CRYPTO GHOST

Name : Ahmad Almorabea

ID : 1113147

Supervisor: Dr. M. Ahtisham Aslam

Chapter 1 Project Outlines

Chapter 2 | Literature Review

Chapter 3 | Analysis

Chapter 1 | Project Outlines

1.1. Introduction

The social network applications and the most successful methods in spreading and sharing information among the internet and it's growing very fast and every day they provide a way to make the users happy but in the same time they are making ways to get people data to save it in there servers and sell it or buy it or even share it with a third parties Organizations. But sometimes people need to exchange information securely but they have to share it with these applications and compromising Privacy. So I thought of solving this problem by taking advantage of Cryptography Science and make an application to make the users encrypt their Images and sending it over the network securely so the company servers will store the data encrypted and they will not know anything about it but in the mean time when the image delivered to the other person he will simply decrypt the image and save it in a simple way so the first person encrypt the image and send it over the network encrypted and when it will delivered to the other person it will be decrypted and in this way we will communicate privately on the Internet

1.2. Problem Statement

We are exchanging information in a daily basis and this information may contains some private data and some of this data should not be public to the world like passport photo ,financial statement file and a lot of family pictures and when we exchange this we are using some third party applications like Whatsapp , Facebook , Gmail , Hotmail , yahoo ….. So our information is not secure if a hacker got this data he can sell it or he can do some serious problem to the uses

1.3. Clear Statement of Aim

The main purpose is to create an application to save the users privacy and to make users communicate securely

1.4. Clear Statement of Objectives

1. Make everybody's file secure
2. Exchange data in a secure way
3. Make society aware of security
4. Make any user use the application not just the technical people
5. Provide a good usability for encryption software
6. Give a good level of security

1.5. Methodology

Crypto Ghost is software for Image Encryption. The Idea behind its design is that password memorized by the user and this password will be the master password (Finger Print) for the user to encrypt and decrypt. Crypto Ghost will use a symmetric Encryption (one key to encrypt and decrypt) and this key will be stored in the application so the user will not have to enter it manually. Crypto Ghost will use the “AES” Algorithm with 256 bits key size

1.5.1. Overview

When first opened. Crypto Ghost asks the user for a passphrase to enter which it then uses to derive the user’s private key. The application will store the fingerprint in the application. Crypto Ghost will refuse weak passphrases completely until user enter a good passphrase. Then the main application will open and in the main screen user will insert the image that he want to encrypt then he will find 3 options. Option number one will ask the user if he want to encrypt the image for himself by his (Finger Print} so no one can decrypt the image except the user. Option number two will ask the user to enter a password to encrypt the image with “in case if he want to encrypt the image for his friends “so he can send the image and the password and the other person will decrypt the image whiten the password it given to him. Option number 3 will ask the user if he wants to keep the original image or he want to overwrite the image itself then the user can press the button that labeled encrypt to encrypt the image.

1.5.2. Algorithm Description

The Advanced Encryption Standard (AES) is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001.AES is based on the Rijndael cipher developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, who submitted a proposal to NIST during the AES selection process. Rijndael is a family of ciphers with different key and block sizes. For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.AES has been adopted by the U.S. government and is now used worldwide. It supersedes the Data Encryption Standard (DES), which was published in 1977. The algorithm described by AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data.

1.5.3. Key Derivation

Crypto Ghost uses AES with 256 bit key size and user can enter a passphrase minimal to 10 characters (80 bits) and an email address to derive a 256 bit key I'm putting the password and the email address with BLAKE2 hash function then I'm injecting the output inside script key derivation function

```
SecureRandom NonceSeed = new SecureRandom ();

 byte[] Nonce = new byte[16]; // 16 bytes = 128 bits
```

1.5.4. File Format

Crypto Ghost will take (PNG, JPEG) image file then convert it to bytes and then preform the encryption. And then the encryption files will contains some known bytes so Crypto Ghost can identify that this is a Crypto Ghost encrypted file. And a hashing for the original image will be inserted to the image before encryption so when the file is decrypted will be check for integrity of the image

1.5.5. File Encryption

When the Application is fires opened user will enter a passphrase and then the Application will generate a Finger Print for this particular user and it will be different for every user

- Application will convert image to bytes
- Application will encrypt the whole image using the Finger Prints
- Application will put some Known bytes to the header of the image so Crypto Ghost will identify their files

```
SecretKeySpec key = new SecretKeySpec(keyBytes,"AES");

 Cipher cipher = Cipher.getInstance("AES/GCM/NoPadding/","BC");
```

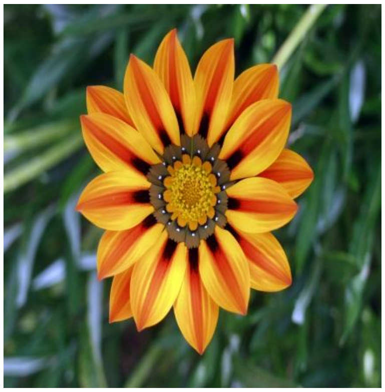
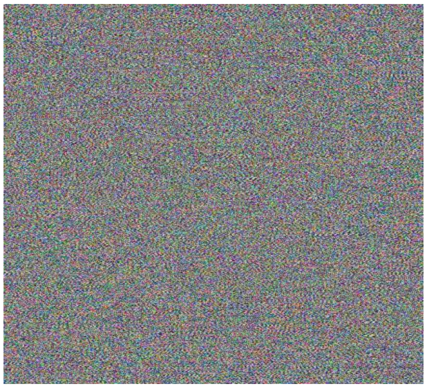
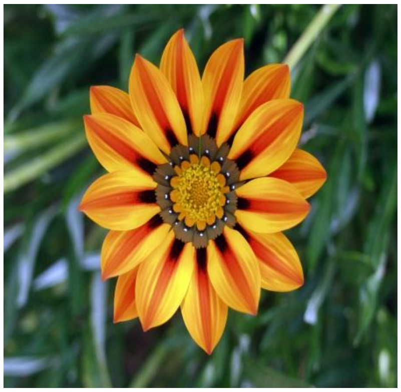
1.5.6. File Decryption

In order to decrypt the image the application will try to decrypt the image automatically if the image encrypted with the same Finger Prints on the device if not the application will ask the user for the passphrase

Steps:

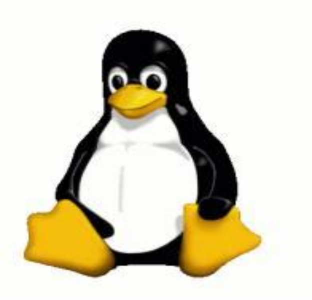
- Application will verify that the file is in fact a Crypto Ghost file
- Application decrypts the file with the passphrase
- Application will compare the Hash with the original Image if it's okay Image will be shown to the user if not it will give the user alert the image maybe tempered

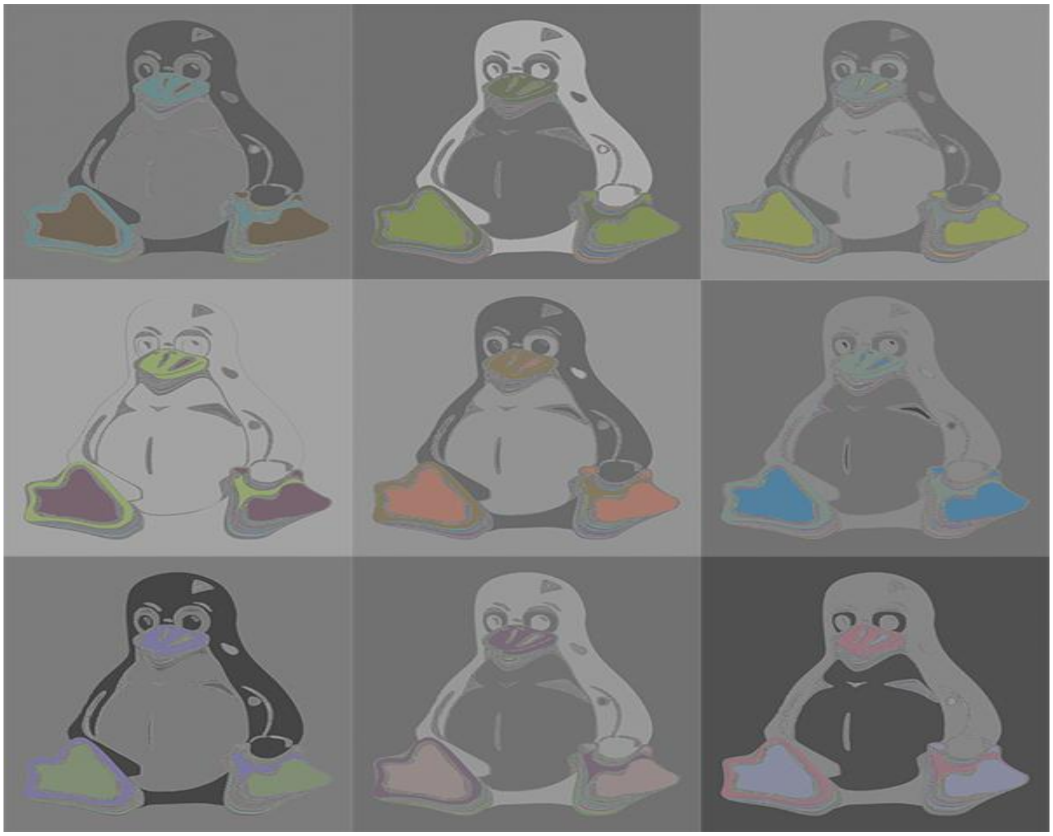
1.5.7. Results

Original Image	Encrypted Image	Decrypted Image
		

1.5.8. Why not to Encrypt with ECB mode

This mode is not secure anymore and has some major security threats, and to demonstrate the idea, if there is any value that’s repeated it will be encrypted with the same value and it will reveal the value and you can see a demonstration in the figure below





2. Project Plan

Project Start Date: 05/10/2014
Project End Date : 1/07/2015
Total Month : 6 Months

Note: The Project will be spread on:

1. **First Semester 2014-2015 (CPIS 498)**
2. **Second Semester 2015 (CPIS 499)**

14.1. Clear Task Specification

Period	Month	Phase	Tasks
First Semester	1	Planning and preparation	• Introduction to the problem • What is the platform that will developed on • What are the current solutions • Limitation of existing solutions • How I will solve this limitation • Enhance the literature with 6 existing solutions • Comparative study of feature and drawback of existing solutions • Study that the encryption process should be based on pixels or bytes • Initial implementation of android app to load and display the image
	2	Analysis	• Gathering requirements • Define the problem definition • Problem gave me motivation to do this application • Identifying system stakeholders • The Project Scope • Functional and non-Functional Requirements • Methodology • What is AES Algorithm • User Flow • Key Derivation • File Format • File Encryption • File Decryption • Results • Result for encrypting using pixels • Questionnaire • Equipment and software required
	3	Design Phase	• Draw use Case Diagram • Draw Class Diagram • Draw Flow Chart Diagram • Initial Design • Poster Designing • Final Presentation

Second Semester	4 & 5	Implementation Phase	- Android Programming - Adjusting Interface by looking at usability point of view
	6	Test	- Unit Testing - Integration Testing - Interface - Review design

14.2. Task Duration

Period	Month	Start Date	End Date	Phase
First Semester	1	05/10/2014	25/11/2014	Planning and Preparation
	2	04/12/2014	09/12/2014	Analysis
	3	09/12/2014	15/01/2015	Design Phase
Second Semester	1	16/1/2015	19/03/2015	Implementation Phase
	2	20/03/2015	07/05/2015	Implementation Phase
	3	08/05/2015	01/07/2015	Test Phase

Chapter 2 | Literature Review

1. Background and Overview of Related work

In this section I will show some existing solution and I will describe the problem of these application

1- Image Crypto :

This app allows you to hide text messages in the pictures! Just enter text, select the image and click the "Crypt" button. A copy of the image will be formed with the prefix "crypt" in its name. Now you can send this file with a hidden message to your friends. If you want to read a message from a file, simply select it and click "Decrypt" button. This application just put the texts inside the image it will not encrypt the text [1]

2- Hide Image :

This app will lock the images that you want to hide In the application and they claims that they are using “AES” cryptography Algorithm but they are not making their software available so we can see the implementation of their work so we can’t just trust this app for encrypting our images... Yes maybe they provide security on the device but who knows if we use this application online [2]

3- Image Crypt :

They are encrypting image with “DES” Algorithm and also they don’t put their software available so we can review their implementation beside DES is not secure anymore for using it on the public [3]

4- File Crypto :

This app allows to encrypt all kind of files by AES encryption technology simply using a password you can easy remember. So you can keep confidential all kind of information related to your intellectual property: text, images, documents and any kind of file that you can browse by the embedded file manager. You can open the decrypted file directly from the app without the needs of an external file manager. When you choose to encrypt a file the app creates the encrypted file with extension .cry. For user convenience the original file is not deleted so the user can decide what to do with it. All encryption processes are executed locally on your device. Maybe the application will provide some security but the source code it’s not available to check and there is no trusted document for proven security beside its not open source [4]

5- Hide Photos – TimeLock :

With the new TimeLock app, it is not only a timeless clock with an alarm function, but also a high security vault for your personal photos and videos. The vault itself is completely invisible, hidden in the design of the clock. The app integrates a powerful vault, which comprehensively protects all the backed up photos and videos in it, while offering unmatched usability. With the latest security technologies TimeLock keeps all of your pictures and videos militarily secure, because the entire contents of the vault is encrypted with the strong 256-bit AES (Advanced Encryption Standard) algorithm. The AES-256-bit encryption method is one of the safest in the world and is used, but there is a problem that is this app it doesn’t provide it’s source code to review [5]

6- Secure Gallery(Pic/Video Lock) :

This application will lock the images that you want to hide with a master password or pattern lock but this app will not encrypt the image so any scan of the memory will reveal the photos [6]

2. Critical Analysis

	Encryption type	Key size	Open source	Provide Steganography
Image Crypto	No Encryption	No key	No	Yes
Hide Image	AES	128	No	No
Image Crypt	DES	56	No	No
File Crypto	AES	128	No	No
TimeLock	AES	256	No	No
Secure Gallery	No Encryption	No key	No	No
Crypto Ghost	AES-GCM	256	Yes	No

3. Overview of implementation Tools

All Application are using Java Programming language to program this application on android operating system

Chapter 3 | Analysis

1. Requirement Capture Or Data Collection

The important thing to understand what the users need and understand what Their requirement, is data gathering. Which is collecting data from users by Known approaches and methods such as interviews, questionnaires, Observation, workshop groups etc. whenever using more than one approach, The quality of information will be high. It will support you to understand the Requirements much better. I conducted Questionnaire and interviews Approach.

1.1. Questionnaire Description

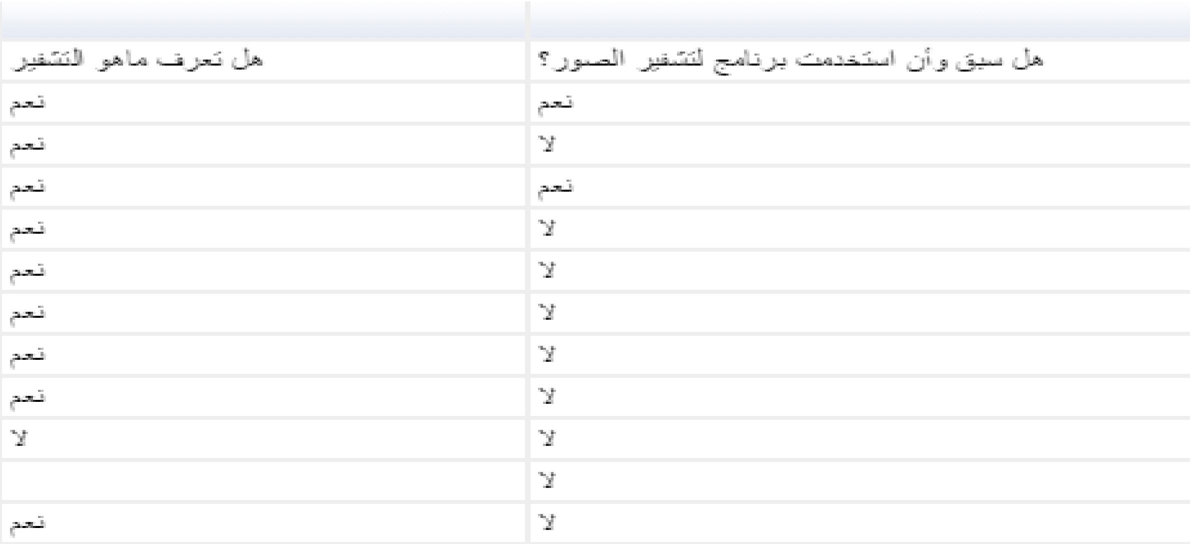
I preformed Questionnaire to know if the users have knowledge about Cryptography or not and to know if they have some good application that they depending on I preformed two versions of questionnaire first questionnaire for the people who knows about Encryption and second questionnaire for the people who are not familiar about encryption so I will give them an idea about encryption.

In the Appendix I will put a screen shot of the questionnaire forms

1.2. Questionnaire analysis

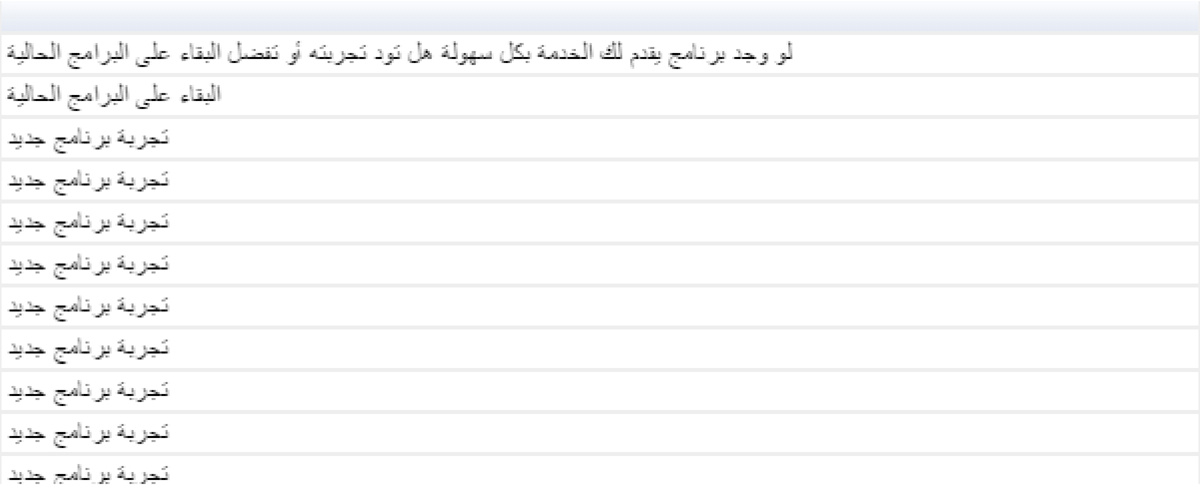
I asked the people if you know what encryption is, and most of them said “yes”

I asked the people if they use any app for image encryption before 9 of them said “yes” and 2 of them said “No”



هل تعرف ماهو التشفير	هل سبق وأن استخدمت برنامج لتشفير الصور؟
نعم	نعم
نعم	لا
نعم	نعم
نعم	لا
نعم	لا
نعم	لا
نعم	لا
نعم	لا
لا	لا
	لا
نعم	لا

I asked the people if they want to try a new application for image encryption most of them they said yes



لو وجد برنامج يقدم لك الخدمة بكل سهولة هل تود تجربته أو تفضل البقاء على البرامج الحالية
البقاء على البرامج الحالية
تجربة برنامج جديد
تجربة برنامج جديد
تجربة برنامج جديد
تجربة برنامج جديد
تجربة برنامج جديد
تجربة برنامج جديد
تجربة برنامج جديد
تجربة برنامج جديد
تجربة برنامج جديد

1.3. Non-structured interviews

Which is I give the idea of the project to the person and see what he prefers or suggests and catch the ideas.

1.4. Conclusion

Most of the users are aware of security and most of them used some encryption software's and this is good because if I build the application they will know how to use it and what is the purpose of it

2. Requirement Specification

2.1. Functional Requirements

- **Signup**

  Signup is the first function the user will use after putting his password for the first time and this is the most important function because from this function the application will store the password of the user and system will generate the unique key for the user (Finger Prints) and this key is the private key of the user to encrypt his Images.

- **Login**

  Login is the function it will appear to the user in every time the user enter the application and in this function it will verify the password of the user and it will verify the unique key is available

- **Image Insertion**

  This the function that the user will insert the image from so the user will click the image insert button and then he will be directed to the gallery to choose a specific image

- **Encryption**

  This is the main function of the application and from this function the user will insert the image from the 'image insertion' and then the user will encrypt his images and it will encrypt the image using the unique key (Finger Prints) that is generated in the signup function and then it will generate the encrypted image

- **Decryption**

  Decryption it will also count as a main function of the application so this function will decrypt the image and it goes like this user choose image from the gallery using the 'image insertion' and then it will decrypted using the unique key if the user encrypted the image with the unique key or decrypt the image using different key if the image encrypted using different key

- **Change Password**

  In this function the user can change his password but it will not change the unique key of the application.

- **Rest Application**

  This function will rest the application completely by deleting the password and the Unique key (Finger Prints).

- **Share**

  This function will allow the user to share the image in any application whether the image encrypted or decrypted.

2.2. Non-functional Requirement

The non-functional requirements are not basics like functional requirements. They represent the qualities of our system.

- **Usability**
- **Response time**

2.3. Software Requirements

1. MS Office Power Point: for project presentation.
2. MS Office Word: for Project Report.
3. MS Visio: For project diagrams.
4. Eclipse: for android programming.
5. NetBeans IDE: for java development tools.

2.4. Hardware Requirements

1. Mobile Phone
2. Laptop

2.5. Software Methodology

I’m going to build the application using Object-Oriented-Programming “OOP” because it’s the First language for building android applications

3. User Profile

3.1. User categories

This Application will help people to encrypt their photos so anyone who are interested in privacy can use this application whether it’s male or female normal people or technical people and the application will be using a simple interface to support that

3.2. Type of users

1. Male
2. Female
3. Normal people
4. Technical people
5. Advanced people

3.3. User Flow

This section will give an example of user flow in order to help demonstrate how Crypto Ghost is supposed to help people.

Scenario 1:

Alice wants to encrypt an image for herself so she will open the Crypto Ghost software and insert the image and she will check the options that she want and she will have an image encrypted in her device and when she want to decrypt it she will open the Application and insert the image and press decrypt and the image will be decrypted

Scenario 2:

Alice wants to send an image to her Friend Bob like (financial paper) so Alice will open Crypto Ghost and insert the image and she will enter a password for her friend and encrypt the image for him and she will send the image and the password to Bob in order for him to decrypt the image

3.4. Stakeholders

- Users : user who will send the image
- Third party applications : the companies that have the servers to store users data

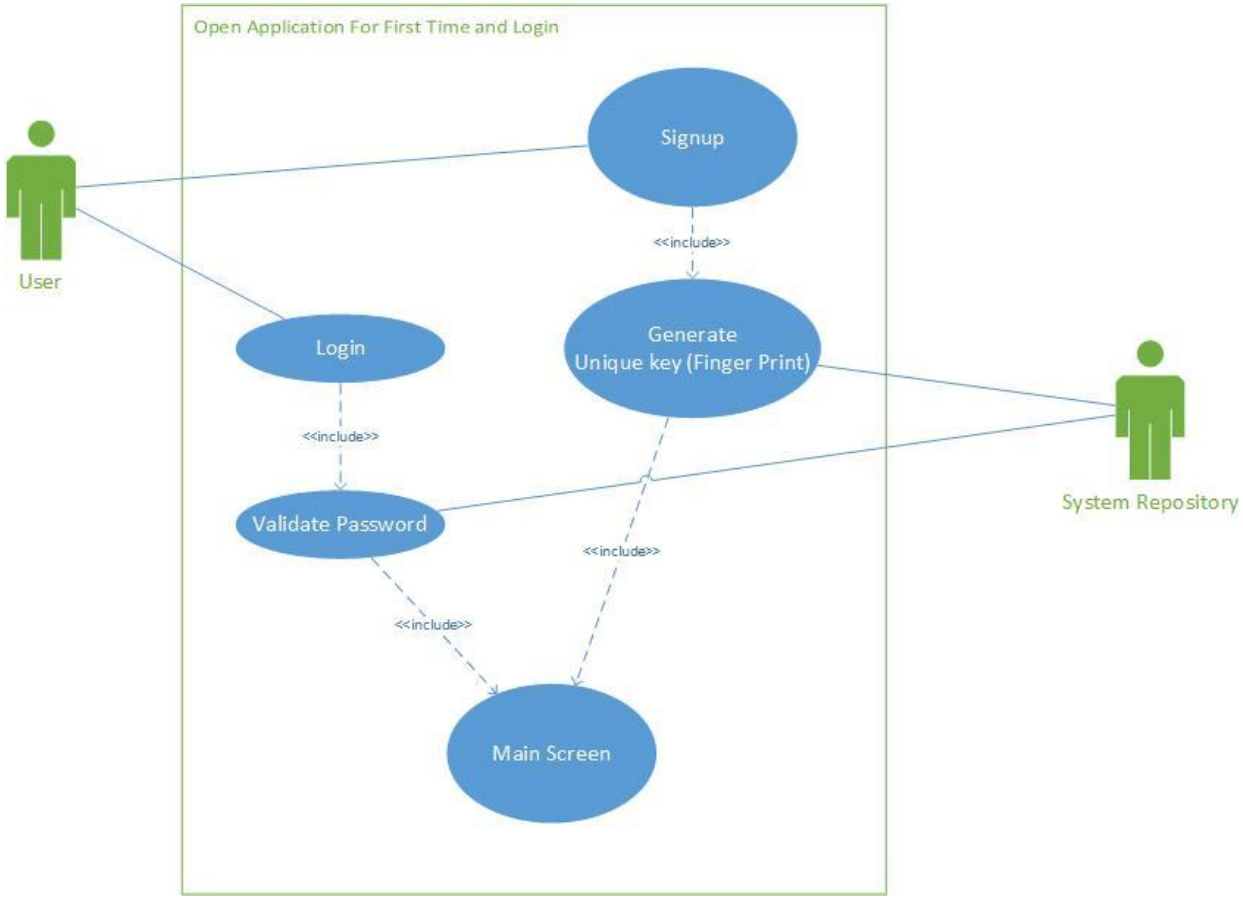
3.5. Environment

1- Crypto Ghost application will be running on Android operating system
2- No Internet is required

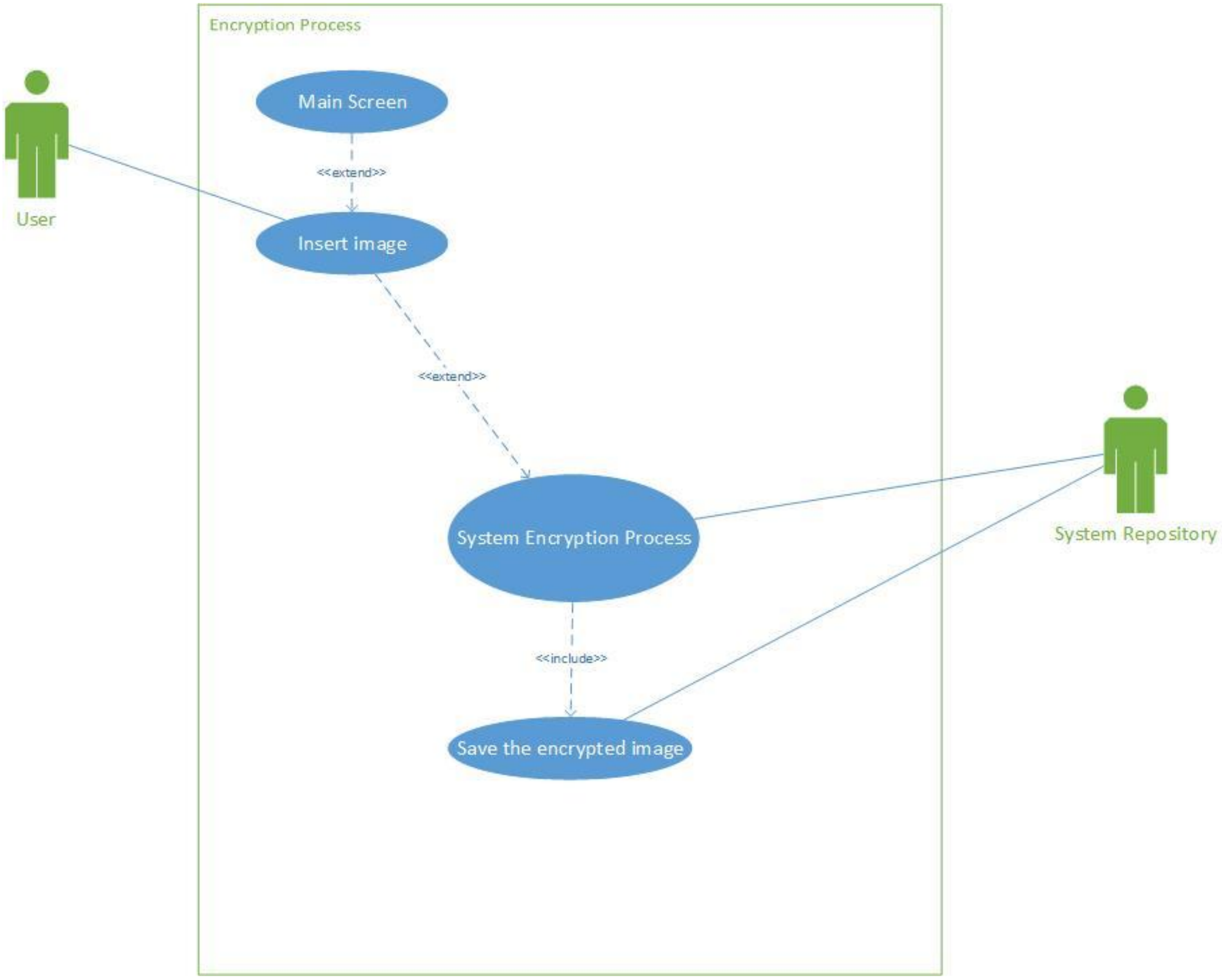
4. Structuring System Requirements

4.1. Use Case Model

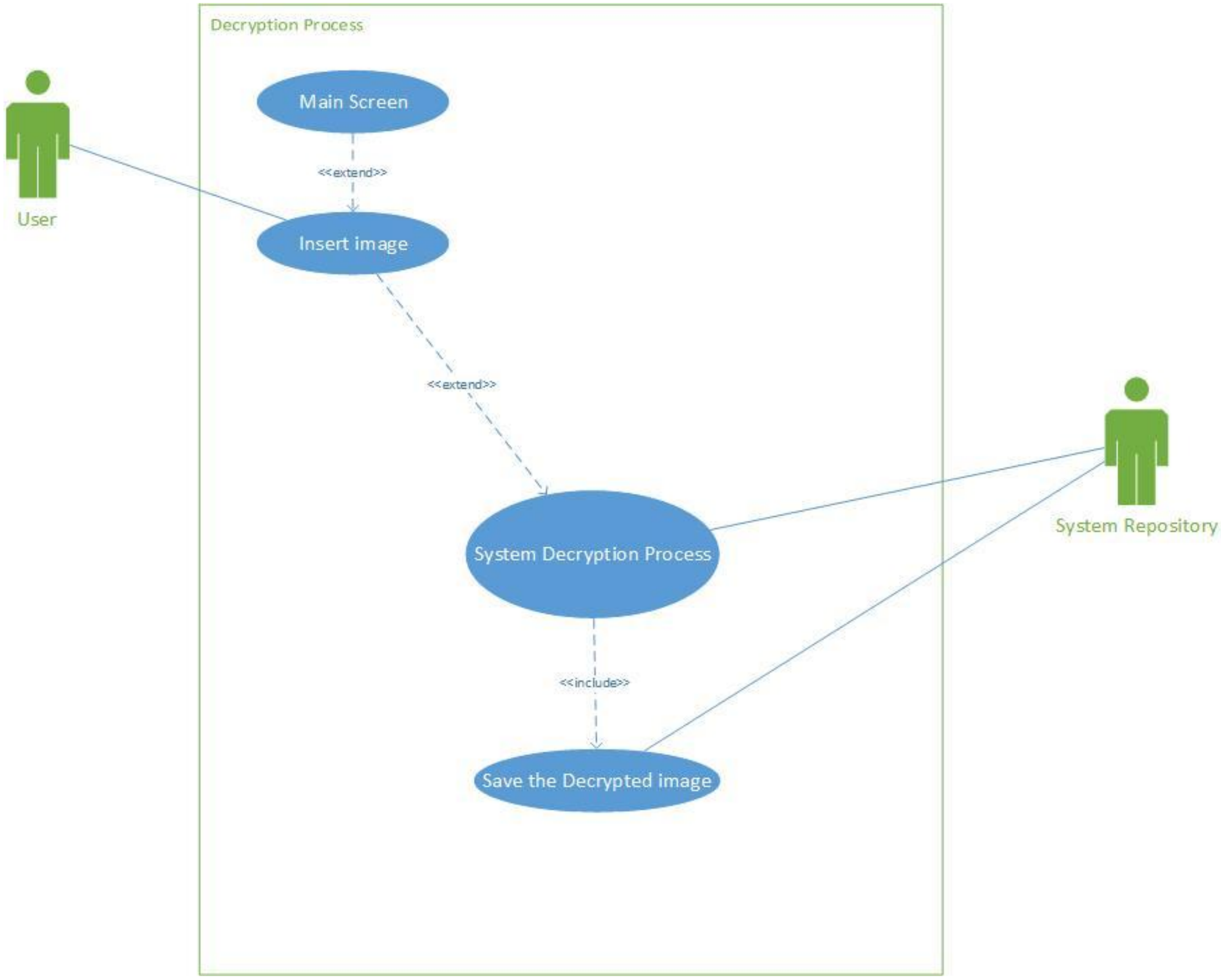
4.1.1. Open Application for First Time and Login



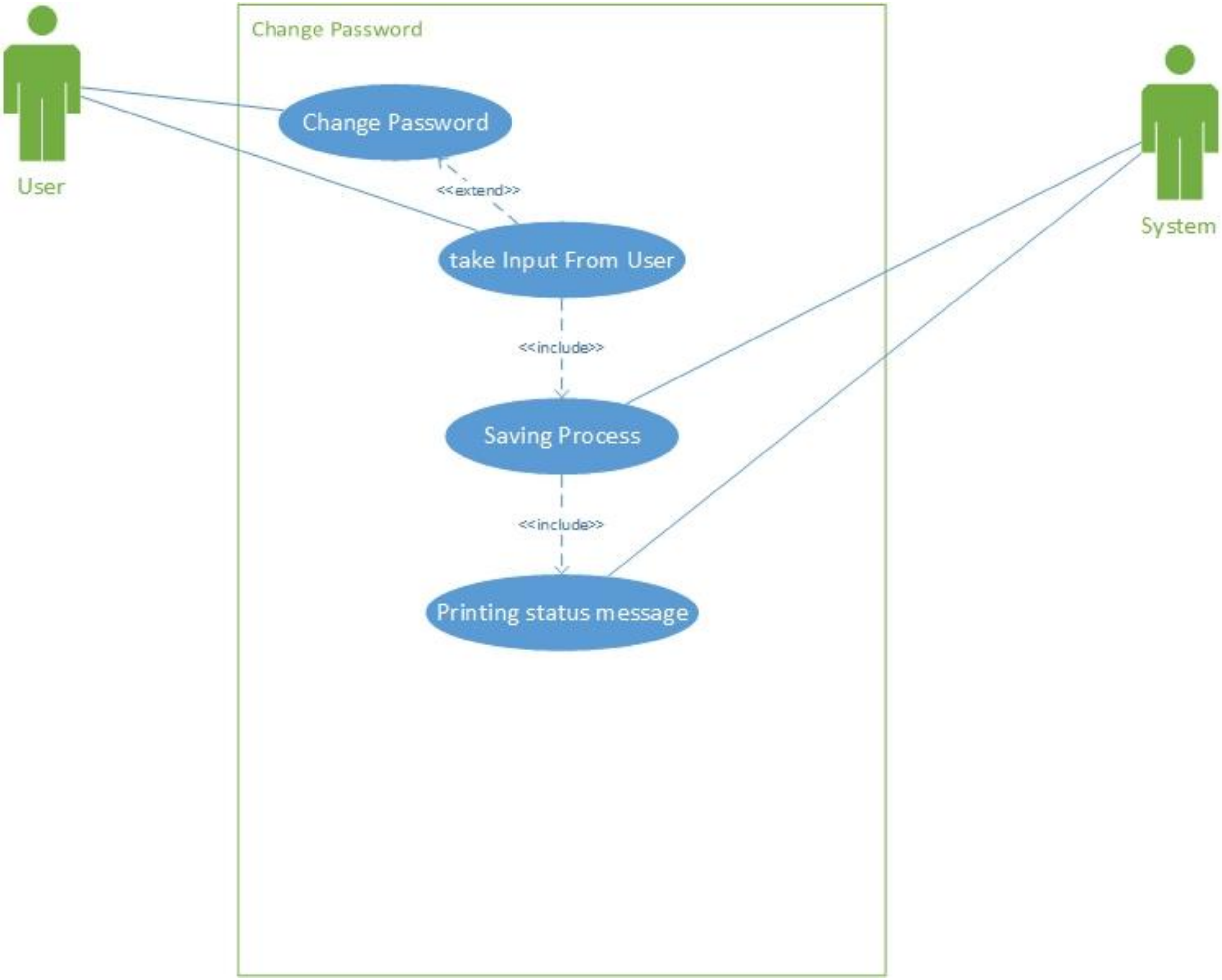
4.1.2 Encryption Process



4.1.3. Decryption Process



4.1.4. Changing Password



4.2. Description of the Use Cases

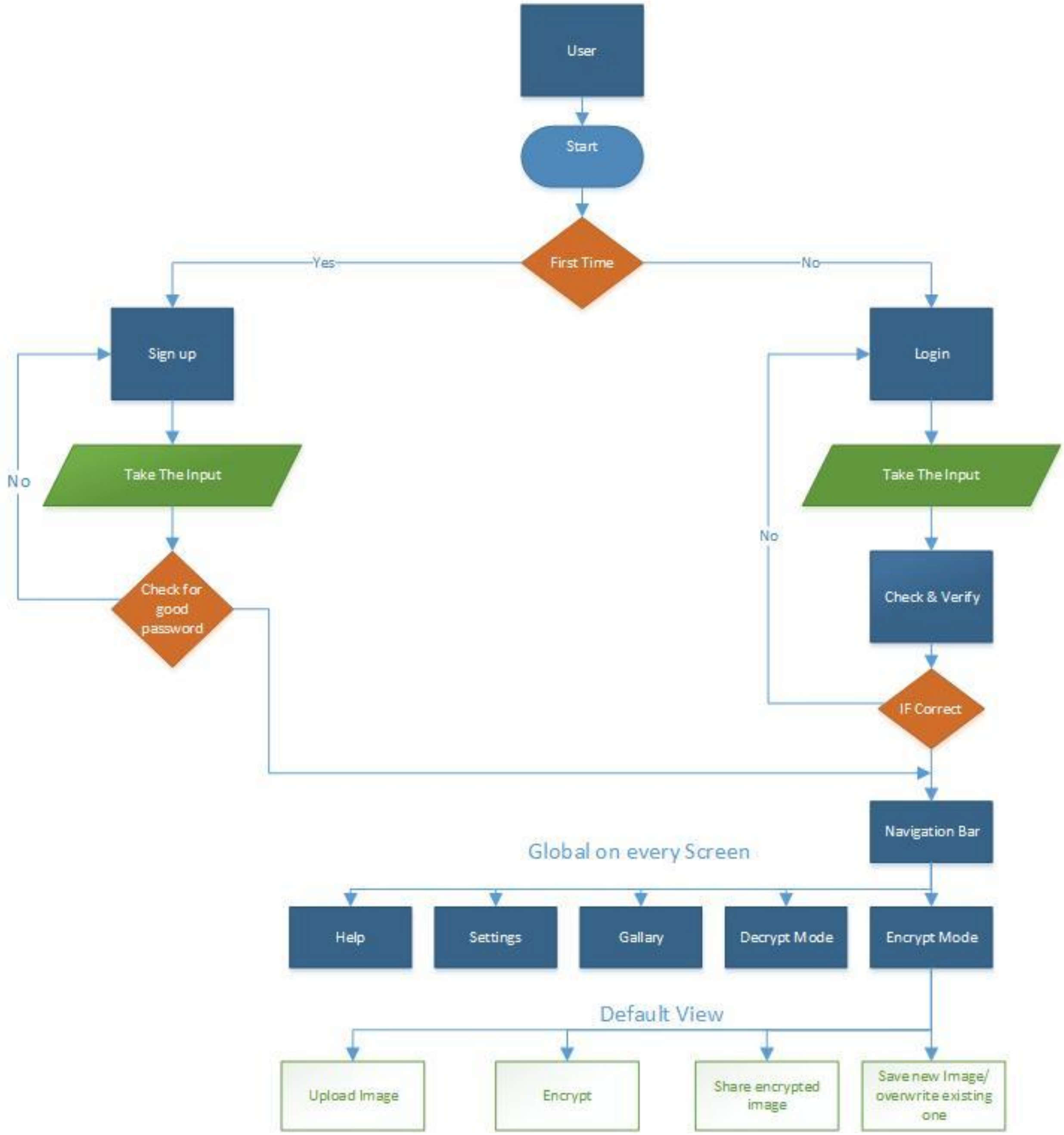
Use case name	Open Application For First Time
Scenario	Saving new password
Triggering Event	Clicking [Application Icon]
Brief Description	Open application for first time and saving a new password
Actors	User
Related Use Cases	
Stakeholders	User, System
Pre-condition	Logged in to the system
Post-condition	The password is saved
Flow of Activities	Actor System 1.Click[App Icon] 2.new password screen 3.Enter new Password 4.Generate Finger Print 5.Main Screen
Exception Conditions	If password is not strong enough the application will give a user a message that try to enter a new password again

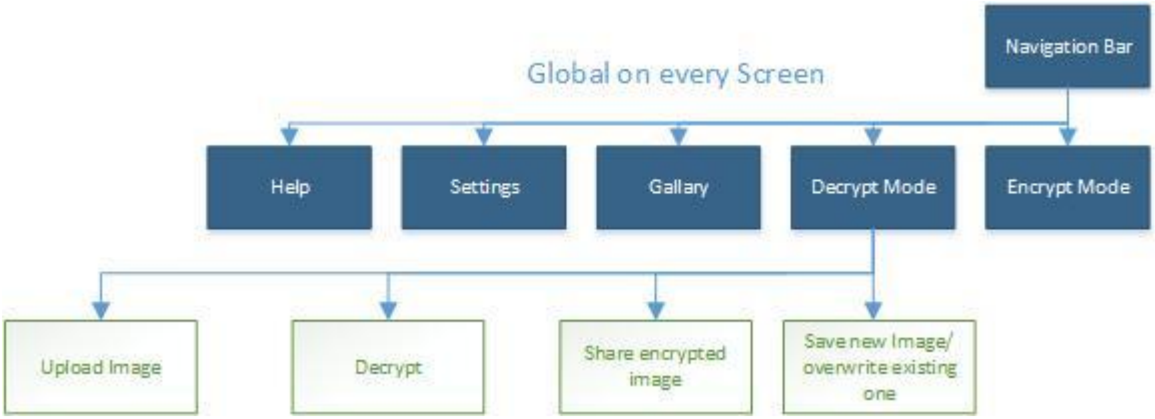
Use case name	Encryption
Scenario	Encrypt images
Triggering Event	Clicking [Encrypt]
Brief Description	Insert image to encrypt
Actors	User
Related Use Cases	Open First Time
Stakeholders	User, System
Pre-condition	Logged in to the system
Post-condition	Encrypted image saved on the device
Flow of Activities	Actor System 1.Click[Insert Image] 2.choose encryption Password 3.Press Encrypt 4.adding special bytes on header 5.adding hash of the original Image inside the image 6.Encryption Process 7.Saving image on device
Exception Conditions	2. User choose encryption process either user Finger Prints or different password 6. User choose either overwrite the existing image or create new image so it will be 2 images original image and the encrypted image

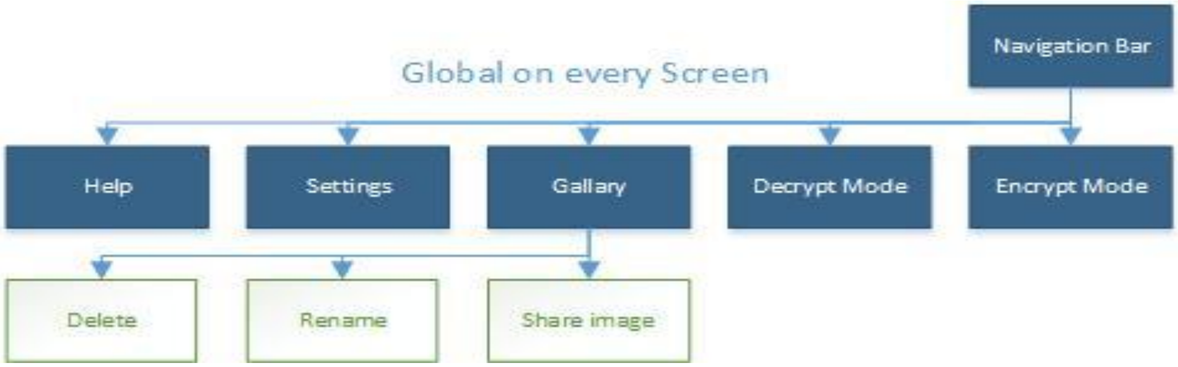
Use case name	Decryption
Scenario	Decrypt images
Triggering Event	Clicking [Decrypt]
Brief Description	Insert image to decrypt
Actors	User
Related Use Cases	Open First Time
Stakeholders	User, System
Pre-condition	Logged in to the system
Post-condition	Decrypted image saved on the device
Flow of Activities	Actor System 1.Click[Insert Image] 2.choose decryption Password 3.Press Decrypt 4.check for special byte 5.Decryption Process 6.Check Image HASH 7.Saving image on device
Exception Conditions	2. User choose decryption password either user Finger Prints or different password 7. User choose either save the image on device or just show the image without saving any image on device

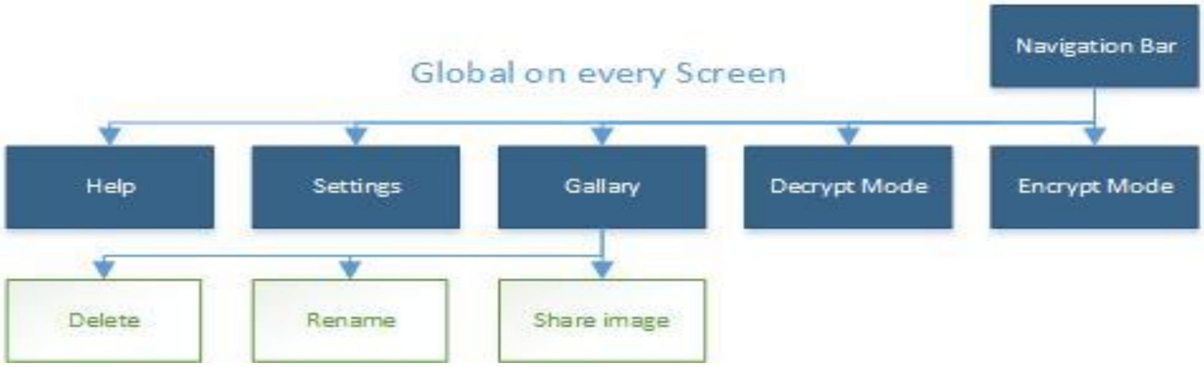
Use case name	Change Password
Scenario	Change Password
Triggering Event	Clicking [Chang Password]
Brief Description	User want to change his password
Actors	User
Related Use Cases	Open First Time
Stakeholders	User, System
Pre-condition	Logged in to the system
Post-condition	Password changed
Flow of Activities	Actor System 1.Click[Change Password] 2.user put new password 3.Press Change 4.Password Saved
Exception Conditions	4. if Password is small will not be accepted

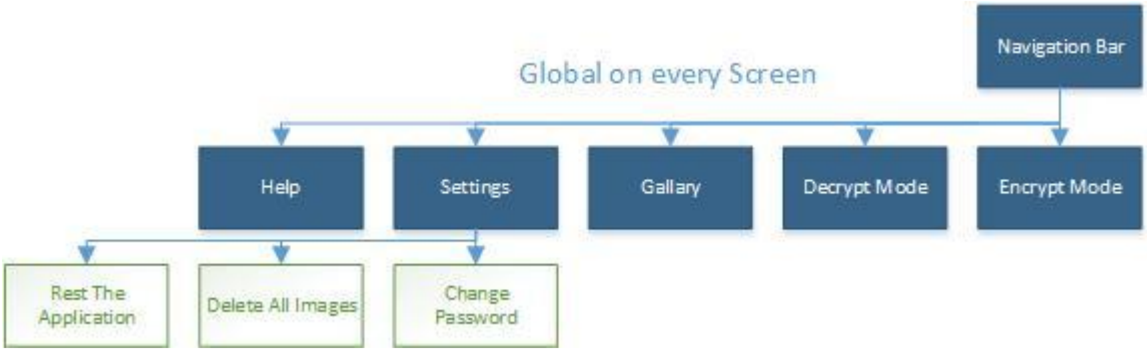
4.3. Data Flow Diagram

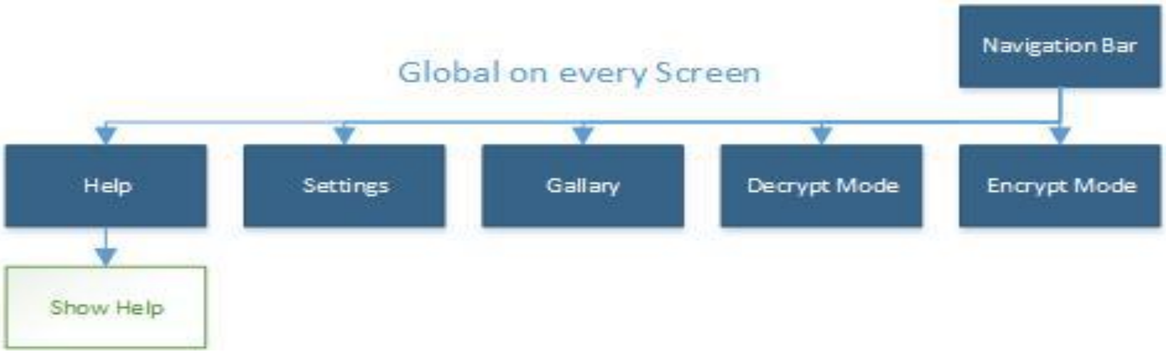



Navigation Bar
Global on every Screen
Help
Settings
Gallary
Decrypt Mode
Encrypt Mode
Upload Image
Decrypt
Share encrypted image
Save new Image/ overwrite existing one


Navigation Bar
Global on every Screen
Help
Settings
Gallary
Decrypt Mode
Encrypt Mode
Delete
Rename
Share image

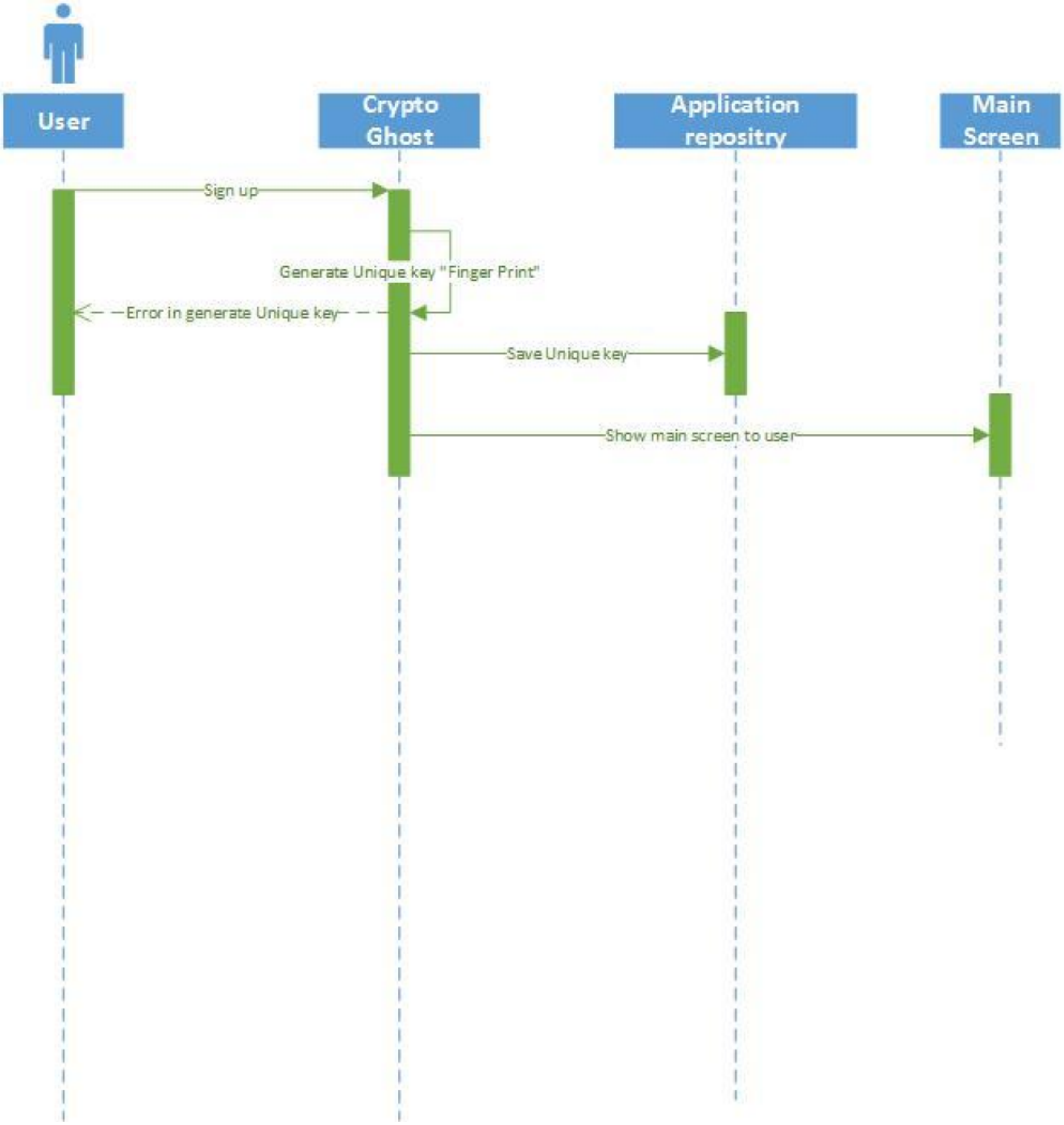

Navigation Bar
Global on every Screen
Help
Settings
Gallary
Decrypt Mode
Encrypt Mode
Delete
Rename
Share image


Navigation Bar
Global on every Screen
Help
Settings
Gallary
Decrypt Mode
Encrypt Mode
Rest The Application
Delete All Images
Change Password

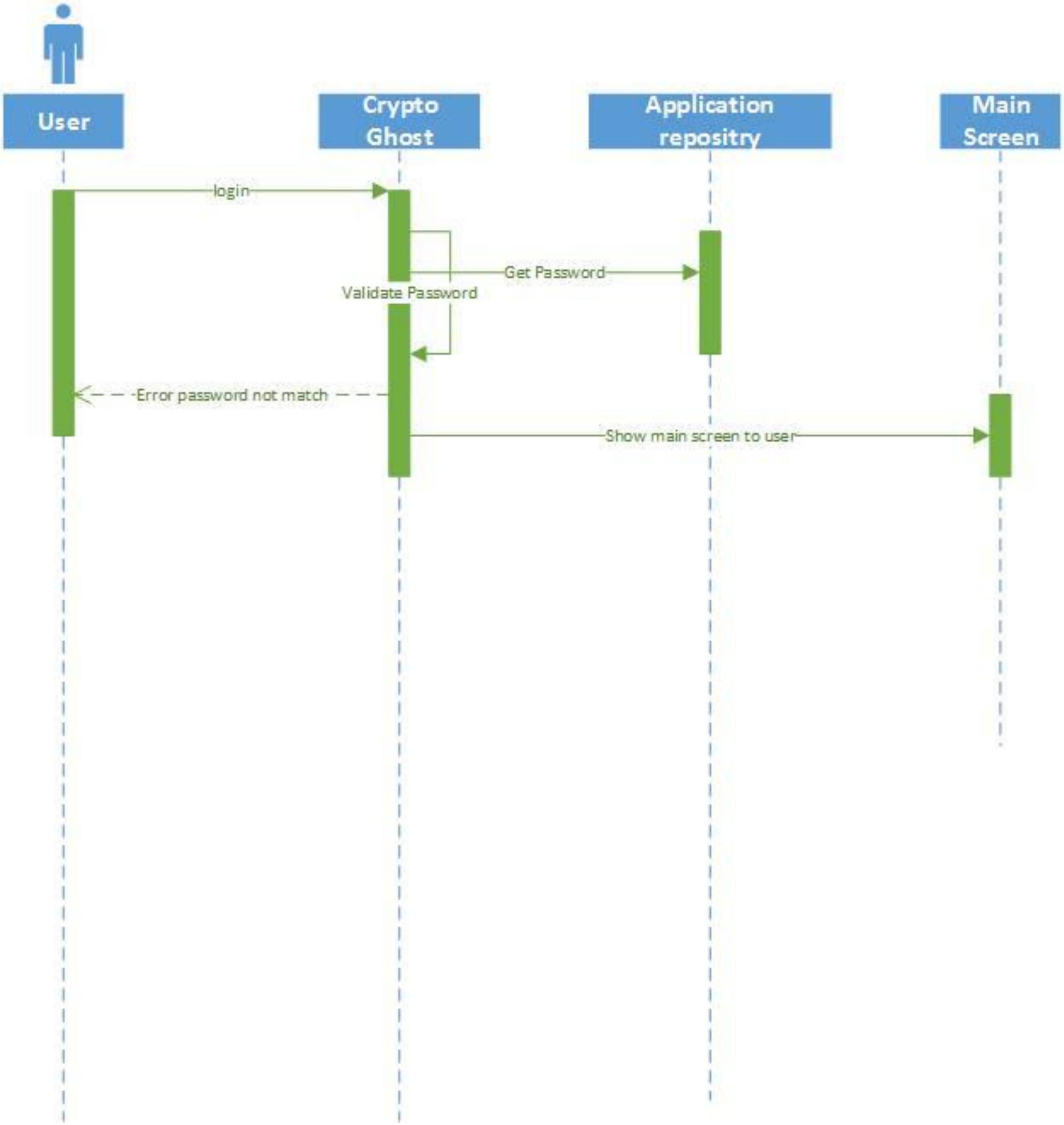

Navigation Bar
Global on every Screen
Help
Settings
Gallary
Decrypt Mode
Encrypt Mode
Show Help

4.4. Sequence Diagram

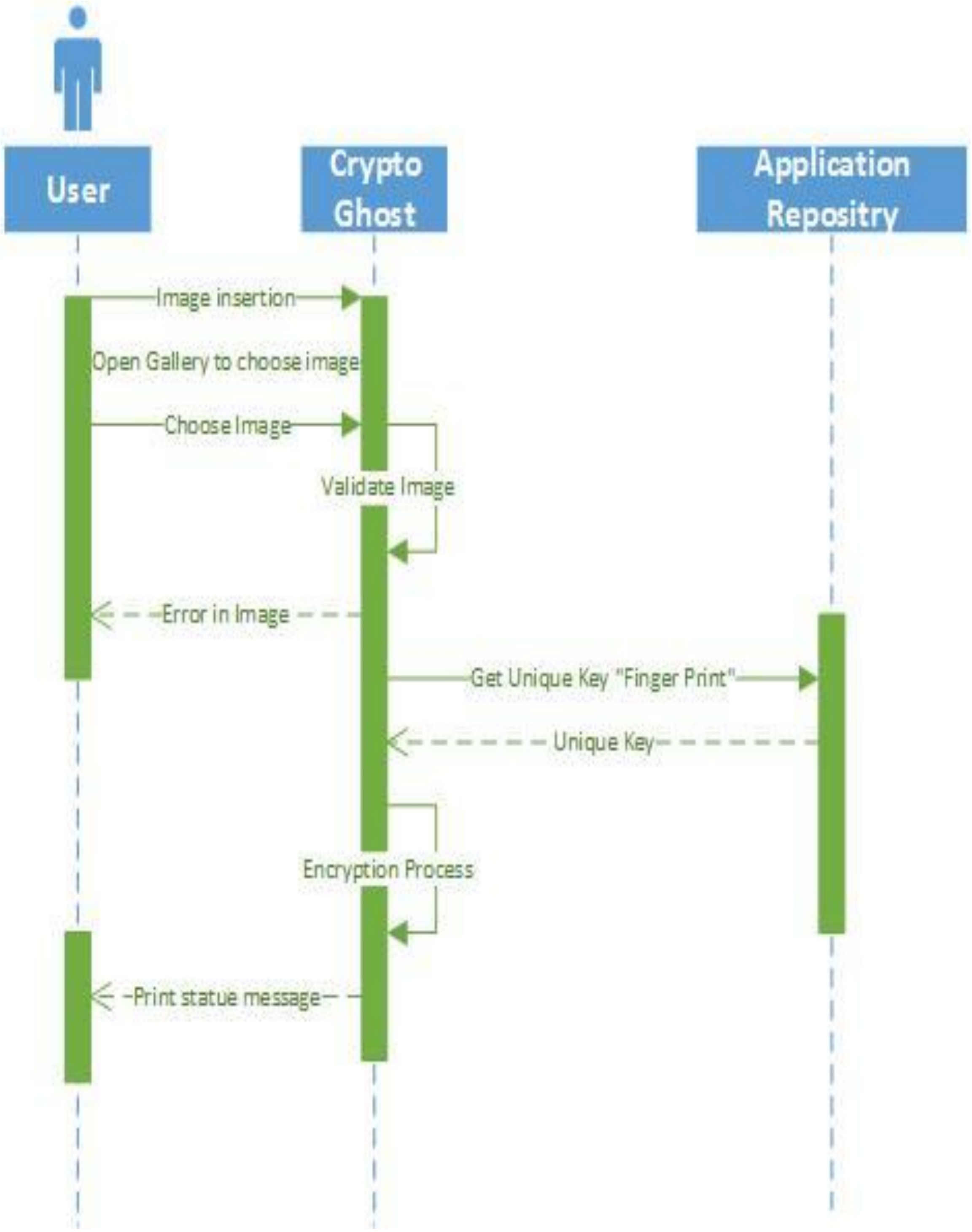
4.5.1. Signup



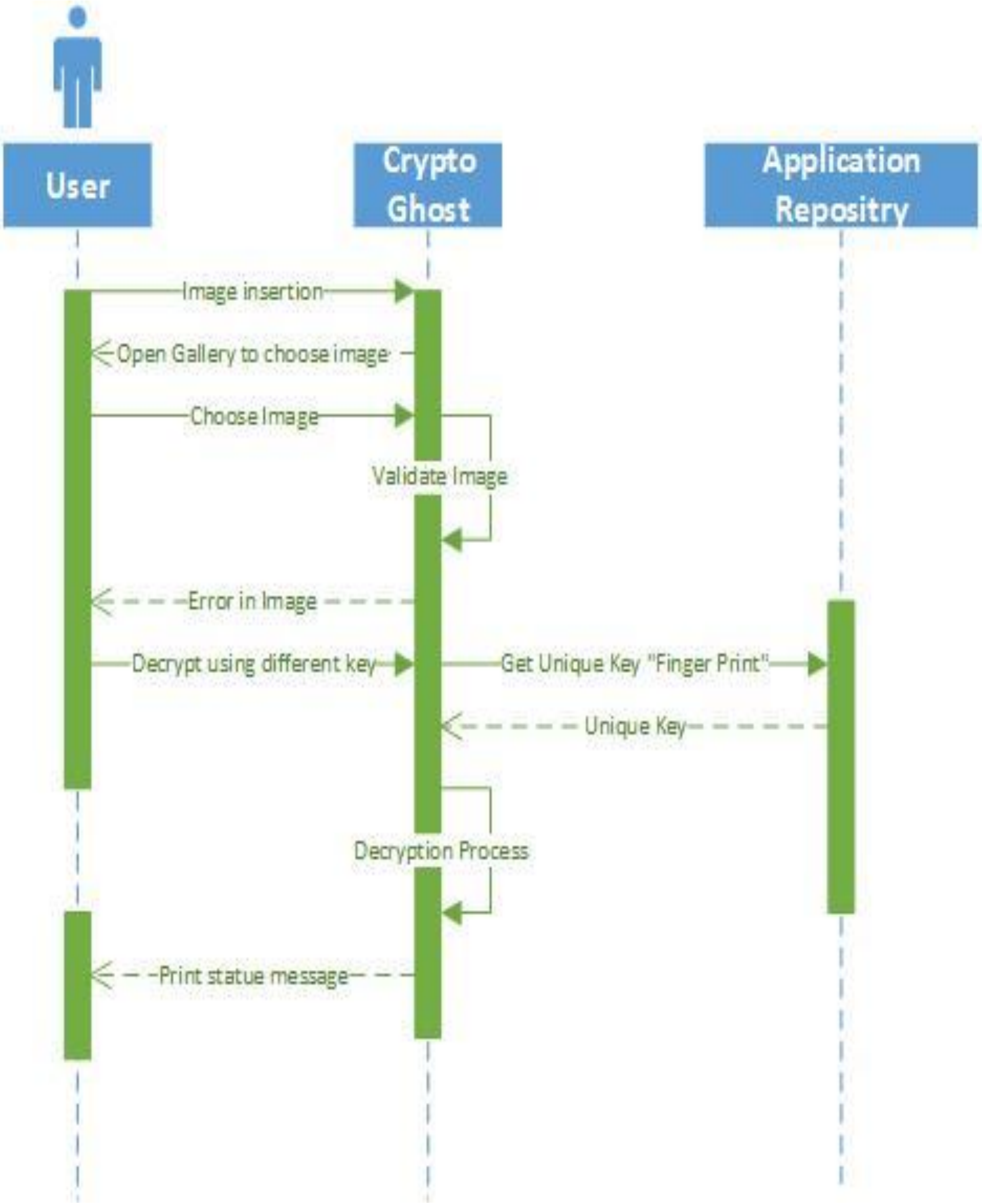
4.5.2. Login


User
Crypto Ghost
Application repositry
Main Screen
login
Get Password
Validate Password
Error password not match
Show main screen to user

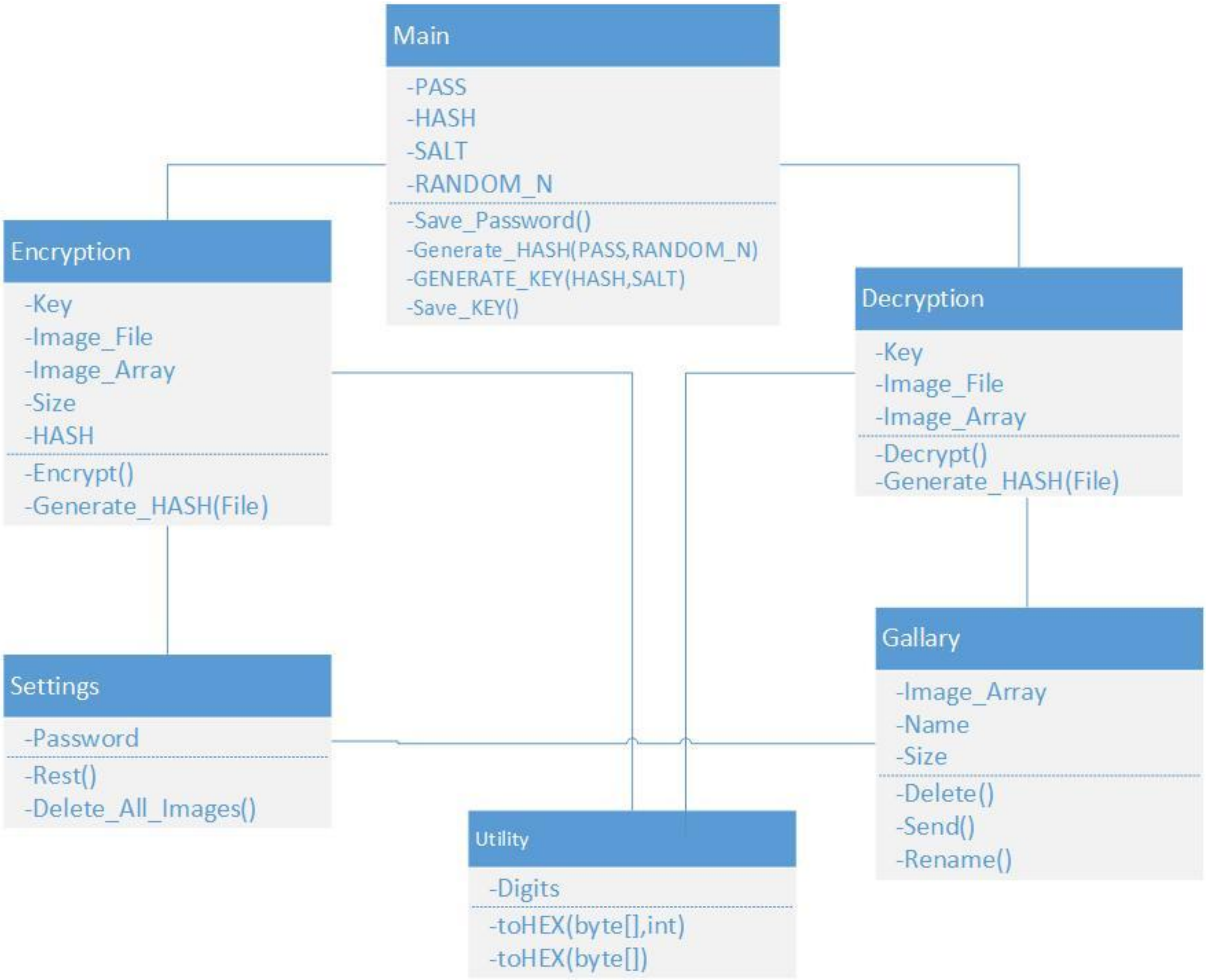
4.5.3. Encryption



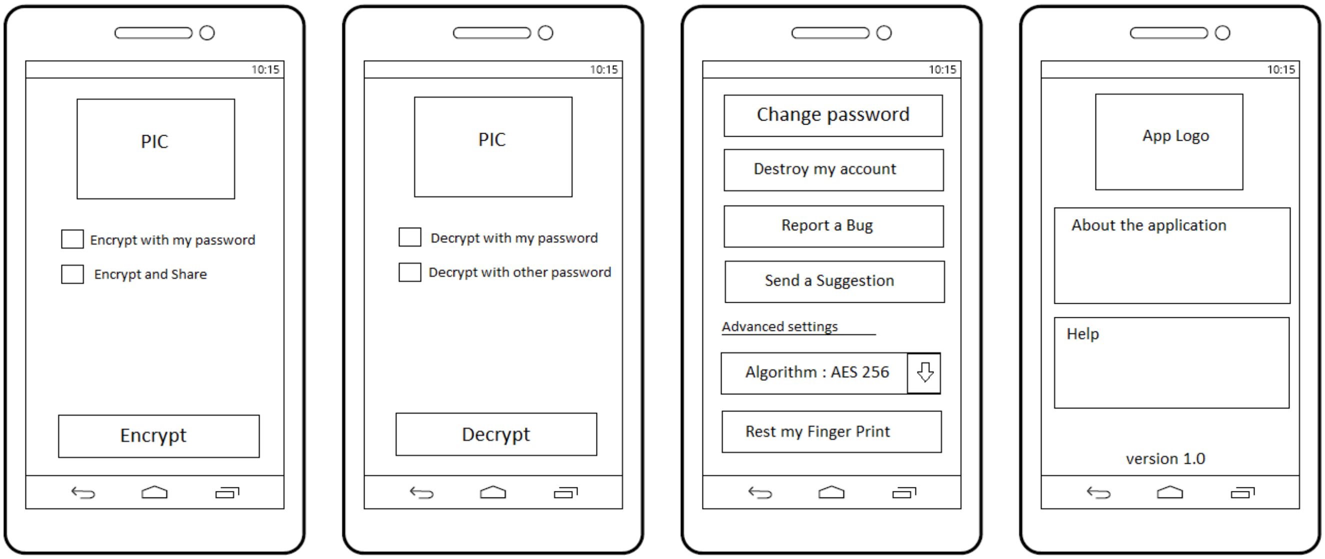
4.5.4. Decryption



4.5 Class Diagram



1 . Initial Design



Encrypt Screen

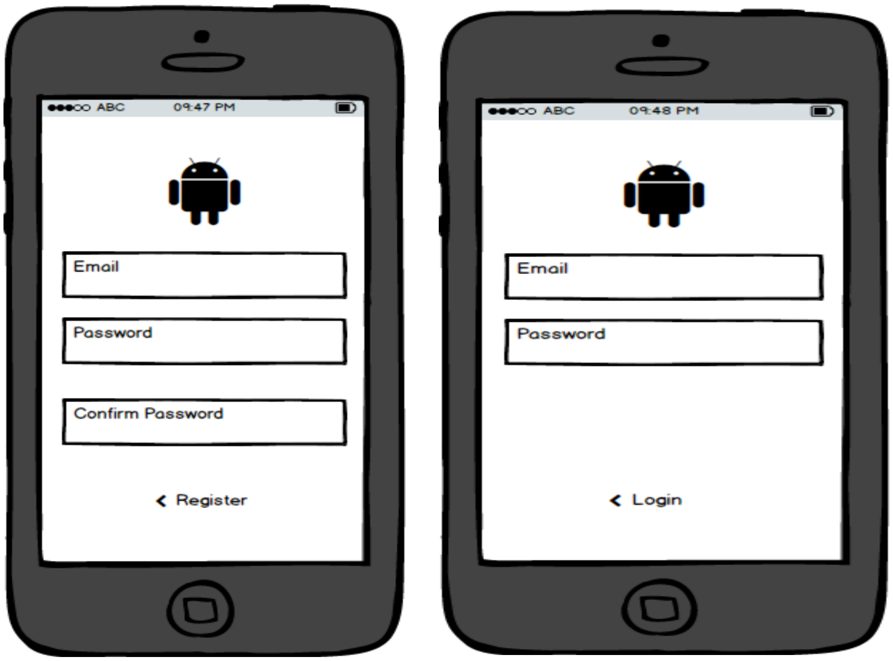
Decrypt Screen

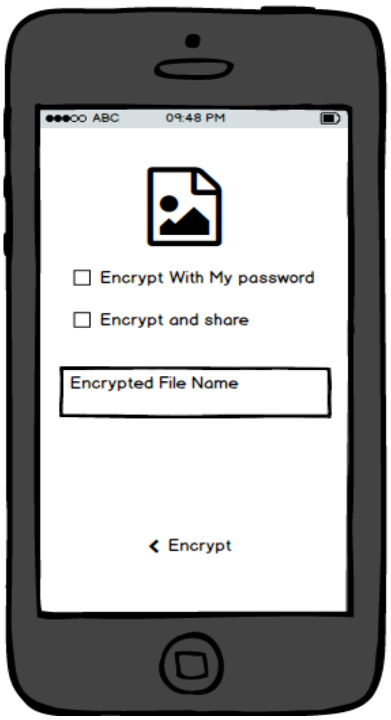
Settings

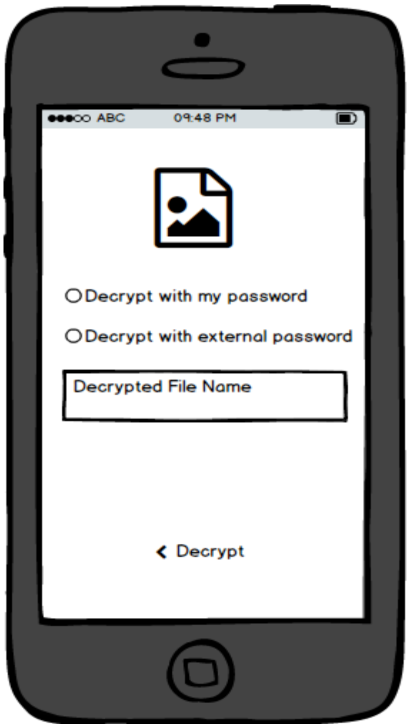
About & Help

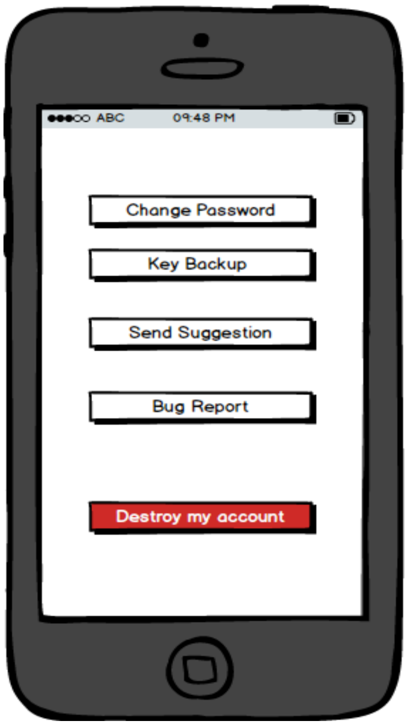
Initial Design

2 . Interface Design




●●●○○ ABC
09:48 PM
□ Encrypt With My password
□ Encrypt and share
Encrypted File Name
‹ Encrypt


●●●○○ ABC
09:48 PM
○ Decrypt with my password
○ Decrypt with external password
Decrypted File Name
‹ Decrypt

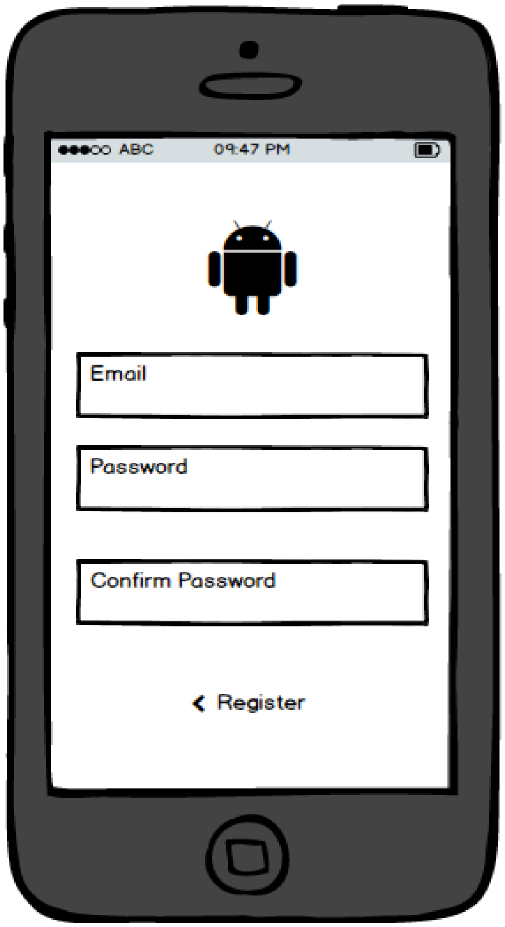

●●●○○ ABC
09:48 PM
Change Password
Key Backup
Send Suggestion
Bug Report
Destroy my account

Chapter 5 | Implementation

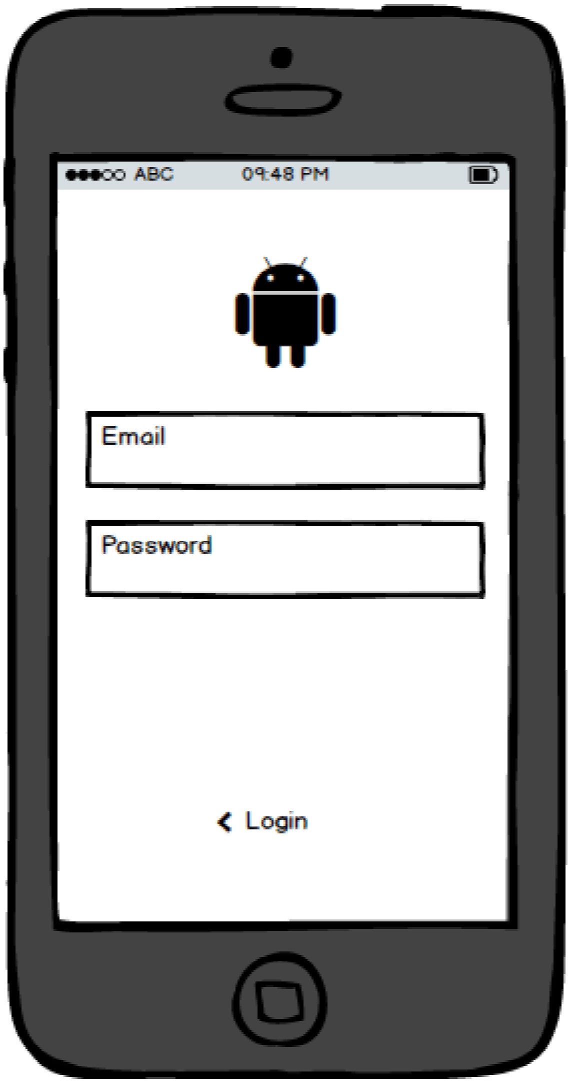
1 . Tools

Android Studio is the IDE for the application and all the coding is done on this tools

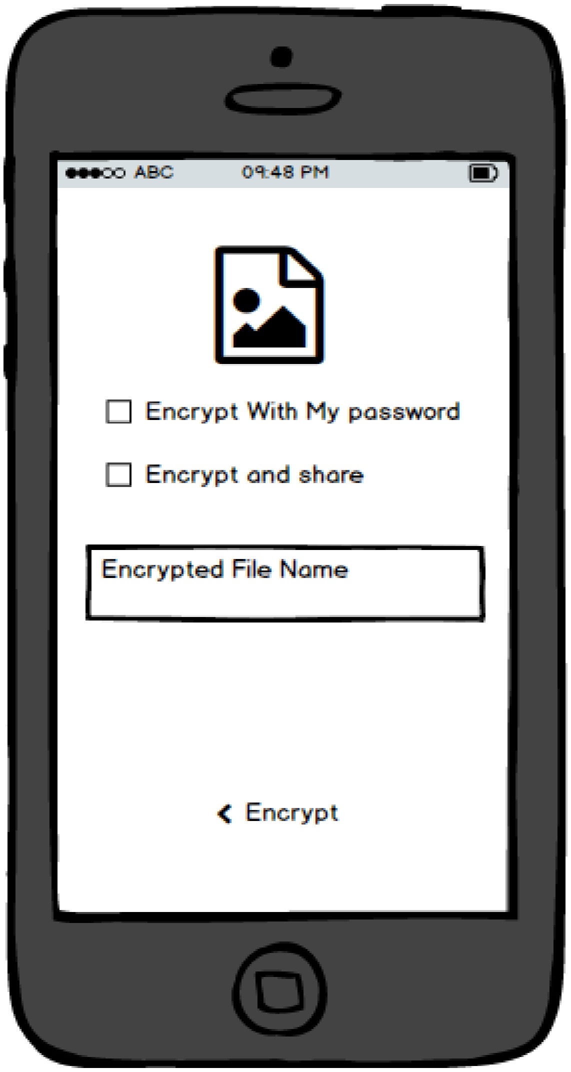
2 . Interface Description



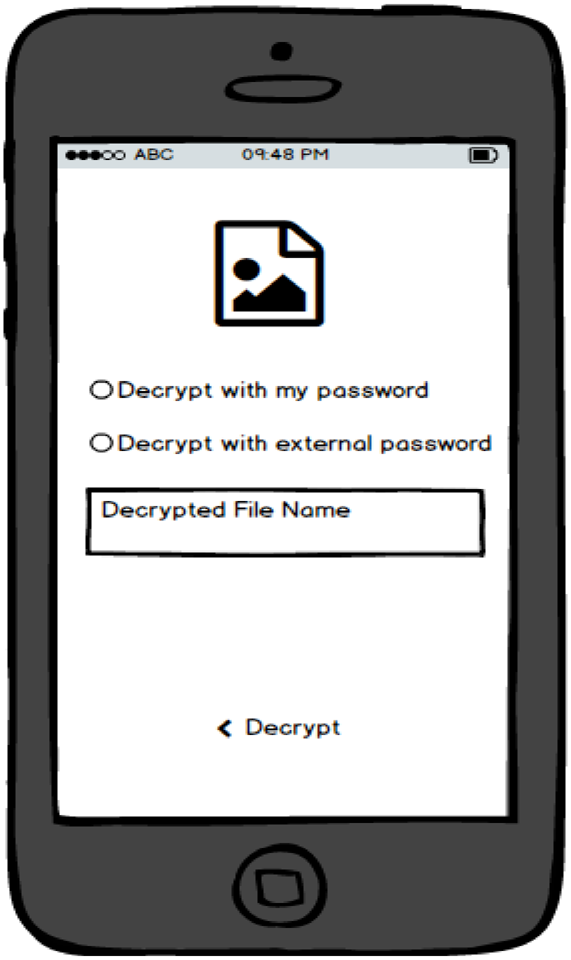
This screen will appear in the first time when the application start and the user will put an email address and a password to create his private key



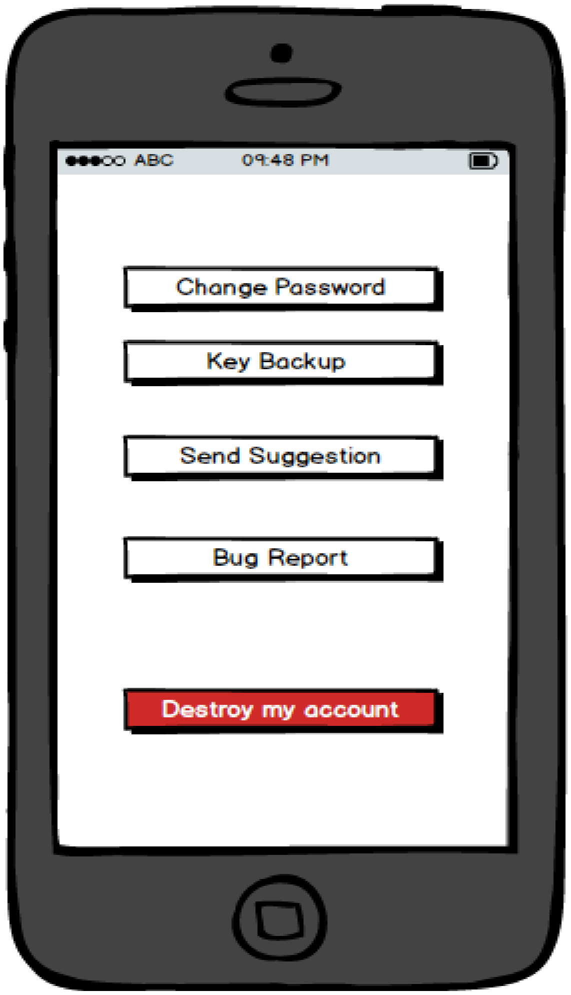
This screen will appear when the user start the application again and in this screen the user will put his email and password and the application will be validating the information



In this screen the user will insert an image and choose one or two option either “encrypt with my password “ or “encrypt and share “ and he have to put a name for the encrypted file after he finish all of this he can press “Encrypt” to encrypt the image



In this screen the user will decrypt the image.. first he has to insert the encrypted image then choose one of the options either “decrypt with my password “ or “Decrypt with external password” and the user has put a name for the decrypted file and then press “Decrypt”



In this screen the user can change the password, take backup of the key , send suggestions , bug report and destroy the account if he want .

Chapter 6 | Usability Testing

1 . Usability and Navigation testing

Encryption software's are known for their complexity and bad design that's why Crypto Ghost took this point in mind and we try I tried to make the application as easy as possible even for non-technical people …

Navigation testing I made sure that the user can go to any screen in any place in the application and I made sure that is working smoothly.

2 . Test Design

The application is compatible with different screen (small, medium, large) to make

Sure that the design is fit and have no problem in any screen.

3 . Pilot Test

I tested the application with 4 people to see the difficulty from the user perspective and to notice unexpected errors and tell now the application has no problem

4 . Results

In this application I made sure to follow the standard from the Google design perspective and I followed Google material design. I tested the application in different screen to make sure that the Interface design is fit for any type of screen. I conducted a pilot test to present a life demo for the user to see the application and to get their feedback upon the application to improve it.

Chapter 7 | Conclusion

1 . Problem and Difficulties

1.1 got so many problem regarding some cryptographic primitives and some problem in the theory

1.2 got some implementation problems, a lot of bugs, but now everything is running perfectly

2 . Feature Work

Improving the application from the usability point of you and test the system for any type of attacks

3. Conclusion

This application is for protecting our privacy so we can share our information with confidence. And I tried to design this application with modern technology. And to make this application available for any type of user not just for the technical people

Reference

[1] https://play.google.com/store/apps/details?id=pl.kchdev.image.crypto&hl=en

[2] https://play.google.com/store/apps/details?id=com.hide.hideimages&hl=en

[3] https://play.google.com/store/apps/details?id=com.samu.the.encrypter&hl=en

[4] https://play.google.com/store/apps/details?id=my.nexus673.cryptofree&hl=en

[5] https://play.google.com/store/apps/details?id=com.projectstar.timelock.android&hl=en

[6] https://play.google.com/store/apps/details?id=com.sp.smartgallery.free&hl=en

[7] Steve Krug, “Don’t make me think” , 2013

[8] Bruce Schneier,Niel Ferguson, Tadayoshi Kohno, “ Crryptogrpahy Enginerring” , ISBN : 978-0-470-47424-2, “Wiley Publishing ”, 2010