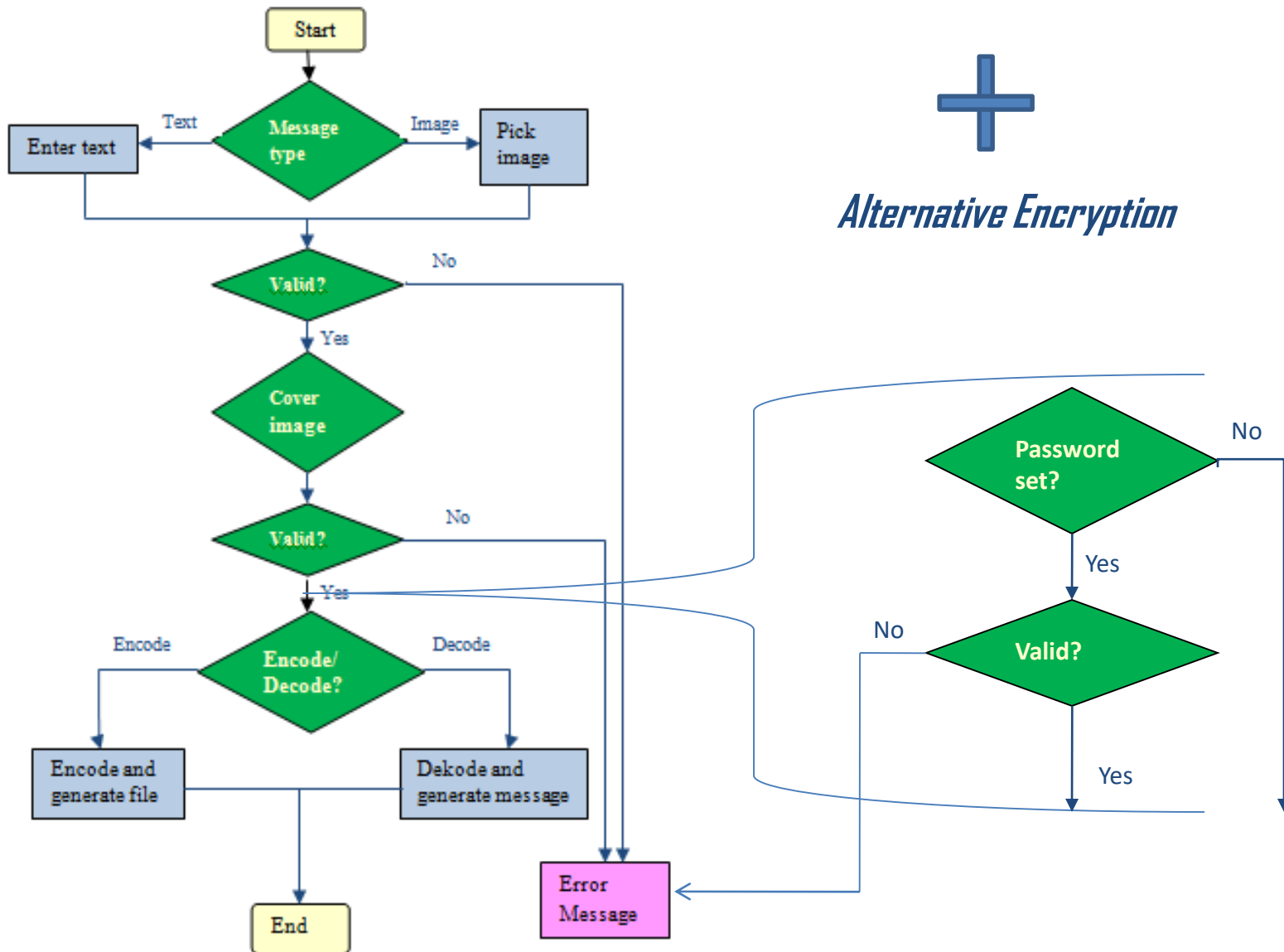
Requirements
Analysis and Definition

Design

Implementation

Testing

Flow chart



Requirements
Analysis and Definition

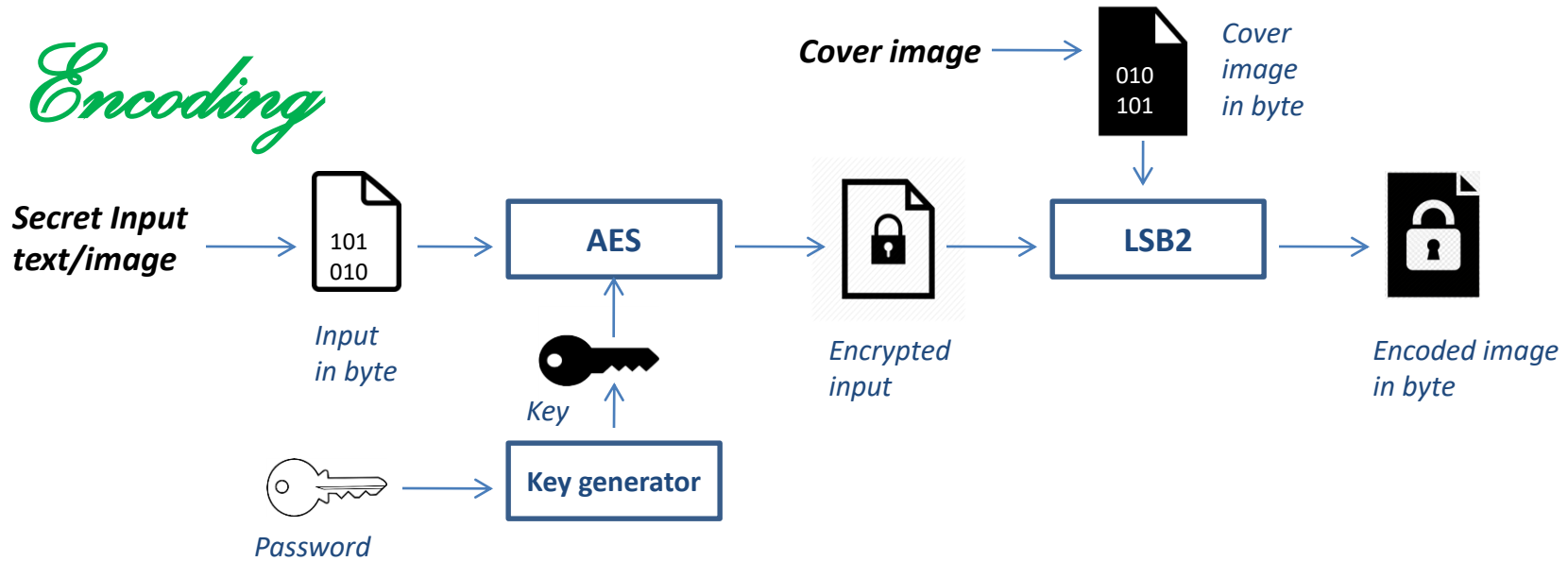
Design

Implementation

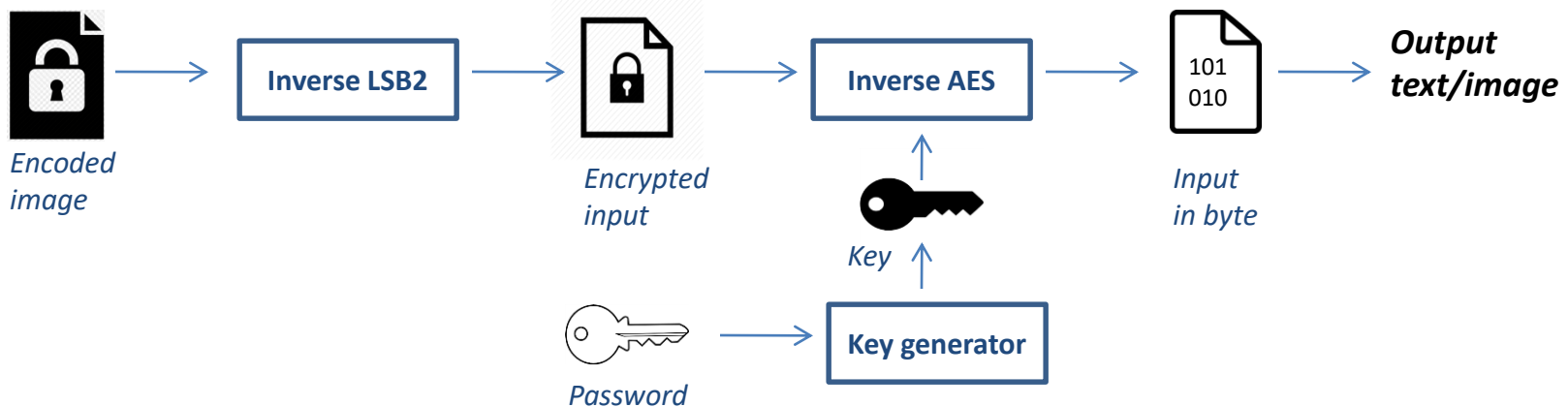
Testing

Full Encoding / Decoding Procedure

Encoding



Decoding



Requirements
Analysis and Definition

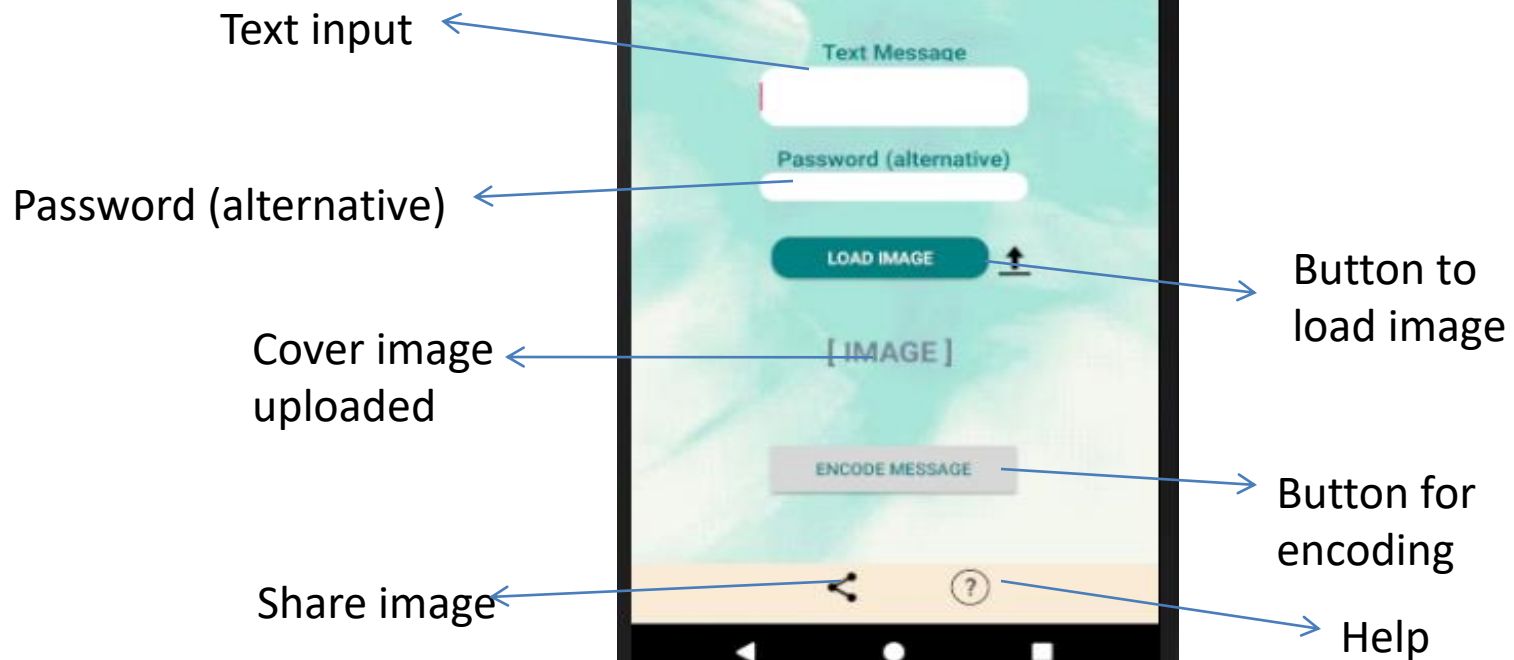
Design

Implementation

Testing

Demonstration

Encode text to image



Requirements
Analysis and Definition

Design

Implementation

Testing

Demonstration(2)

Decode text from image

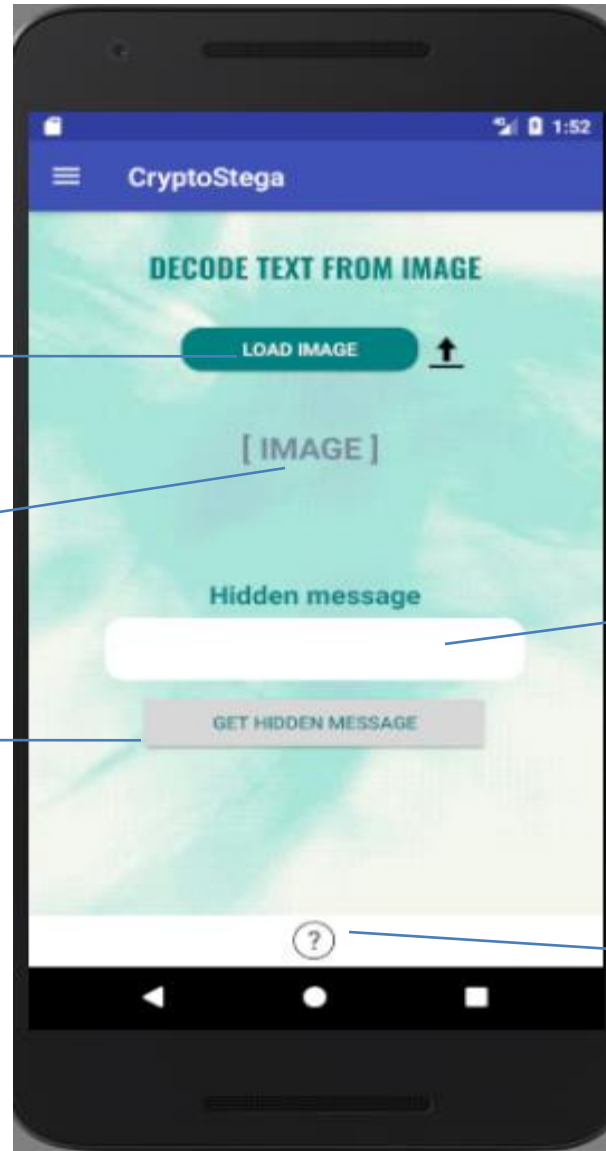
Button to load image

Place where the chosen image will be shown

Button for decoding

Place where the hidden message will show

Help



Requirements
Analysis and Definition

Design

Implementation

Testing

Functional testing

Focus : on functions of applications

Why?

Easier to implement

Ensures that the **software works as expected**

Ensures the **proper working of all the functionalities**



Requirements
Analysis and Definition

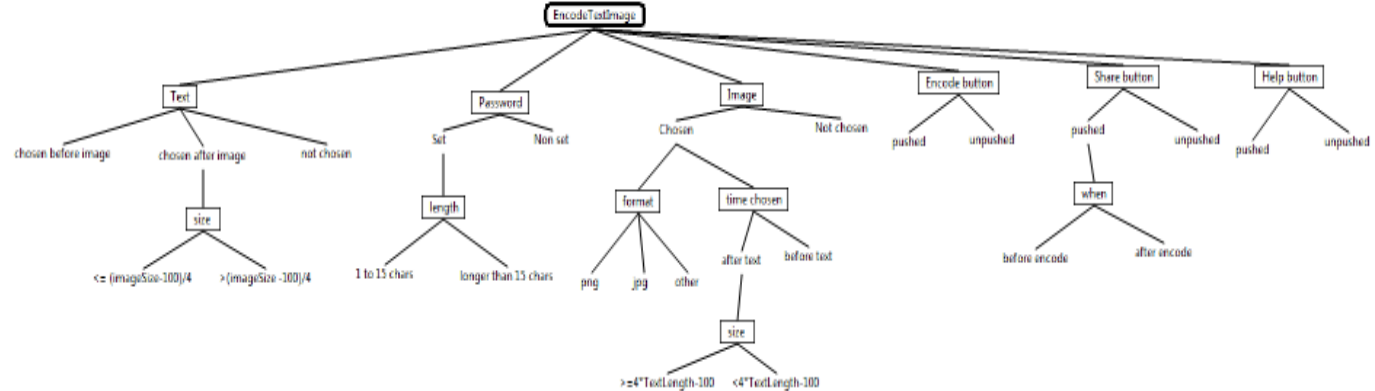
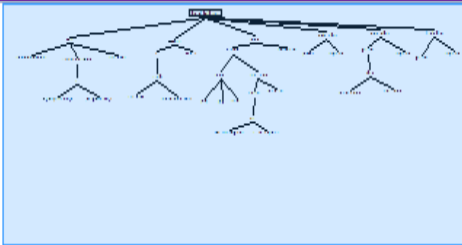
Design

Implementation

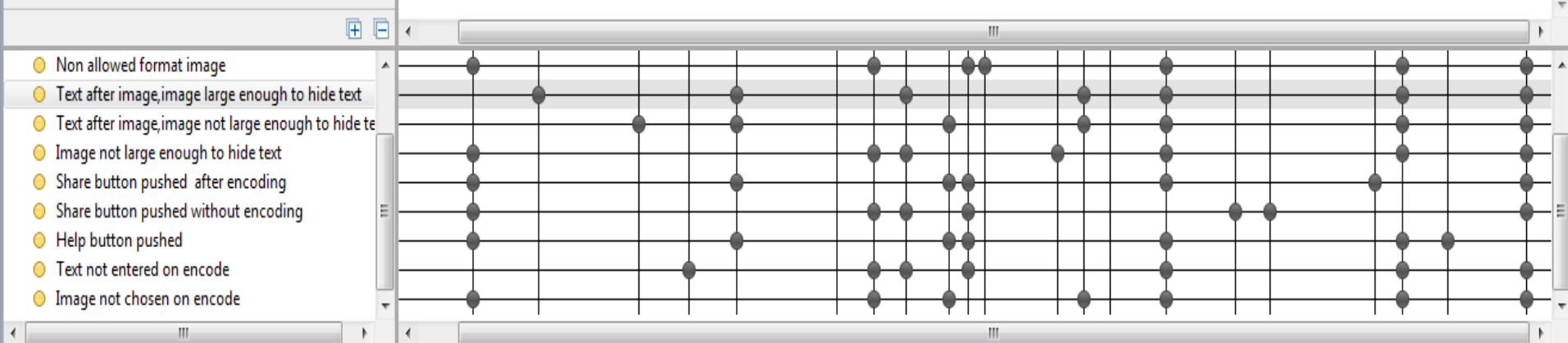
Testing

Testona classification tree and test cases

(Encode text to image)



Min. test cases = 7
Max. test cases = 1008



Requirements
Analysis and Definition

Design

Implementation

Testing

Test cases

(Encode text to image)

Test case	Text	Password	Image type	Image time chosen	Encode button	Share button	Help button	Expected result
1.Encode without encryption	Chosen before image	Not set	png	$\geq 4 * \text{TextLen} + 100$	pushed	unpushed	unpushed	Encoded image
2.Encode with encryption	Chosen before image	1 to 15 chars	jpg	$\geq 4 * \text{TextLen} + 100$	pushed	unpushed	unpushed	Encoded image with encrypted info
3.Password longer than allowed	Chosen before image	Longer than 15 chars	png	$\geq 4 * \text{TextLen} + 100$	pushed	unpushed	unpushed	"Maximum password length is 15 chars" ERROR MESSAGE
4.Non allowed format image	Chosen before image	Not set	Other	$\geq 4 * \text{TextLen} + 100$	pushed	unpushed	unpushed	"Not allowed format type.Should be PNG or JPG" ERROR MESSAGE
5.Text after image,image large enough to hide text	$\leq (\text{ImageSize} - 100) / 4$	1 to 15 chars	png	before text	pushed	unpushed	unpushed	Encoded image
6.Text after image,image not large enough to hide text	$> (\text{ImageSize} - 100) / 4$	1 to 15 chars	jpg	before text	pushed	unpushed	unpushed	"Image not large enough to hide text" ERROR MESSAGE

Requirements
Analysis and Definition

Design

Implementation

Testing

Test cases (2)

(Encode text to image)

Test case	Text	Password	Image type	Image time chosen	Encode button	Share button	Help button	Result
7. Image not large enough to hide text	Chosen before image	Not set	png	$<4 * \text{TextLength} + 100$	pushed	unpushed	unpushed	"Image not large enough to hide text" ERROR MESSAGE
8. Share button pushed after encoding	Chosen before image	1 to 15 chars	jpg	$\geq 4 * \text{TextLength} + 100$	pushed	pushed	unpushed	
9. Share button pushed before encoding	Chosen before image	Not set	png	$\geq 4 * \text{TextLength} + 100$	unpushed	pushed	pushed	"No image encoded" ERROR MESSAGE
10. Help button pushed	Chosen before image	1 to 15 chars	jpg	$\geq 4 * \text{TextLength} + 100$	pushed	unpushed	unpushed	Help screen displayed
11. Text not entered on encode	Not entered	Not set	png	Before text	pushed	Unpushed	Unpushed	"Insert text message" ERROR MESSAGE
12. Image not entered on encode	Chosen before image	Not set	Not chosen	Not chosen	pushed	unpushed	Unpushed	"Cover image not chosen" ERROR MESSAGE

Requirements
Analysis and Definition

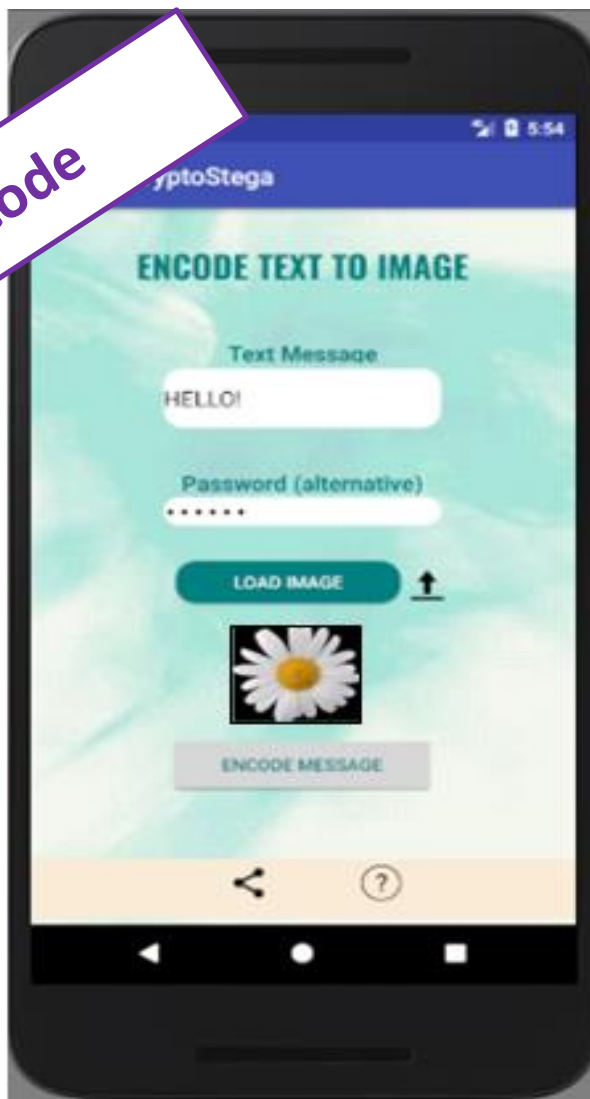
Design

Implementation

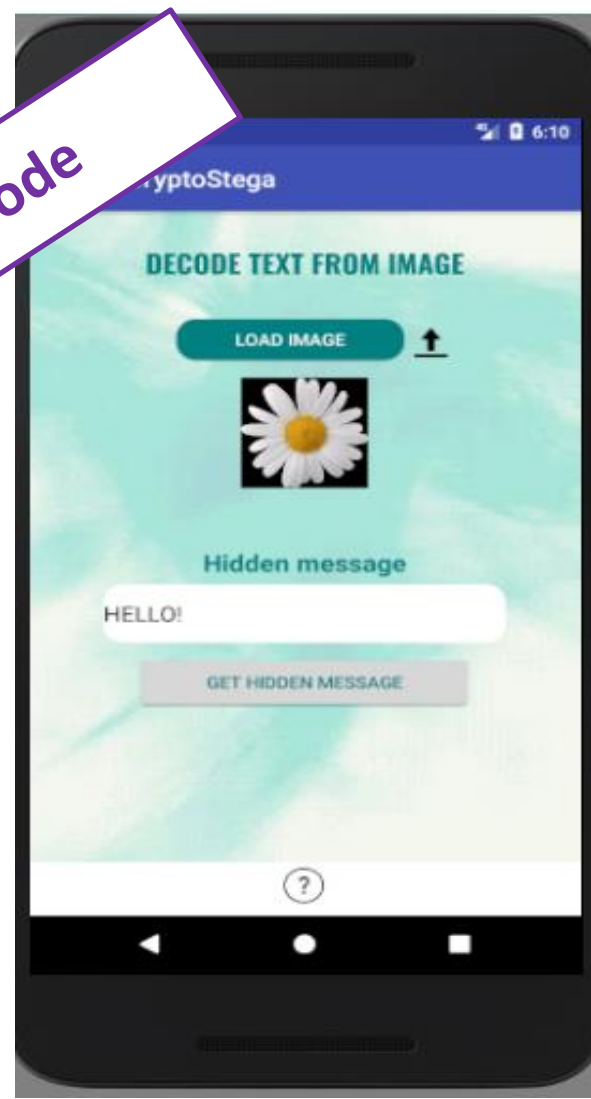
Testing

Demonstration

Encode



Decode



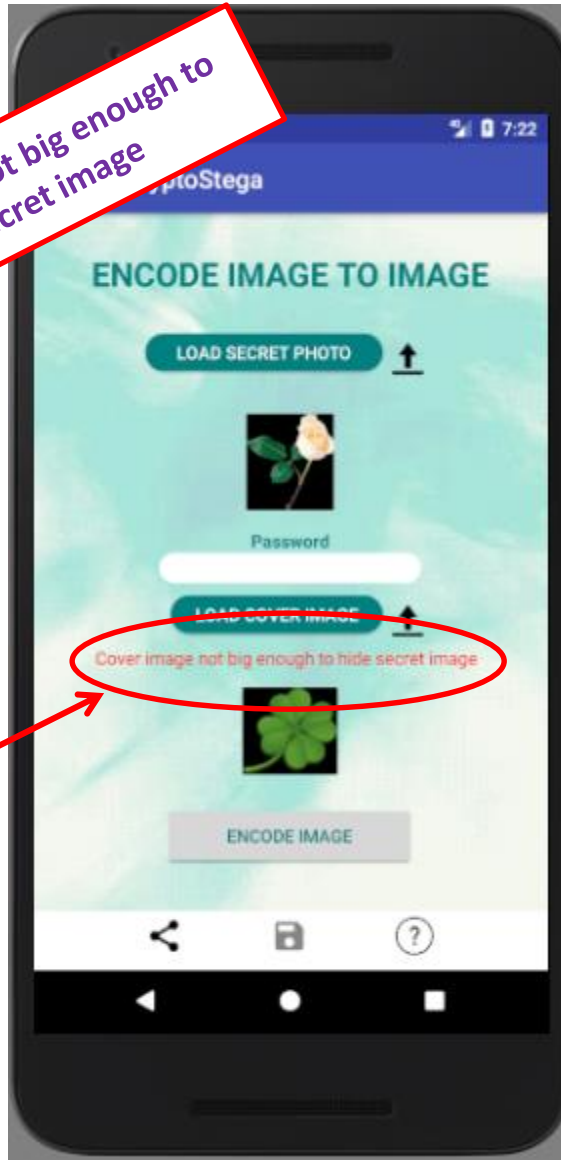
Requirements
Analysis and Definition

Design

Implementation

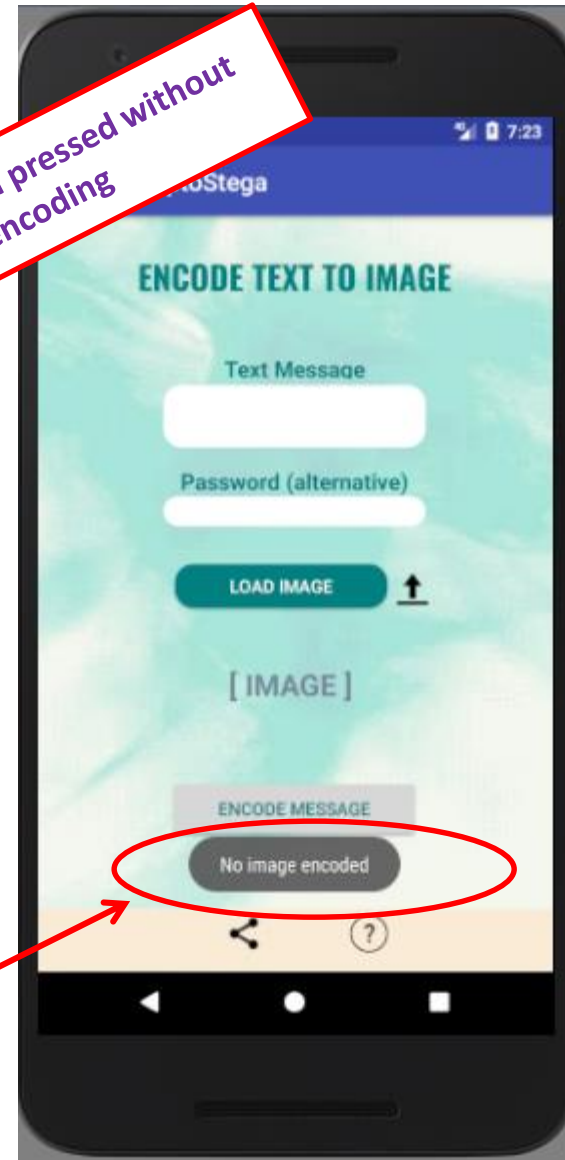
Testing

Error messages



Cover image not big enough to hide secret image

Cover image not big enough to hide secret image



Share button pressed without encoding

No image encoded

Requirements
Analysis and Definition

Design

Implementation

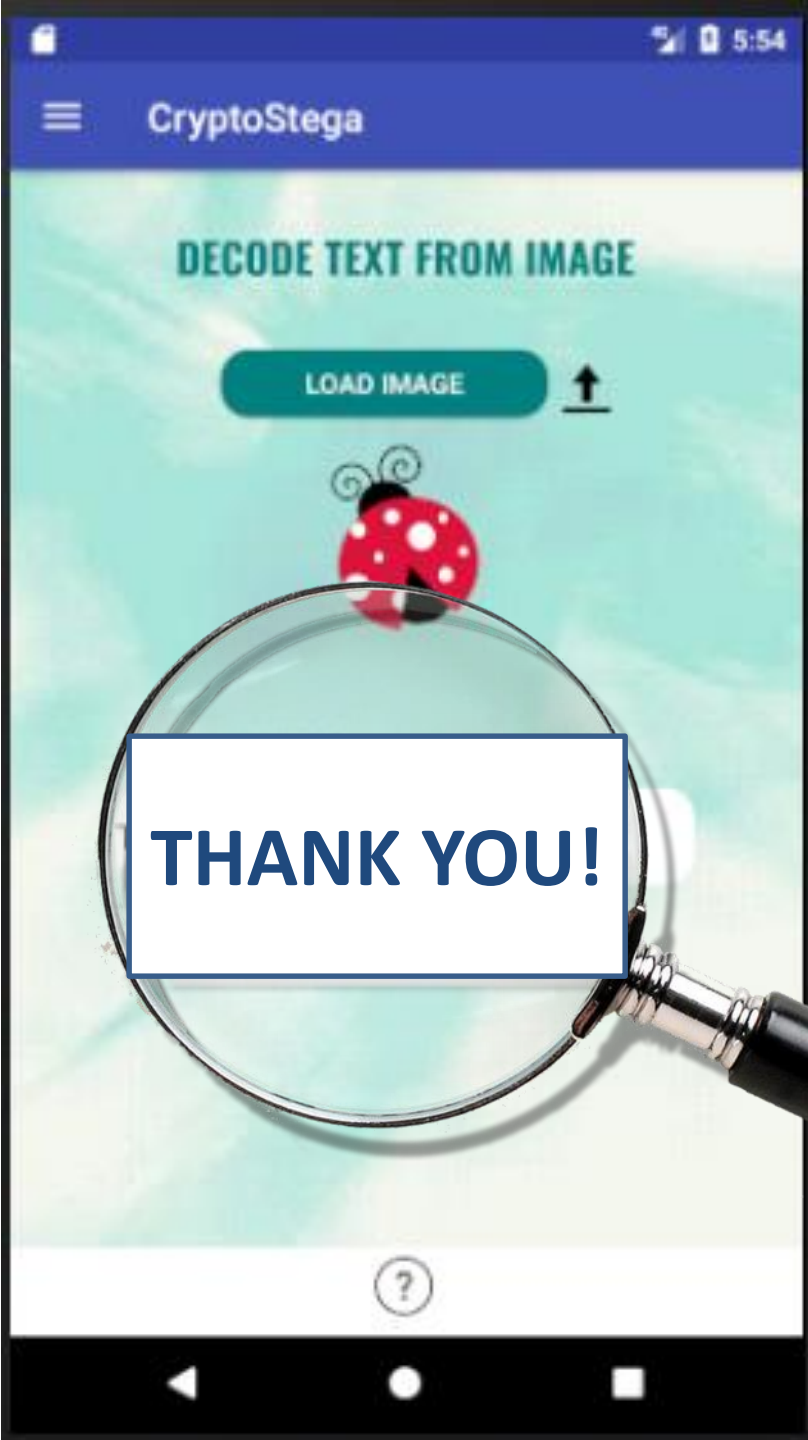
Testing

CONCLUSIONS

Developed new skills along with the learned ones

The application :

- ✓ Hides the information successfully
- ✓ Uses steganography in an effective way
- ✓ Retrieves the hidden information correctly
- ✓ Decreases the possibility that the information will be caught by a third party using cryptography



THANK YOU!