جامعة الملك سعود
King Saud University
كلية علوم الحاسب والمعلومات
قسم تقنية المعلومات

STEGO

# STEGO

IT 497: Graduation Project Report
Product Release-2

Prepared by
Raneem Jabour, 441200274
Layan Alshowiman, 439200887
Nouf Almutlaq, 441201273
Sarah AlFaris, 441201219
Lamya almutairi, 437204055

Supervised by
Dr. Kholoud Saad Alsaleh

Second Semester 1444
2022/2023

# Table of Contents

# STEGO

*Raneem Jabour [1], Layan Alshowiman [2], Nouf Almutlaq [3], Sarah Alfaris [4] and Lamya almutairi [5]*

[1]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 441200274@student.ksu.edu.sa
[2]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 439200887@student.ksu.edu.sa
[3]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 441201273@student.ksu.edu.sa
[4]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 441201219@student.ksu.edu.sa
[5]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 437204055@student.ksu.edu.sa

**Abstract (English):**
More security is needed as Internet usage expands globally. Most Internet users are unaware of the potential risks, especially if they use the Internet to send sensitive data, so we developed our application to help users share secret messages as they innocent messages to protect their information. Our application was created using the agile methodology which is one of the most popular approaches to project management. We finally ended up with an android STEGO application that enables the user to share a hidden secret message in an image and does not raise any suspicion and only the intended receiver knows that there is a hidden message and can decrypt it.

**Abstract (Arabic):**
هناك حاجة إلى مزيد من الأمان مع توسع استخدام الإنترنت على مستوى العالم. لا يدرك معظم مستخدمي الإنترنت المخاطر المحتملة، خاصةً إذا كانوا يستخدمون الإنترنت لإرسال بيانات حساسة، لذلك قمنا بتطوير تطبيقنا لمساعدة المستخدمين على مشاركة الرسائل السرية وكأنها رسائل بريئة خالية من السرية لحماية معلوماتهم. تم إنشاء تطبيقنا باستخدام منهجية "اجايل" التي تعد واحدة من أكثر الأساليب شيوعًا لإدارة المشاريع.
لقد انتهينا أخيرًا مع تطبيق اندرويد ستيقو الذي يمكّن المستخدم من مشاركة رسالة سرية مخفية في الصورة بدون إثارة أي شك ولا يعلم إلا المتلقي المقصود أن هناك رسالة خفية ويمكنه فك تشفيرها.

**Keywords:** Steganography; Cryptography; Encryption; Decryption; Android.

# List of Figures

# List of tables

# Introduction

# 1 Introduction

Given the amount of data that is being generated and transmitted electronically in the world today, lots of people want to keep this communication invisible for several reasons.

It is no surprise that numerous methods of protecting that data have evolved. One of the rapidly growing methods is steganography. Steganography is the practice of concealing a secret message behind a normal message. Understanding how to hide information and prevent it from misuse or disclosure is very useful for people, organizations and those who want to hide their information, so that it will not be detected, and this is what Steganography offers us. Depending on the nature of the cover object (actual object in which secret data is embedded), steganography can be divided into types: Text Steganography, Image Steganography, Video Steganography and Audio Steganography [1,2].

Since many people and organizations need to use this technique for many reasons, we came up with the STEGO application that allows users to exchange confidential information and data and communicate with each other without drawing any attention plus without the need to hide the entire communication from everyone, even if these messages are in front of them, no one will know that these messages carry private and confidential information, they appear as innocent messages.

Agile project management, one of the most well-liked methods for organizing projects into phases, was the methodology used to create the STEGO program. It is an iterative strategy that makes use of ongoing planning, learning, and improvement, teamwork, evolutionary development, and early delivery. Additionally, it promotes adaptable reactions to change.

In this document we cover the first touches to solving this problem, starting with identifying and describing the problem and then developing a solution. Drafting the product vision and setting objectives in addition to the scope, background, literature review, and then we started designing the product roadmap and developing the system, after that we tested the system in system evaluation, and at the end, we concluded with conclusion and future work.

## 1.1 The Problem

As the global use of the Internet grows, more security is required. Most Internet users don't give their online privacy much thought and are unaware of the potential threats. Your safety and privacy are both at risk, especially if you use the Internet to do sensitive duties like online

banking and sharing vital business data. However, if the information shared is not securely conveyed to our intended recipients, it may result in a privacy violation or public humiliation from which we may need a long time to recover. The information security issue also particularly impacts big and small businesses whose success depends on having an online presence. Therefore, steganography is practiced by those wishing to convey a secret message or code.

Our application is designed to allow users to exchange secret messages embedded in images - as an innocent image- without drawing any attention or suspicion using steganography technique.

## 1.2 The Solution

To answer the question of how to convey a message covertly to its intended recipient, STEGO addresses the security issue associated with data transmission across Internet networks. STEGO will allow the user to use the camera on their phone to snap a picture or select one from their library of previously taken pictures, and then they can add a secret message to it. The hidden message is then sent embedded in the image after that. The hidden message in the image can only be seen by the intended recipient who also possesses a shared secret key. The existence of a secret message is unknown to other users.

## 1.3 Product

### 1.3.1 Product Vision

For a person, who needs to hide his secret message, STEGO is a mobile application, that conceals a message within images, unlike other steganography tools, our product allows the sharing of secret key using asymmetric key cryptography to increase the security.

### 1.3.2 Objectives

- Product (customer focus-value)
    - Provide the registration feature in the application for the users.
    - Provide the ability to log in /log out for the users.
    - Provide the ability to add a new friend to exchange images.
    - Provide secret key sharing using asymmetric key cryptography to increase security.

- Provide the ability to upload or take a photo to hide messages.
- Provide the ability for users to share images.
- Provide the encoding feature to the sender to hide a secret message within the images.
- Provide the ability to delete a friend.
- Provide the decoding feature to the recipient to get the secret message.
- Provide the ability to receive notification for receiving images, receiving friendship requests and for accepting /declining the friendship request.
- Provide the ability for users to view or edit their profile's information.
- Provide the ability for users to update their shared secret key.

- Project (solution focus-plan)
  - We need to read more about steganography and how to deal with text.
  - We need to understand the user's needs and how to embed his secret message.
  - Building a mobile application framework using visual studio, to allow the sender to encode messages and then the receiver decodes the messages.
  - We need to test the application to see if it achieves the user's needs.
- Learning (student focus)
  - We will learn to program using flutter framework.
  - Using a new IDE for building an app using visual studio.
  - Learn how steganography works with text using technology and encryption key, least significant bit (LSB).

### 1.3.3 Scope

STEGO is an android mobile application that aims to help users who need to hide their messages. It will embed a secret message within images and enable the sender to encode messages and the recipient will be able to decode the message. In addition, STEGO allow the sharing of secret key using asymmetric key cryptography to provide a secure channel and its supports English only.


Background

# 2 Background

In this chapter we will define steganography, explain the difference between steganography and cryptography, present the different kinds of cryptography and what we will use in this project.

## 2.1 Definition of steganography

The concept of "What You See Is What You Get (WYSIWYG)" which we encounter sometimes while printing images or other materials, is no longer precise it does not always hold true. Images can be more than what we see with our Human Visual System (HVS); hence, they can convey a lot of messages embedded in them [3]. Steganography is considered the art and science of concealing the very existence of information by hiding it in another information, and its techniques have been in use for hundreds of years. People have hidden information by a multitude of methods and variations. For example, ancient Greeks wrote text on wax-covered tablets. To pass a hidden message. Another method was to shave the head of a messenger and tattoo a message or image on the messenger's head. After the hair grew back, the message would be unseen until the head was shaved again [4].

## 2.2 Steganography & Cryptography

Steganography and cryptography are techniques that are used to secure the communication of a secret message. Steganography hides the secret message in ways that are invisible. Therefore, it conceals the fact that a communication is taking place. Because it doesn't require encrypting data with a key or scrambling data, it is not a type of cryptography. Instead, it is a method of data concealment that may be carried out ingeniously. Only the intended recipients should be the only ones who are aware of any hidden communication when steganography is performed correctly. This makes it a practical strategy for circumstances where direct contact is unsafe.

On the other hand, cryptography is the act of scrambling plaintext into unintelligible cipher-text using an encryption algorithm. However, it could appear to others that an encrypted message does exist. Both sciences can be combined to produce better protection of the message. In this case, when the steganography fails and the message can be detected, it is still of no use as it is encrypted using cryptography techniques [5]. There are different kinds of steganography depending on the cover object used to hide information. Steganography can hide information in almost all digital file formats such as images, audio, video, and text files. Images are the

most popular way used for steganography. So, in this project we will be focusing on hiding information in image files [6].

All available methodologies used for steganography are divided into three groups: Generative Adversarial Networks (GANs), Convolutional Neural Networks (CNNs), and traditional image steganography. Many traditional methods are based on the Least Significant Bits (LSB) technique. The LSB method hides the message by inserting the message at the lower or rightmost bits in the cover work file as a medium to hide the message, it's the easiest way to embed secret information and operates well for image steganography.

To increase the detection difficulty of secret data, a pseudorandom sequence can be used to control the location, into which the secret binary information is going to be embedded. Although poor in security, the LSB method is simple and easy to implement, embeds and extracts information fast, and has a high hiding capacity. However, this technique is not based on machine-learning or deep-learning algorithms. On the other hand, CNN-based techniques rely upon deep convolutional neural networks to embed and extract secret messages. Finally, GAN-based methods utilize some GAN variant [8].

## 2.3 Symmetric & Asymmetric Key Cryptography

Before we use steganography on the image file, first we will use cryptography to encrypt the message intended to hide. There are two kinds of cryptography, symmetric and asymmetric cryptography. Symmetric cryptography is also known as the shared secret key cryptography, where a single key is used for both encryption and decryption. The sender encrypts the plain-text using the shared secret key and the receiver decrypts the encrypted message to obtain the plain-text using the same shared secret key. The shared secret key should be only known by authorized people. On the other hand, asymmetric cryptography, also known as the public key cryptography, employs two different keys, one for encryption and the other for decryption of the message. The first key which is used for the encryption is kept public and it's called the public key, while the other key used for the decryption is kept secret and it is never shared, called the private key [5].

There are different kinds of algorithms used for symmetric encryption, including DES and AES algorithms, which are two of the most widely used algorithms [9]. Both two algorithms are a symmetric key block cipher which means that it encrypts blocks of data in rounds. Data Encryption Standard (DES) was developed in 1972 by IBM, it was created due to a lack of

cryptographic standards. It uses a 56-bit key length and a 64-bit block size. DES was adopted by the USA government as standard symmetric ciphers in 1976. However, it is vulnerable to brute-force key attack today, where every possible key is generated and attempted [10,11]. The advanced encryption standard (AES) is the current US standard in symmetric ciphers. It was published by NIST in 2000. Its main purpose was to replace the DES algorithm, due to some vulnerable aspects of it. There are three versions of the AES algorithm depending on length of the key (128-bit, 192-bit, and 256-bit) to encrypt 128-bit blocks of data, this versatility can produce faster and more secure symmetric block ciphers. AES has the best ability to protect sensitive data from attackers and has not allowed them to break the encrypted data as compared to other proposed algorithms [12-15].

For the encryption of the secret message, we will use AES symmetric cryptography. Symmetric cryptography uses a shared secret key between the sender and the receiver, the secret key must be kept secret in both ends. However, sharing the secret key between sender and receiver is an issue since if the communication medium that is used to share the secret key is compromised it will lead to the compromise of the secret key and the data encrypted using it. Thus, we use asymmetric cryptography when communicating the shared secret key to increase security [7].

There are two sides in an encrypted communication: the sender, who encrypts the data, and the recipient, who decrypts it. As the name implies, asymmetric encryption is different on each side; the sender and the recipient use two different keys. Asymmetric encryption, also known as public key encryption, uses a public key-private key pairing: data encrypted with the public key can only be decrypted with the private key.

Commonly asymmetric encryption algorithms include RSA. It depends on the factorization problem. The public key is the multiplication of two very large prime numbers. The private key is also derived from the same two prime numbers. So, if somebody can factorize the large number, the private key is compromised. Therefore, encryption strength totally lies on the key size and if we double or triple the key size, the strength of encryption increases exponentially. RSA keys can be typically 1024 or 2048 bits long, but experts believe that 1024-bit keys could be broken soon. But till now it seems to be an infeasible task [16].

Both the public and the private keys can encrypt a message using RSA cryptography. A message is decrypted using the opposite key to that which was used to encrypt it. Because it offers a way to ensure the privacy, integrity, validity, and non-repudiation of electronic

communications and data storage, RSA has grown to be the most popular asymmetric algorithm. Because it is challenging to factor huge integers that are the product of two large prime numbers, RSA is secure. These two numbers can be easily multiplied but extracting the original prime numbers from the sum is thought to be impossible given the time required, even with today's supercomputers [17].

We utilized a couple of libraries in our code to implement all the above. First, to generate the public and private keys for our users we used the "crypton" package, which is a library that supports RSA keys generation. For the secret message encryption, we first had to generate a shared secret key by using the “flutter_aes_ecb_pkcs5” package, which is a library that supports AES 128-bit symmetric key generation. And by using the “steganograp” package we implemented steganography and symmetric encryption, which is encrypting the secret message using a generated secret key, and then encoding the secret message into the image.


Literature
Review

# 3 Literature Review

In this chapter we will introduce several studies that are related to our field of study.

People have concealed information throughout history using a wide range of techniques. For instance, text was written on wax-coated tablets by the ancient Greeks. Also, a person would scrape wax off a tablet, write a message on the underlying wood, and then cover the tablet once more with wax to make it seem empty and unused to pass a secret message. Another brilliant technique involved shaving a messenger's head and tattooing a message or image on it, the message would not be noticed once the hair grew back until the head was shaved once more. A popular type of invisible writing was made possible by invisible inks, these inks were the mainstay of steganographic technology throughout the initial stages of World War II it means that a letter that appears innocent could contain a quite different message written between the lines [4].

After examining relevant literature, we discovered that there are different kinds of steganography depending on the cover object used to hide information. Steganography can hide information in almost all digital file formats such as images, audio, video, and text files. Images are the most popular way used for steganography [6].

As we said earlier in the background, steganography consists of three categories: Generative Adversarial Networks (GANs), Convolutional Neural Networks (CNNs), and traditional image steganography. Least Significant Bits (LSBs) are used in many traditional methods. With the LSB method, the message is hidden by placing it in the lower or rightmost bits of the cover file, which makes it the easiest method for embedding secret information and for imaging steganography [8].

## 3.1 Competitive Product Analysis

This competitive analysis focused on applications with similar features to the STEGO application. The goal is to investigate and learn from the functions of competitors to avoid making the same mistakes.

1- NoClue

NoClue is a free android mobile Application. Through this Application, the user can hide some secret text information inside any image file and reveal them using a password whenever he wants. It does not allow the user to share images within the application or across platforms like WhatsApp, telegram, or other platforms. Furthermore, sending the shared secret key is not available through this application.

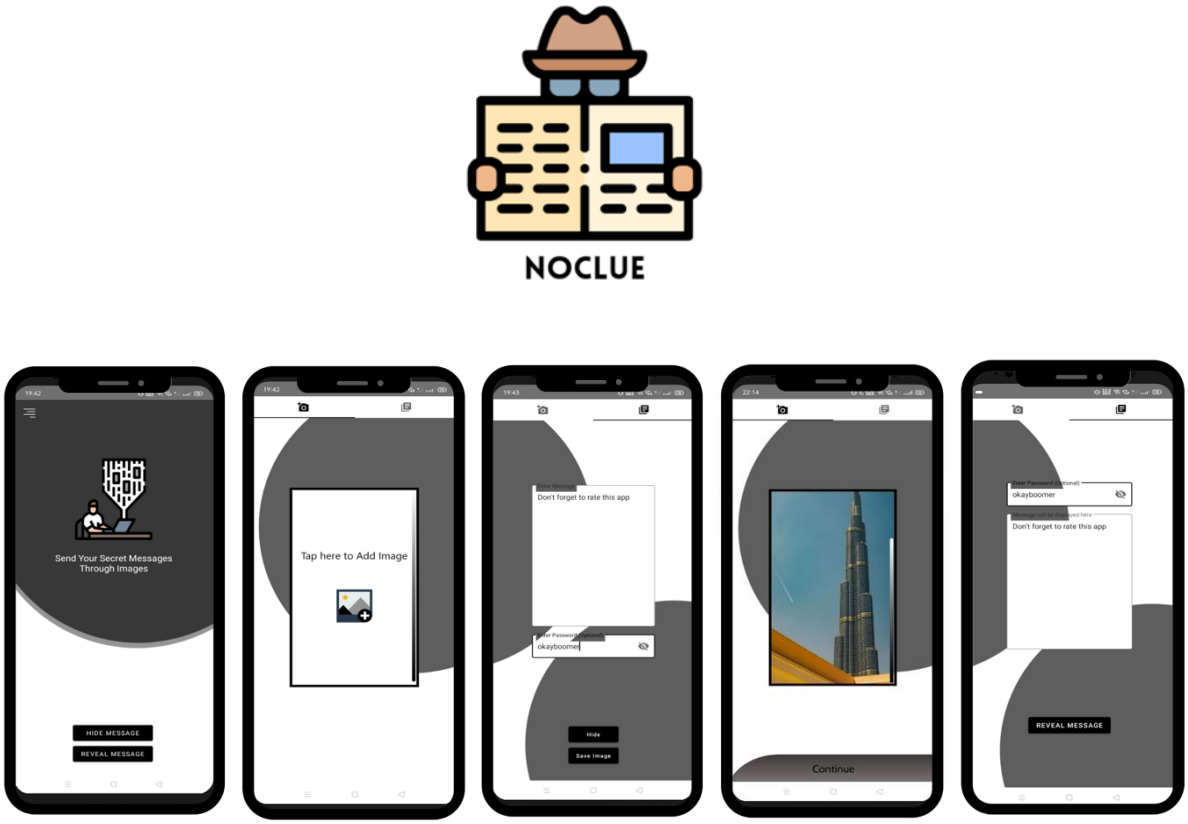


Figure 1: NoClue application

2- PixelKnot

PixelKnot is a free android mobile application that helps the user to hide messages in images and share them with his friends using the shared secret password that can unlock the hidden message. It does not allow the user to send the shared secret key through the application. In addition, the user cannot be able create an account and receive notification from the application.

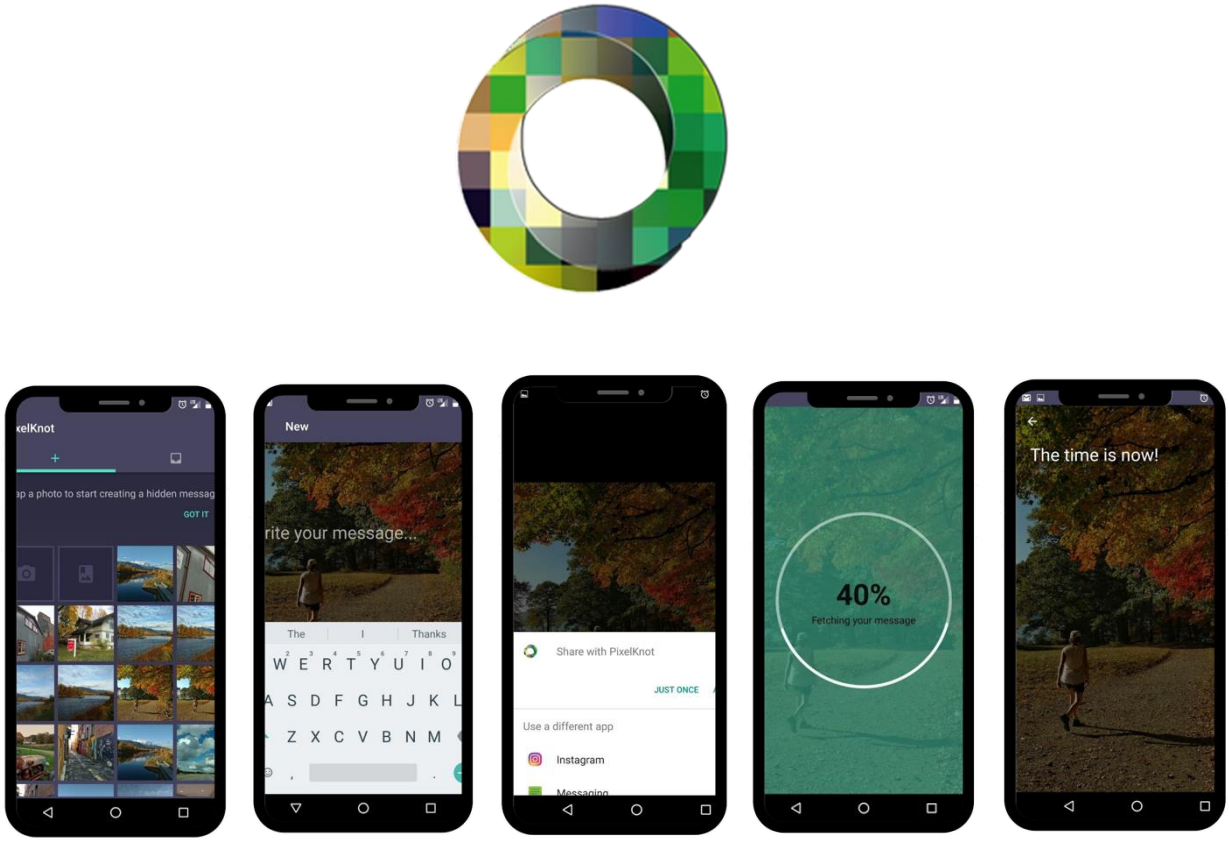


Figure 2: PixelKnot application

3- StegoMagic

StegoMagic is a non-free iOS application that helps the user to hide information using images, without any visual changes in the image and without using a secret key. It does not provide the feature of sharing the image from within the application.

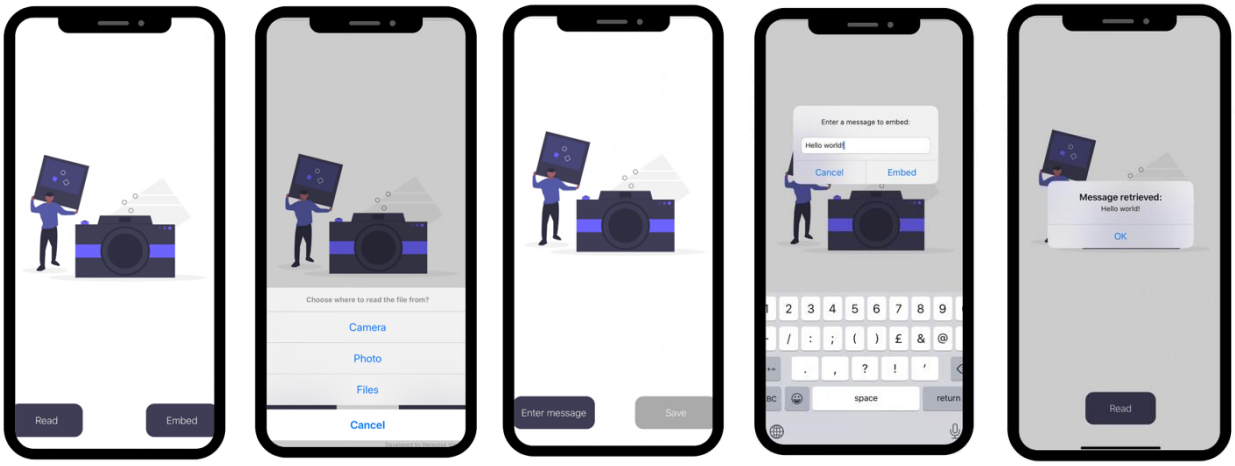


Figure 3: StegoMagic application

4- kosh-invisible storage

Kosh is a free iOS application it takes a message from the user and hides it into an image. The user can encode and decode the message into an image within the application. Also, the image can only open with a secret key. It does not provide the feature of sharing the image from within the application. In addition, it does not allow the user to create an account in the application.

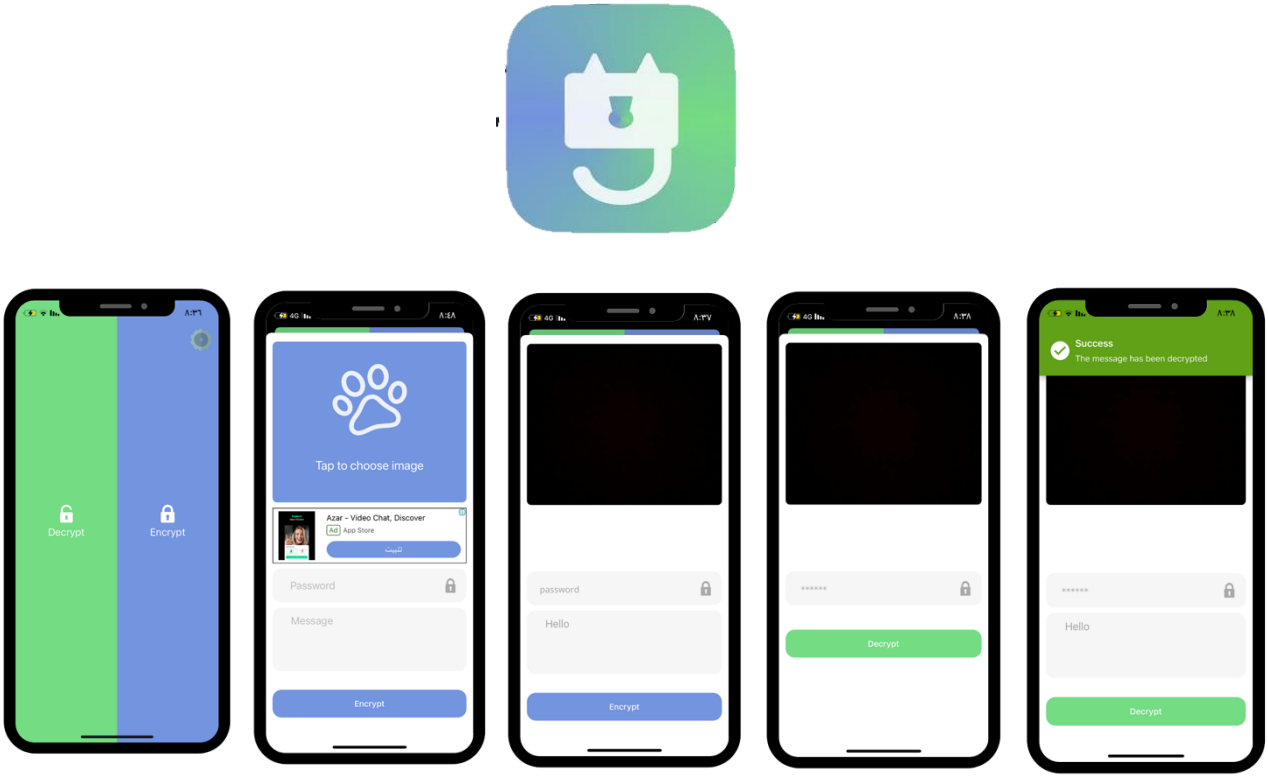


Figure 4: Kosh application

5- Phidden

Phidden is a free iOS application that helps the user to send secret messages to his friends and let others believe that they have only sent them a photo. The user only needs to select a picture, write their message, and then Phidden will hide it inside the image so they can share it. Whoever receiver will be able to read the shared message just by opening it with Phidden and without any other application or software.

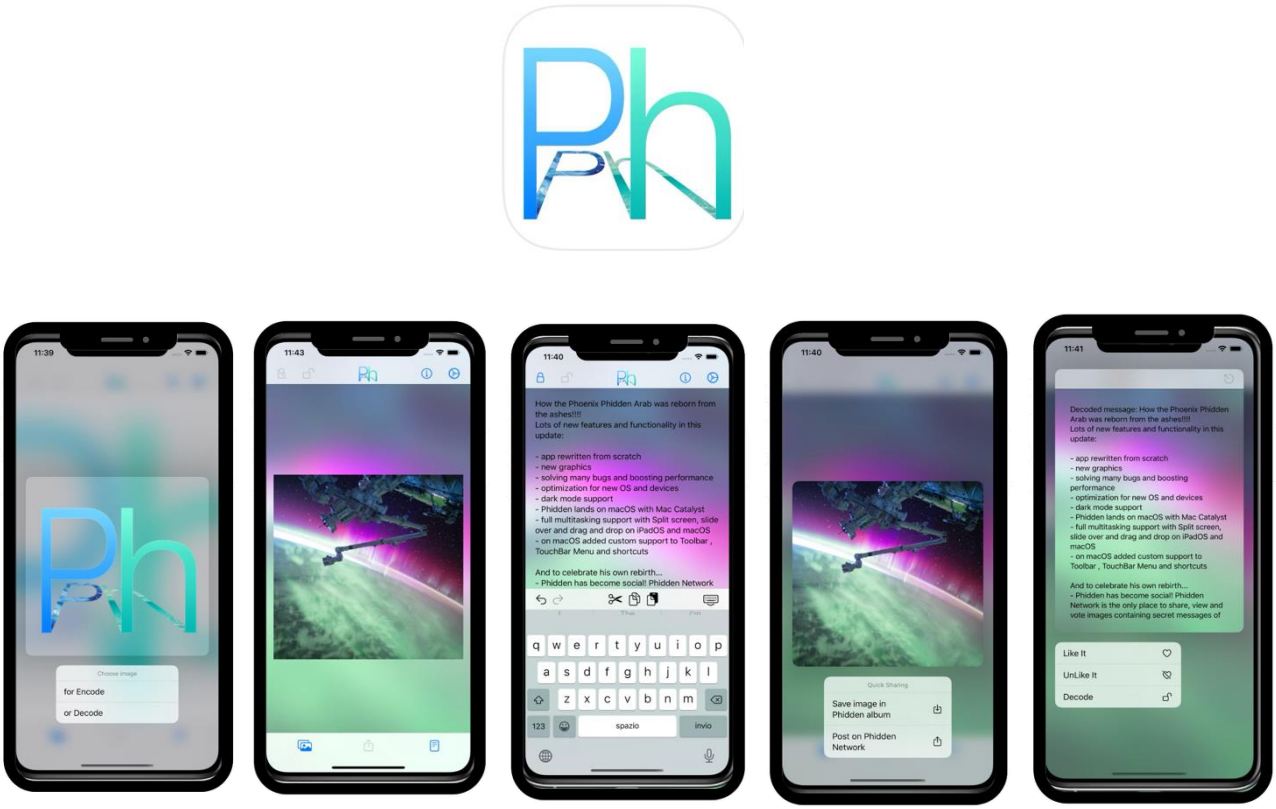


Figure 5: Phidden application

The main features of the previously mentioned applications are shown in figure 6.

| FEATURES | Phidden | Pixelknot | NoClue | StegoMagic | Kosh - Invisible Storage | STEGO |
|---|---|---|---|---|---|---|
| ABILITY TO UPLOAD OR TAKE A PHOTO | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| ENCODE SECRET MESSAGE IN IMAGE | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| DECODING IMAGE TO GET THE SECRET MESSAGE | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PROVIDE THE REGISTRATION FEATURE | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ |
| USE THE SHARED SECRET KEY TO ENCODE / DECODE THE MESSAGE | ✗ | ✓ | ✓ | ✗ | ✓ | ✓ |
| SHARE WITHIN THE APPLICATION | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ |
| SHARE SECRET KEY USING ASYMMETRIC KEY CRYPTOGRAPHY | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ |
| ABILITY TO ADD A NEW FRIEND OR DELETE FRIEND | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ |
| ABILITY TO RECEIVE NOTIFICATION | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ |
| COST FREE | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ |
| UPDATE SHARED SECRET KEY | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ |
| IOS / ANDROID | IOS | ANDROID | ANDROID | IOS | IOS | ANDROID |

Figure 6: Comparison between 5 applications and the proposed application

- Discussion:

Steganography applications are one of the most important today due to the frequent cyberattacks and the need for users to transfer their data or send it securely within images without raising doubts. So, after studying similar applications to our STEGO application we found that all previously proposed apps provide the feature of uploading an image that already exists on the phone or taking an image from the camera directly and encoding messages within the images and decoding it through the application which means it is an important feature so many users are looking for an application that provides these features. Also, one of the advantages of some applications is the use of a secret key to encode/decode the messages and share images using platforms. However, the unique feature of our STEGO application is that the message can only be decoded or encoded from within the STEGO application and that it can only be opened using a shared secret key that is sent to the other party via the application. In addition, the user will be able to register in STEGO and add friends to share the images with them. Also, the application will be able to notify the user when a new secret message has been sent by his friends.

# System Design and Development

# 4 System Design and Development

This chapter covers the methodology. It discusses the agile approach, scrum framework and tools used to develop the application.

## 4.1 Methodology

STEGO application was developed by following agile as a methodology, which is one of the most popular approaches to project management by breaking the project into several phases. It is an iterative approach that employs continuous planning, learning, improvement, team collaboration, evolutionary development, early delivery. and it encourages flexible responses to change.

Scrum is a framework consisting of roles, events, and artifacts. Combined, these elements form an agile project management methodology, enables teams to better manage their work. The Scrum methodology requires people to work on the project. There are three Scrum roles defined as: Scrum Master, Product Owner, and Development Team. The Scrum Master is responsible for making sure that the Scrum Team adheres to the values and principles of Agile methodology. Additionally, the scrum master ensures the adherence to the processes and practices that the team agreed they would use. The Scrum Product Owner is responsible for converting requests from the customer into actionable user stories, manages the product log, oversees sprint planning, and attends Scrum meetings. While the scrum development team is made up of all team members who work on developing software or products.

For scrum events, there are five events: sprint planning, daily scrum, sprint evaluation, sprint improvement, and the sprint itself. Events are intended to foster team collaboration and ensure that there is a constant line of communication between Scrum team members throughout the product or software development lifecycle.

Scrum artifacts help track the progress of the Scrum team on a given Scrum project. The three main Scrum artifacts are the product backlog, sprint backlog, and product increment. The product backlog is a prioritized list of features, defects, or technical work that has yet to be worked on, the sprint backlog is a list of all items from the product backlog to be worked on during a sprint. This list is put together by prioritizing items from the product backlog until the team feels they’ve reached their capacity for the sprint. And the product increment is the sum

of product work completed during a sprint, combined with all work completed during previous sprints.

Table 1 shows STEGO scrum team.

Table 1 STEGO Scrum Team

| **Scrum Team** | |
|---|---|
| Product Owner: (Project supervisors) | Dr. Kholoud Saad Alsaleh |
| Developers: (Student names) | Raneem Jabour<br>Layan Alshowiman<br>Nouf Almutlaq<br>Sarah AlFaris<br>Lamya almutairi |
| Scrum Master: | Dr. Hend Alrasheed |
| Stakeholders: | Project Committee. |

We used Jira to manage all the files of our project such as weekly meeting notes, backlog, and documents. We also used GitHub to upload and modify the code for the application [1][2]

[1] https://2022-gp22.atlassian.net/jira/software/projects/GP22/pages

[2] https://github.com/LayanSM1/2022-GP1-22

## 4.2 System Requirements

### 4.2.1 System Users

The users of our application are people who need to exchange their secret messages with others. Their characteristics are as follows, ages are 16 years and above, their educational level can be from high school and above. Familiar with using Android applications, know how to read and write in English, have an Android operating system device to run the app and have basic technical skills to work with the app.

### 4.2.2 Requirements Elicitation and Analysis

Understanding user needs and behavior is important in developing solutions that satisfy these needs. A survey technique is an efficient, adaptable, and rapid method for determining user needs and requirements. The second method is to interview people who specialize in this field, it is the most effective technique for allowing the interviewee to confirm their understanding of the questions and provide a detailed response.

Our survey was created using Google Forms. Furthermore, we conducted interviews in order to get a better understanding of how and whether people were interested in sending secret messages disguised as innocent messages.

- Survey Analysis

We created the survey using Google Forms to survey Arabic speakers who are interested in exchanging secret messages; the survey consists of 7 questions (attached in Appendix B) and was distributed via social media channels; 85 responses were received. To help us to understand our users' behavior and how much they are interested in hiding secret messages/information. We found that 61.2% don’t know the steganography term, 9.4% know about it and 29.4% might know about it. For difficulties they face when hiding messages, we found that 29.4% of users face these difficulties. As for the number of times they needed to hide their messages, the percentages were as follows 49.4% 1-3 times a month,25.9% more than 3 times a month, and 24.7% 1-3 times a week. As for the use of steganography, it is used more in both work and personal communications fields. 91.8% believes that making a steganography application will make communication and exchange secret messages better. 80% of users they confirmed would use our app to exchange messages containing their confidential information.

We conclude that there are users who need to send several secret messages in a short period of time, and they think that it is better to use a program that provides them with this feature, as they need to use it, especially in the personal and work fields.

- Interview Analysis

Interviews were conducted with Mr. Hmoud Alhalmani, Dr. Alia Alabdulkarim, Mr. Fahd Al-Duraibi, and Dr. Arwa Al-Sultan. The interview consists of 5 questions, which are (attached in Appendix A).

Mr. Hmoud Alhalmani discussed the importance of steganography and cryptography, and how to safeguard sensitive information from being accessed by unauthorized persons. Dr. Alia Alabdulkarim explained that steganography means hiding the existence of a message (or malware) using a host file, and how to prevent unauthorized parties from accessing sensitive information depended on the type of information and where it is kept. She also discussed the considerations she makes regarding the app's security, specifically when selecting whether to use it or not, such as secure transmission, secure key generation and storage, and most importantly, secure and strong algorithm.

Mr. Fahad Alduraibi, Dr. Alia Alabdulkarim, and Dr. Arwa Alsultan discussed the importance of steganography and cryptography. Mr. Fahad Alduraibi stated that steganography is important for specific practices but not for all. Dr. Arwa Alsultan responded to the first question by stating that steganography is a method for concealing a message in a photo. Mr. Hmoud Alhalmani, Dr. Alia Alabdulkarim, Mr. Fahad Alduraibi, and Dr. Arwa Alsultan all agreed that steganography and cryptography worked well together. Mr. Hmoud Alhalmani stated that steganography is important for specific practices but not for all.

At the end of this chapter, we analyzed and compared applications in the market that are similar to the proposed application to give a vision of possible features that could be added. We will propose a mobile application that will allow users to hide secret messages in images using a shared secret key and share them within the application in a secure environment for data transmission and notify users when any other user shares an image with them. We analyzed the interview with Mr. Hmoud Alhalmani, Dr. Alia Alabdulkarim, Mr. Fahad Alduraibi, and Dr. Arwa Alsultan and the survey that was distributed through social media channels.

### 4.2.3 User Interactions

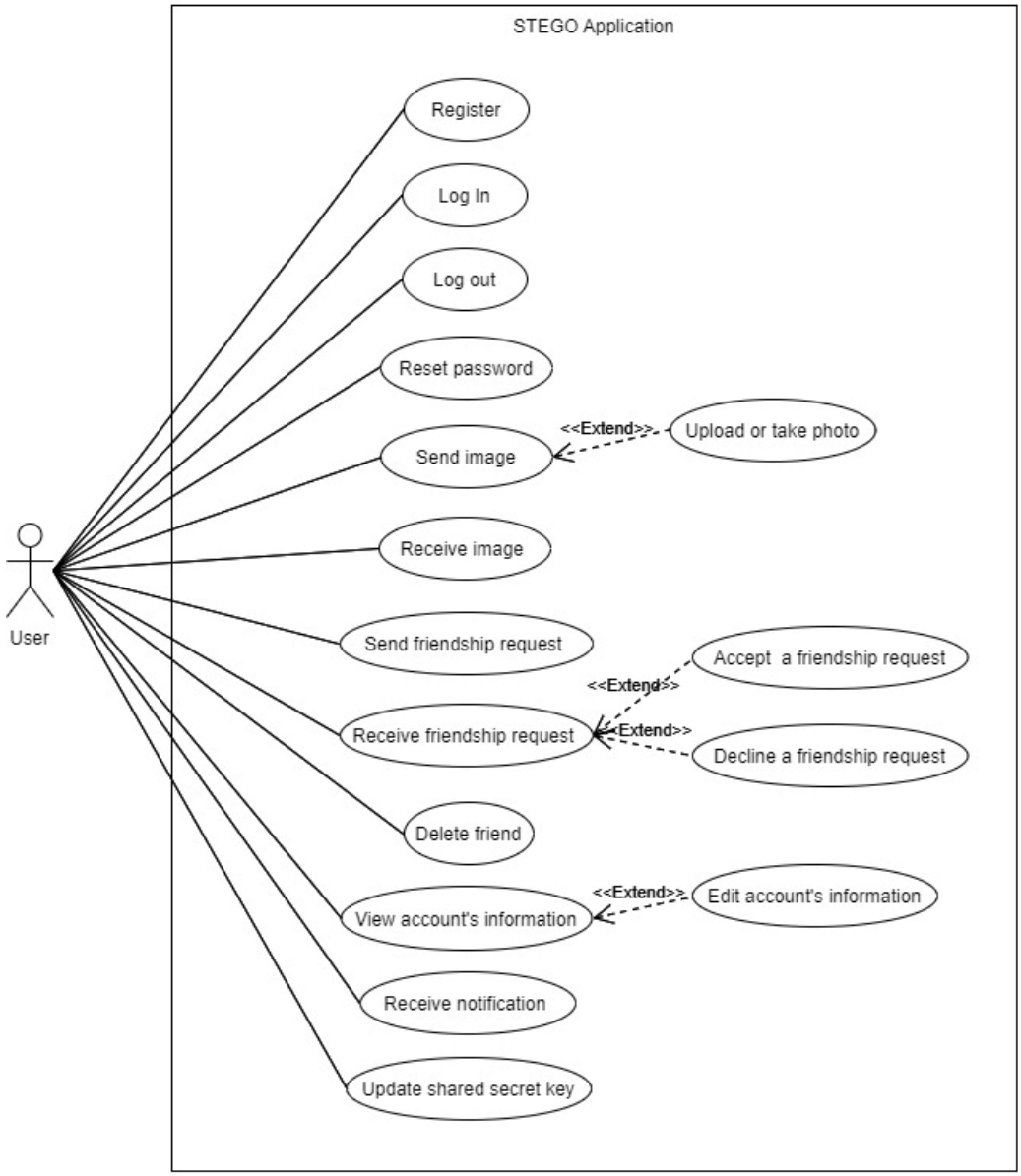


Figure 7: Use Case Diagram

### 4.2.4 Roadmap and Product Backlog

As shown in figure 8, we have created a roadmap for our application that shows the releases we have been through and the sprints of those releases.

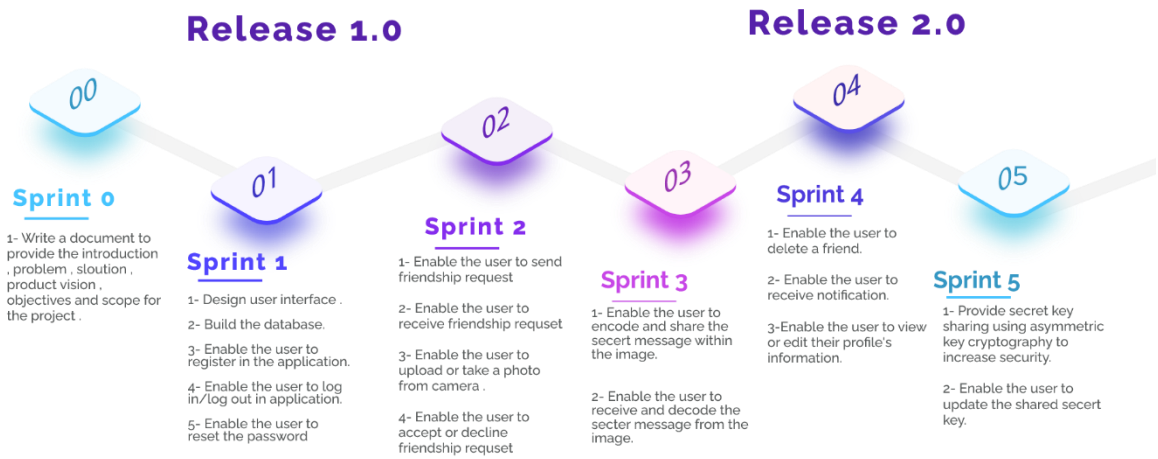


Figure 8: Roadmap

Table 2: Product Backlog

| **ID** | **PBI** (User stories) | **Size** | **Type** (Feature, defect, technical work, knowledge acquisition) | **Status** (To do, in progress, or Done) | **Acceptance Criteria** The conditions of satisfaction that must be met for that item to be accepted. |
|---|---|---|---|---|---|
| 1 | As a user, I want to be able to register my profile so that I can have my own account. | 5 | Feature | Done | - As a user, if I press on the register button, then I should be redirected to the register page.<br><br>- As a user, if I fill in the fields and press on the register button, then the system should validate my information.<br><br>- As a user, If I enter a password consisting of letters, numbers, and symbols less than 8 characters, then an error message should be displayed.<br><br>- As a user, if any information is not validated, then an error message should be displayed.<br><br>- As a user, if my information is validated and accepted, then I should be redirected to verify email page.<br><br>- As a user, if my information is validated and accepted, then an email |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | verification link should be sent to my email<br><br>- As a user, if my email is verified, then I should be redirected to the home page.<br><br>- As a user, if I press on resend email button, then a new email verification link should be sent to my email.<br><br>- As a user, if I press on cancel button, then I should not be registered in the system. |
| 2 | As a user, I want to be able to log in to my account so that I can use the app. | 2 | Feature | Done | - As a user, if I press on the log in button, then I should be redirected to the log in page.<br><br>- As a user, if I entered my log in credentials, then the system should verify it.<br><br>- As a user, if my log in credentials is verified, then I should be redirected to the OTP page.<br><br>- As a user, if my log in credentials is not verified, then an error message should be displayed.<br><br>- As a user, if I'm redirected to the OTP page, then an |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | OTP code should be sent to my phone number.<br><br>- As a user, if I enter the OTP code, then the system should verify it.<br><br>- As a user, if the OTP code is verified, then I should be redirected to the home page.<br><br>- As a user, if the OTP code is not verified, then an error message should be displayed.<br><br>- As a user, if I press on the resend OTP button, then a new OTP code should be sent to my phone number. |
| 3 | As a user, I want to be able to log out of my account so that I can leave the account. | 1 | Feature | Done | - As a user, if I press on the log out button, then I should be logged out of my account and redirected to the login and register page. |
| 4 | As a user, I want to reset my password, so that I can access my account in case I forgot my password. | 1 | Feature | Done | - As a user, if I fill in the field, then the system should validate it.<br>- As a user, if the entered information is validated and accepted, then an email message should be sent to my email.<br>- As a user, if the email message was sent, then a successful message should be displayed. |

| | | | | | |
|---|---|---|---|---|---|
| 5 | As a user I want to send friendship request so that I can add a new friend. | 5 | Feature | Done | - As a user, if I press on the plus button, then I should be redirected to the add friend and requests page.<br><br>- As a user, if I enter a friend's email, then the system should validate it to ensure the field is not empty.<br><br>-As a user, if the textbox is empty, then an error message should be displayed.<br><br>- As a user, if I send a friendship request, then a successful message should be displayed.<br><br>- As a user, if I send a friendship request to a new friend, and the new friend does not have an account, then an email invitation should be sent to the friend's email.<br><br>- As a user, if I send a friendship request to a new friend, and the new friend was already added, then a message should appear specifying that the new friend already exists.<br><br>- As a user, if I send a friendship request to a new friend, and the new friend |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | already sent a friendship request to me, then a message should appear specifying that you already received a friendship request from this user. |
| 6 | As a user I want to receive friendship request so that I can accept or decline the new friend. | 5 | Feature | Done | - As a user, if I receive a friendship request, then a notification message should be displayed.<br><br>- As a user, if I receive a friendship request, then the request should be displayed in the request list.<br><br>- As a user, if I receive a friendship request, then a I can accept or decline the request. |
| 7 | As a user, I want to accept a friendship request, so that I can exchange messages with my friends. | 2 | Feature | Done | - As a user, if I accept the friendship request, then the friend should be added to my friends' list.<br><br>- As a user, if I accept the friendship request, then a successful message should be displayed.<br><br>- As a user, if I accept the friendship request, then the request should be removed from the requests page.<br><br>- As a user, if I accept the friendship request, then the |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | friend who sent the request should receive a notification. |
| 8 | As a user, I want to decline a friendship request, so that I can choose who I can exchange messages with. | 2 | Feature | Done | - As a user, if I decline the friendship request, then the friend should not be added to my friends' list.<br><br>- As a user, if I decline the friendship request, then the request should be removed from the requests page.<br><br>- As a user, if I decline the friendship request, then the friend who sent the request should receive a notification. |
| 9 | As a user, I want to upload/take a photo, so that I can hide a message within it. | 3 | Feature | Done | - As a user, if I press on the photo icon, then a list of two options, camera or photo library, should appear.<br><br>- As a user, if I choose the photo library option, then I should be accessed to my photo library.<br><br>- As a user, if I choose the camera option, then I should be accessed to my phone camera.<br><br>- As a user, if I press on back button, then I should |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | be redirected to the chat page. |
| 10 | As a user, I want to send an image embedded with a secret message to a friend, so that I can share the stego image and ensure the confidentiality of the message. | 5 | Feature | Done | As a user, after choosing the photo, a textbox to write the secret message should appear.<br><br>As a user, if I enter a message in the textbox, then the system should validate it to ensure that it is not empty.<br><br>-As a user, if the textbox is empty, then an error message should be displayed.<br><br>As a user, if I exceed the message limit, which is 100 characters, then an error message should be displayed.<br><br>- As a user, if I press on the send button, then the image should be sent to the friend.<br><br>- As a user, if I send the image, then a successful message should be displayed.<br><br>- As a user, if I send the image, then the image should be displayed in the chat page. |

| | | | | | |
|---|---|---|---|---|---|
| 11 | As a user, I want to receive an image embedded with a secret message so that I can get the secret message within it. | 5 | Feature | Done | -As a user, if I receive an image, then a notification should be displayed.<br><br>- As a user, if I receive an image, then a blue dot should be displayed on the right side of the chat.<br><br>- As a user, if I receive an image, then the chat in the friends list should be at the top of the list.<br><br>-As a user, if I receive an image, then I can see it in the chat page.<br><br>-As a user, if I open the received image, then the show secret message button should be displayed.<br><br>-As a user, if I press on the show secret message button, then the secret message should be displayed.<br><br>- As a user, if the secret message cannot be displayed, them an error message should be displayed |
| 12 | As a user, I want to delete a friend so that I can manage who I share secret messages with. | 1 | Feature | Done | -As a user, if I hold pressing on one of the chats in the friends list, then a list of two options, delete a friend |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | and update secret key, should appear.<br><br>-As a user, if I press on delete a friend, then a confirmation message should appear.<br><br>-As a user, if I press on Ok button, then the friend should be deleted from my friends' list.<br><br>-As a user, if the friend is deleted, then a successful message should be displayed.<br><br>-As a user, if I press on cancel button, then the friend should not be deleted from my friends' list. |
| 13 | As a user, I want to receive a notification so that I can notice when my friends have shared with me an image or accepted/declined my friendship request. | 2 | Feature | Done | -As a user, if I receive an image, then a notification should be displayed.<br><br>-As a user, if the friendship request, I sent got accepted or declined, then a notification should be displayed.<br><br>-As a user, if I receive a notification then I should see a notification dot on the application icon. |

| | | | | | |
|---|---|---|---|---|---|
| 14 | As a user, I want to view my profile's information so that I can manage my account. | 2 | Feature | Done | -As a user, if I press on the account button, then I should be redirected to my profile page.<br><br>-As a user, if I'm redirected to the profile page then all my information should be displayed. |
| 15 | As a user, I want to edit my profile's information so that I can update my information. | 3 | Feature | Done | -As a user, if I press on the edit profile button, then I should be redirected to the edit profile page.<br><br>- As a user, if I press on the edit profile button, then the system should validate my information.<br><br>- As a user, if my information is not validated, then an error message should be displayed.<br><br>- As a user, if my information is validated and accepted, then my information should be edited.<br><br>- As a user, if my information is edited, then a successful message should be displayed.<br><br>- As a user, if I press on the update password button, then I should be redirected |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | to the update password page.<br><br>- As a user, if I fill in the fields in update password page, then the system should validate it.<br><br>- As a user, if the entered information is not validated, then an error message should be displayed.<br><br>- As a user, if the entered information is validated and accepted, then a confirmation message should be displayed.<br><br>-As a user, if I press on Ok button, then the password should be updated.<br><br>-As a user, if I press on cancel button, then the password should not be updated.<br><br>- As a user, if my password is updated, then a successful message should be displayed.<br><br>- As a user, if I press on cancel button in update password page, then I should be redirected to the edit profile page without editing my information. |

| | | | | | |
|---|---|---|---|---|---|
| 16 | As a user, I want to update the secret key shared with my friend, so that it will increase the security of the secret message. | 3 | Feature | Done | -As a user, if I hold pressing on one of the chats in the friends list, then a list of two options, delete a friend and update secret key, should appear.<br><br>-As a user, if I press on update secret key, then a confirmation message should appear.<br><br>-As a user, if I press on Ok button, then the secret key should be updated.<br><br>-As a user, if the secret key is updated, then a successful message should be displayed.<br><br>-As a user, if I press on cancel button, then the secret key should not be updated. |
| **Non-functional Requirements** | | | | | |
| 17 | As a user, I want the application to be available 95% of the time I try to access it, so that I don't get frustrated and find another application to use. **(Availability)** | - | Feature | Done | - |
| 18 | As a user, I want the application's response time to be ranged between 3 to 30 seconds given a good | - | Feature | Done | - |

| | | | | | |
|---|---|---|---|---|---|
| | Internet connection so that I don't become annoyed and try another application. **(Performance)** | | | | |
| 19 | As a user, I want the application to be secured by authenticating the user email, password, and OTP so that unauthorized users cannot access my account. **(Security)** | - | Feature | Done | - |
| 20 | As a user, I want the application to encrypt the password in the database using a SCRYPT hash so that it will not be detected in the event of an attack. **(Security)** | - | Feature | Done | - |
| 21 | As a user, I want to use all the application's features without having any training so that it does not have to take me a lot of time to learn how to use it. **(Usability)** | - | Feature | Done | - |
| 22 | As a user, I want to exchange messages in a secure channel using cryptography so that I can keep my secret messages safe from attackers. **(Security)** | - | Feature | Done | - |

## 4.3 System Design

### 4.3.1 Architectural Diagram

We will make use of System architecture to create a structured solution that satisfies all the technical and operational requirements because our functionality will be made available as an Android application. The architecture that works well for our system is MVC as shown in figure 9.

A model is an architecture component that contains a database's logical operations and stores the data locally on a browser. The view is the element that displays the data to users and contains all the necessary features for user interaction. As it links the model and the view, the controller is the fundamental component of the architecture.

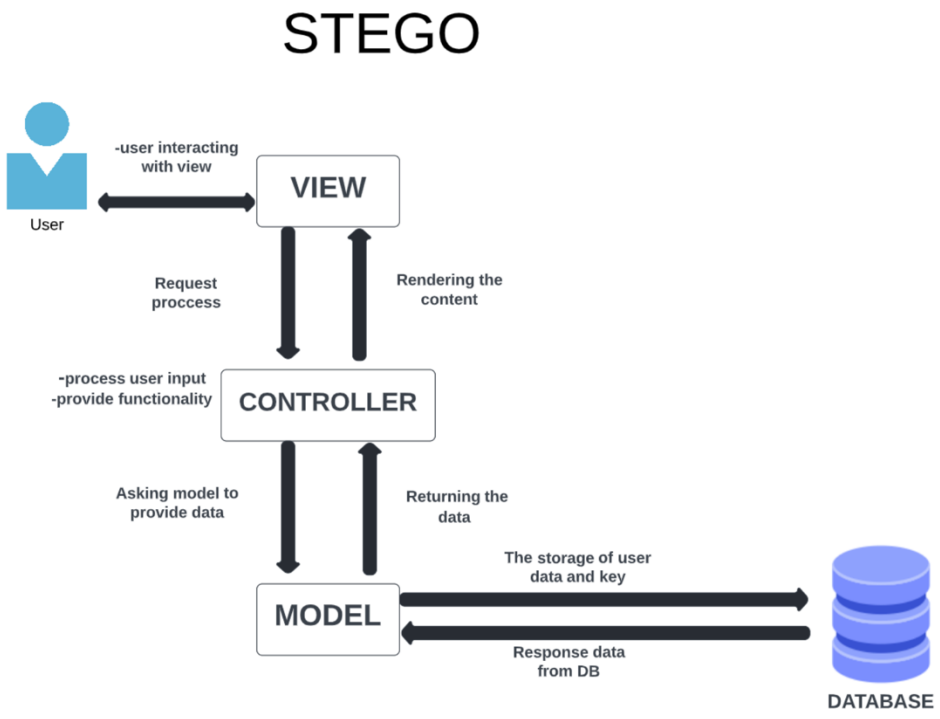


Figure 9: Architectural Diagram

### 4.3.2 Class Diagram /DFD

As shown in figure 10, the main purpose here is to gain a general understanding of how the system is decomposed, and how the individual parts work together to provide the desired functionality.

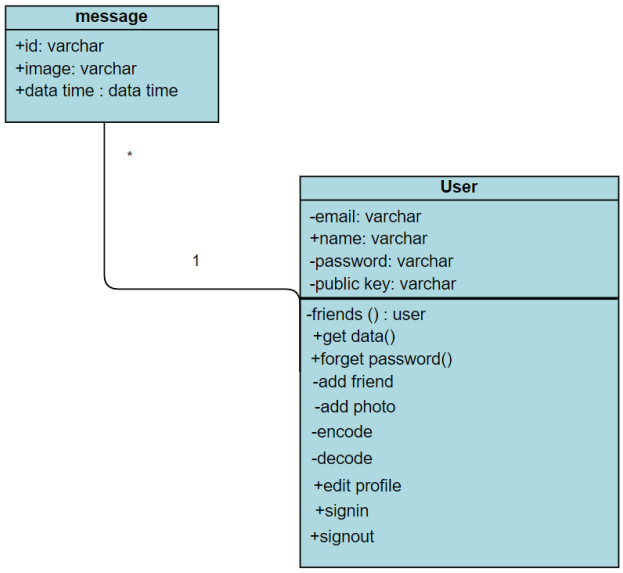


Figure 10: Class Diagram / DFD

### 4.3.3 Component Level Design

In this section we presented our component level design for 4 major user stories using flowcharts.

- Register Flowchart

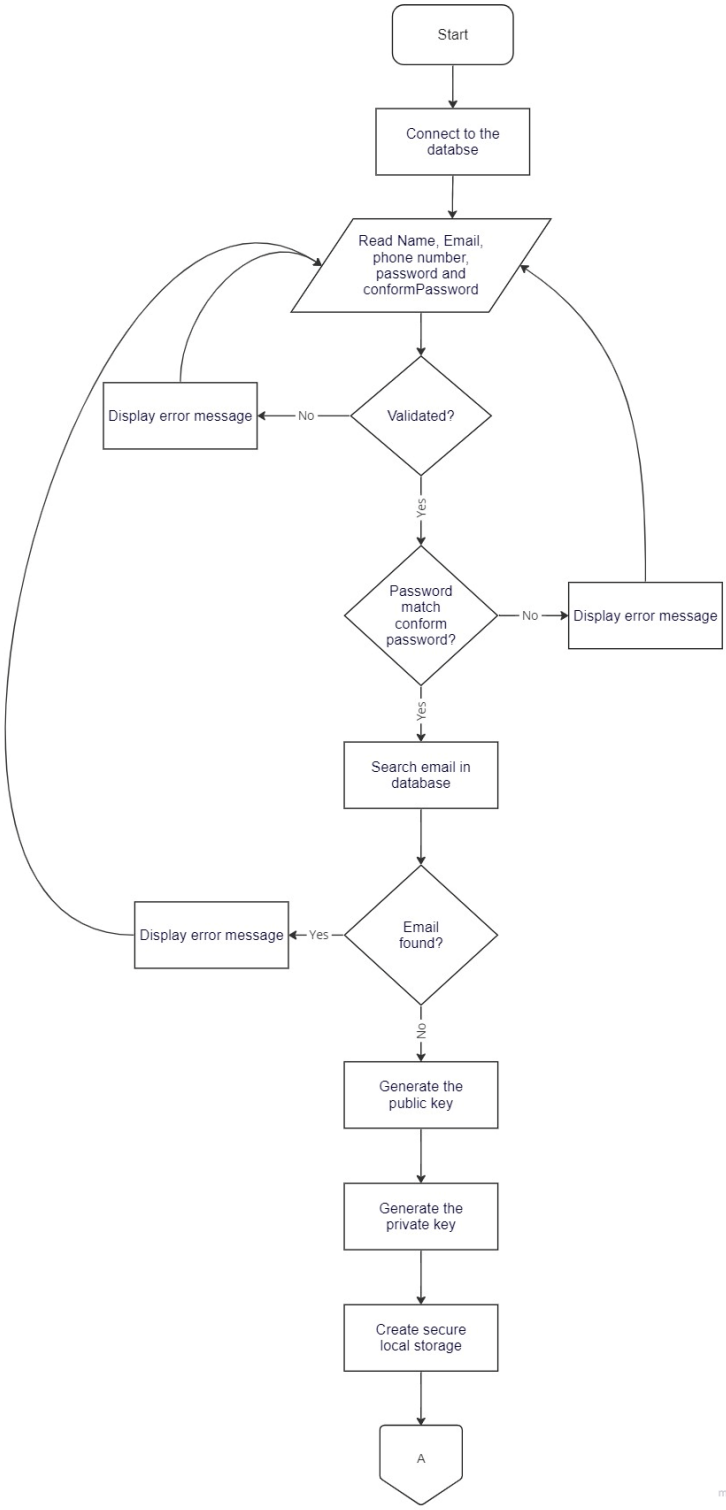


Figure 11: Register Flowchart part 1

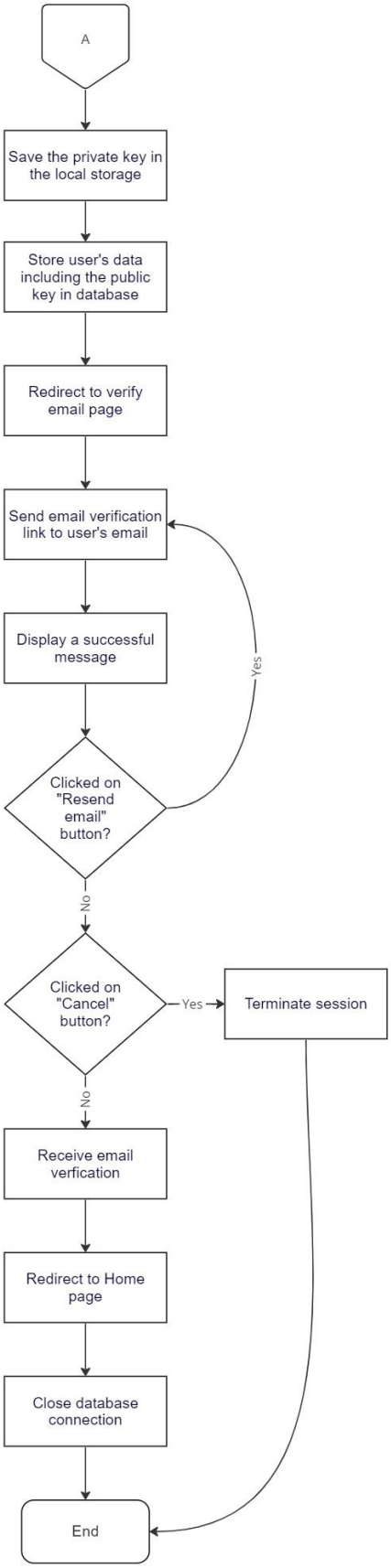


Figure 12: Register Flowchart part 2

- Log in Flowchart

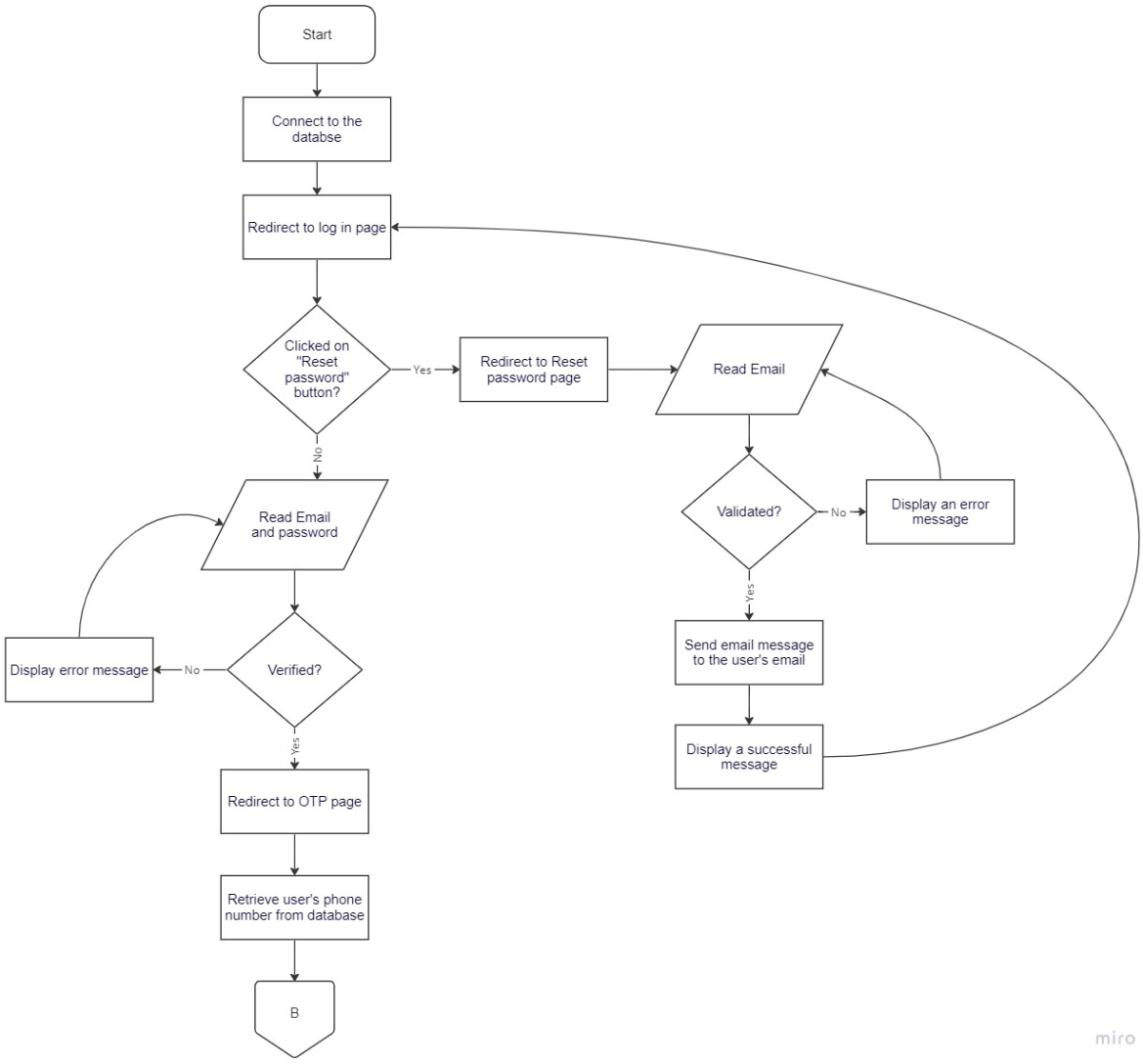


Figure 13: Log in Flowchart part 1

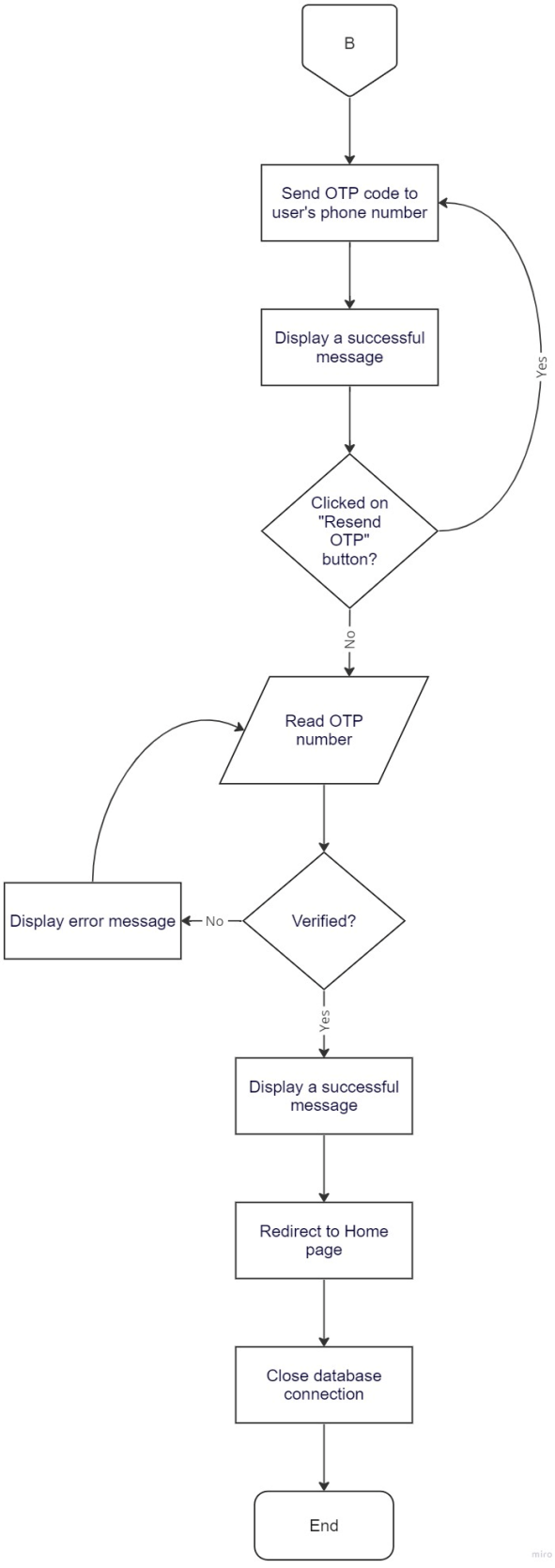


Figure 14: Log in Flowchart part 2

- Send a Friendship Request Flowchart

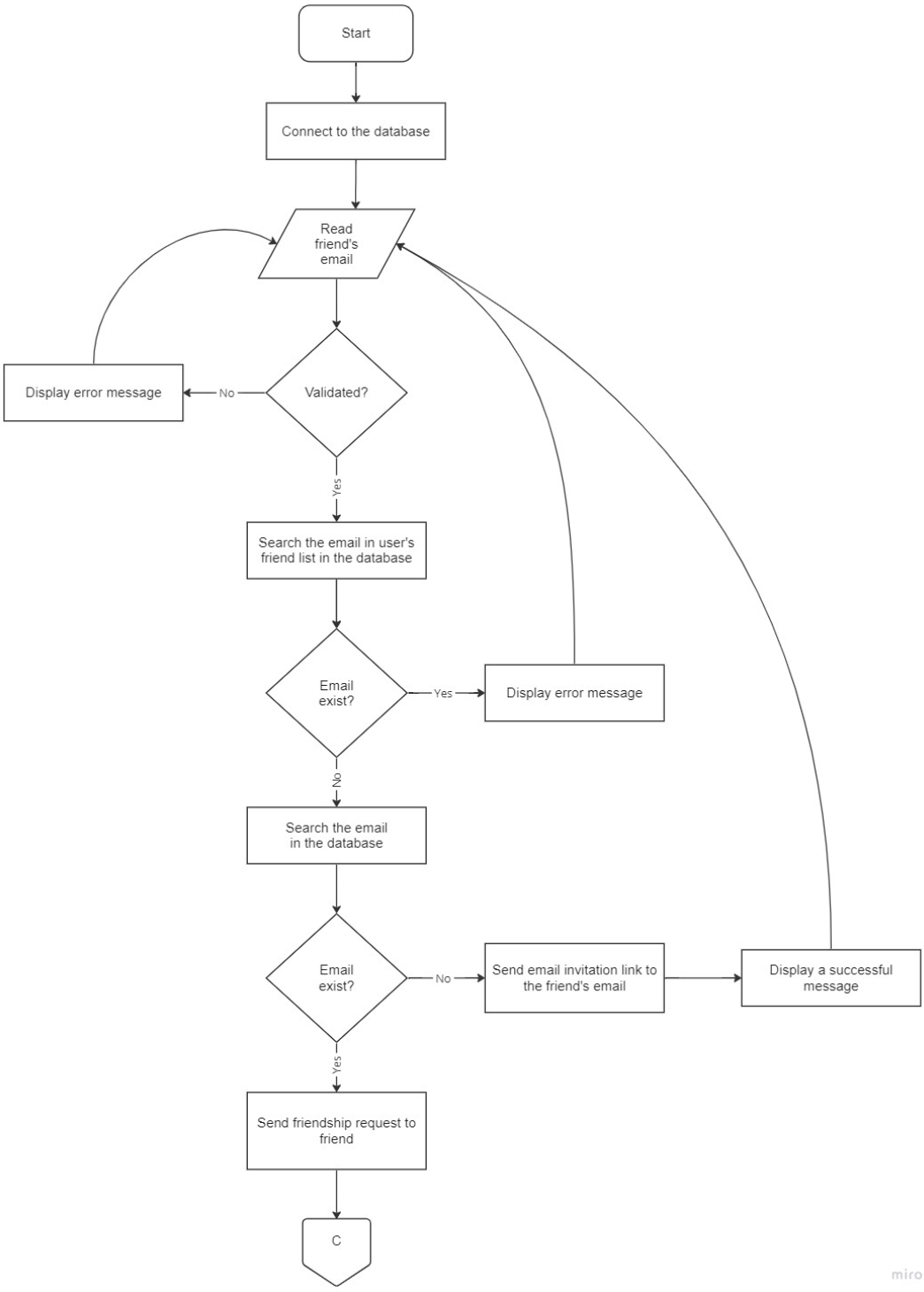


Figure 15: Send a Friendship Request Flowchart part 1

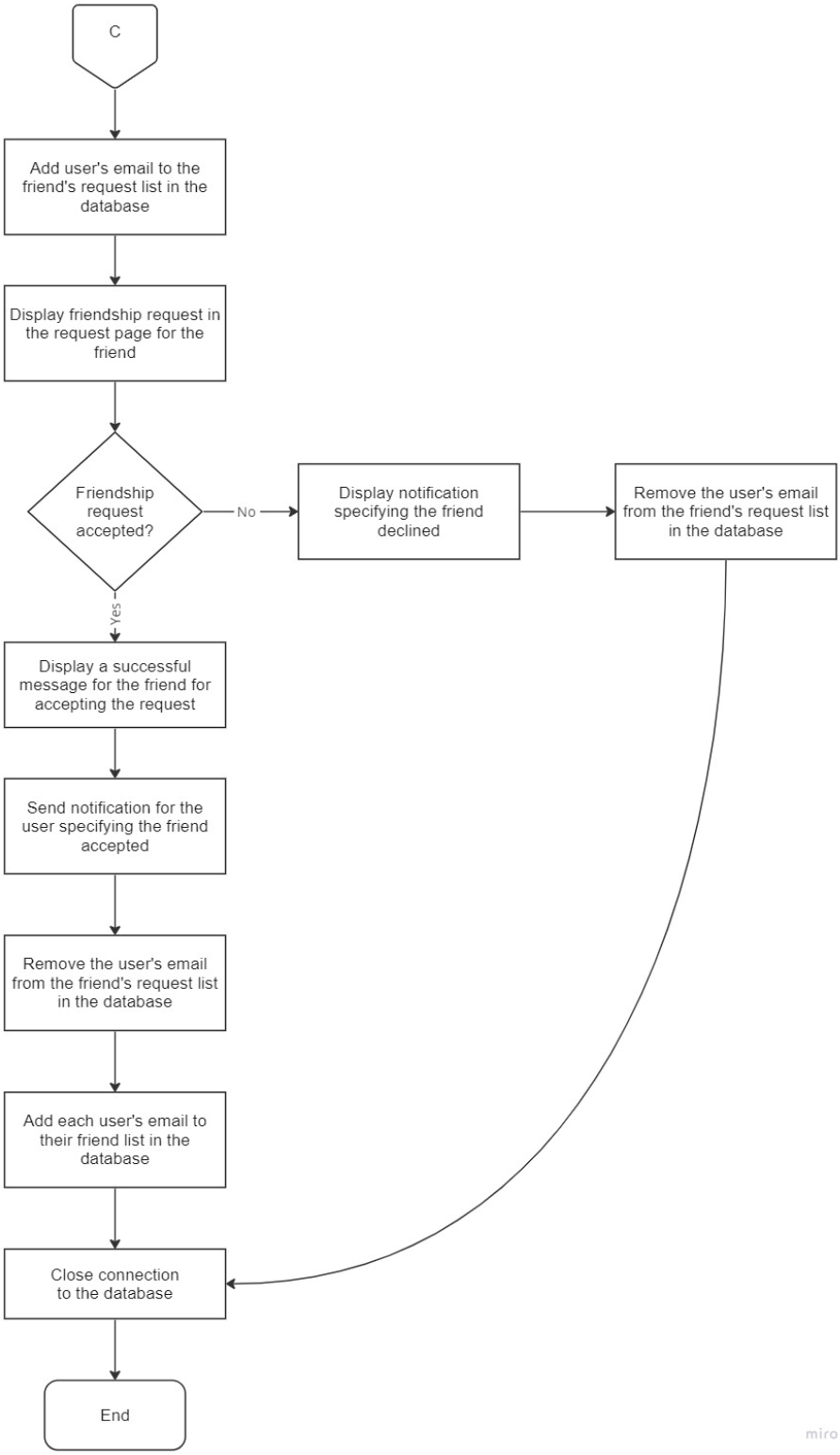


Figure 16: Send a Friendship Request Flowchart part 2

- Receive Image Flowchart

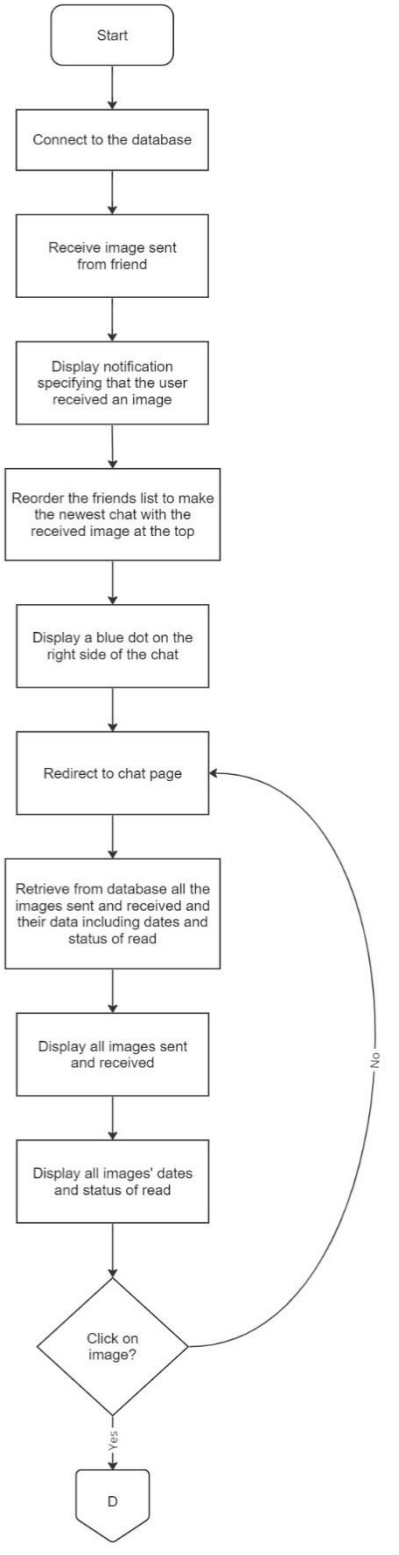


Figure 17: Receive Image Flowchart part 1

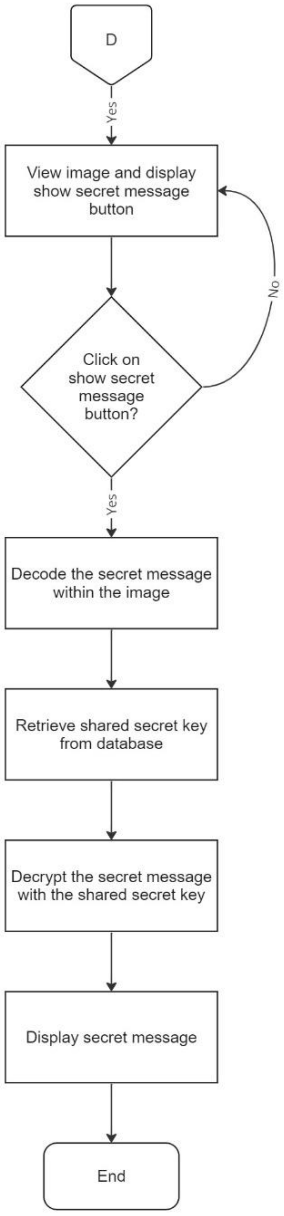


Figure 18: Receive Image Flowchart part 2

## 4.4 Data Design

### 4.4.1 Data Models

Stego application uses a NoSQL database type, we used NoSQL database for its reliability, scalability and efficient language used for communicating with the database and its robust features and tools. It stores data in a very organized way and ensures data integrity. Moreover, because NoSQL is established and backed by a large technical community, our team won't encounter challenges we can't overcome.

We have stored all user data and chat data between users in Firebase. The only thing stored locally on the user's device is the private key.

- The ER Diagram

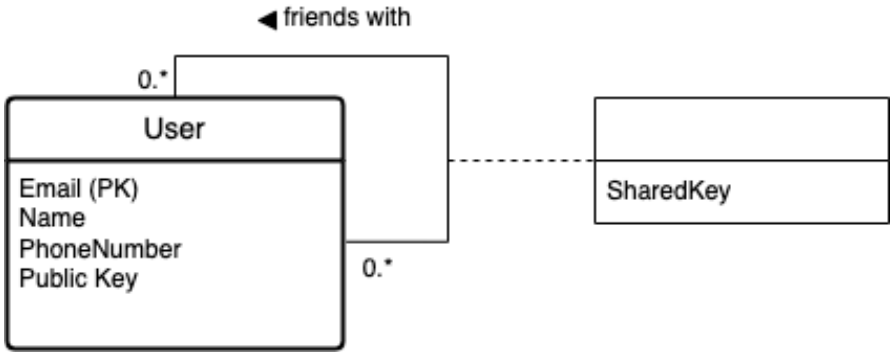


Figure 19: ER Diagram

- The non-relational data model

As shown in figure 20, we represented our firebase hierarchical model using a non-relation data model.

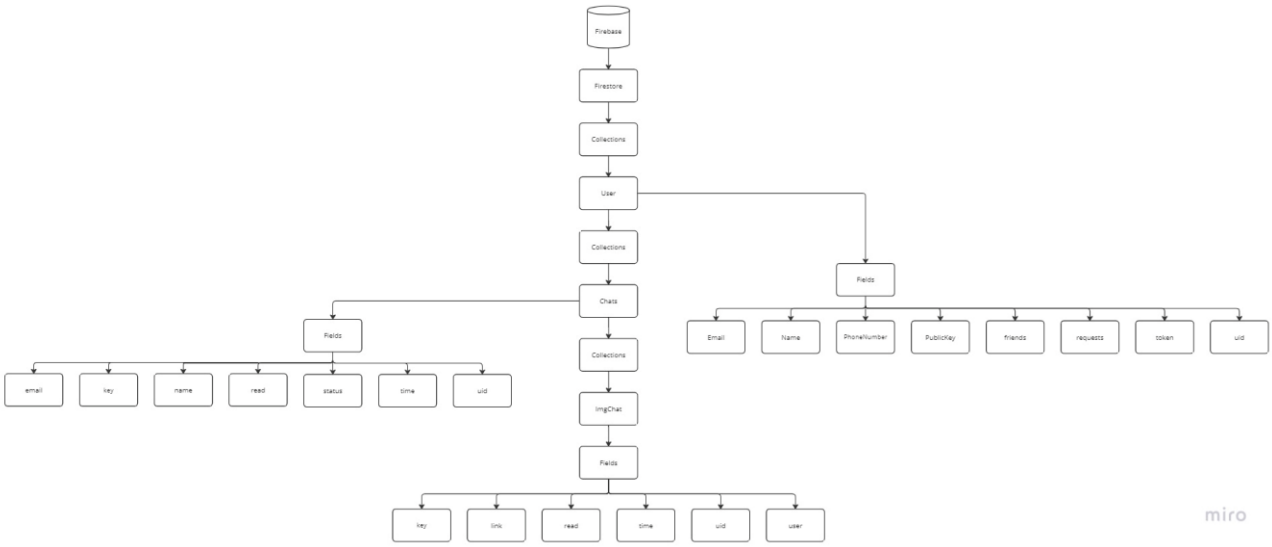


Figure 20: Non-relational Data Model

## 4.5 Interface Design


Figure 21: STEGO Map

- UX guidelines incorporated while designing our interface:
  - Make relevant information easy to find.
  - Simplify your navigation.
  - Make your site accessible to all people.
  - Use familiar fonts.
  - Offer informative feedback.
  - Offer error prevention.
  - Permit easy reversal of actions.

- STEGO Interfaces:

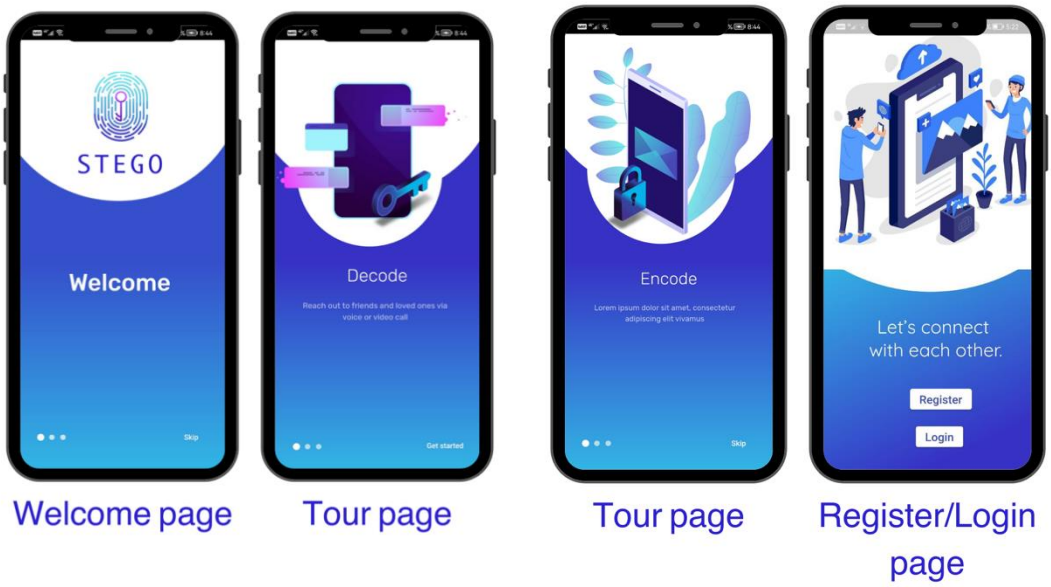


Figure 22: Interfaces part 1

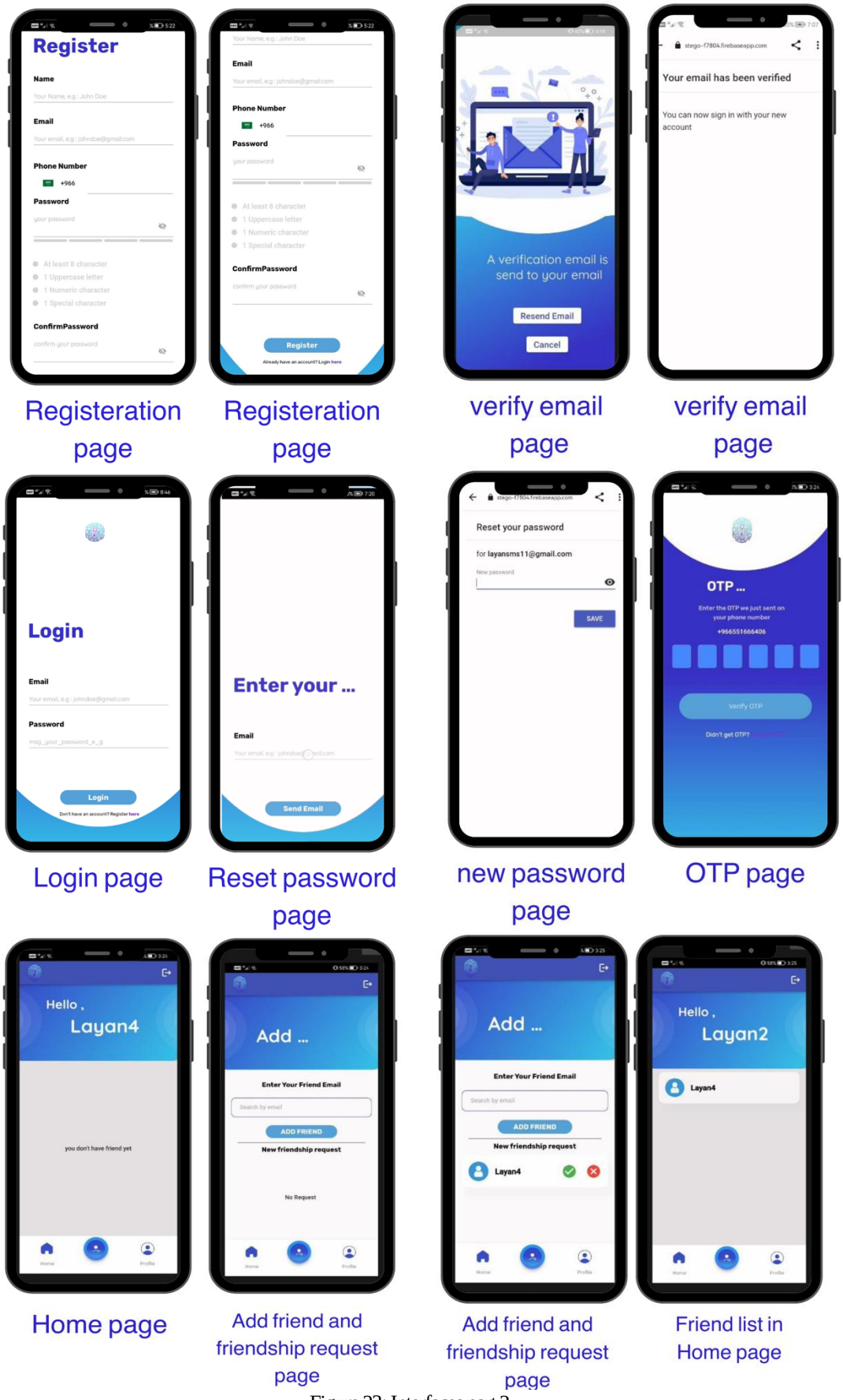


Figure 23: Interfaces part 2

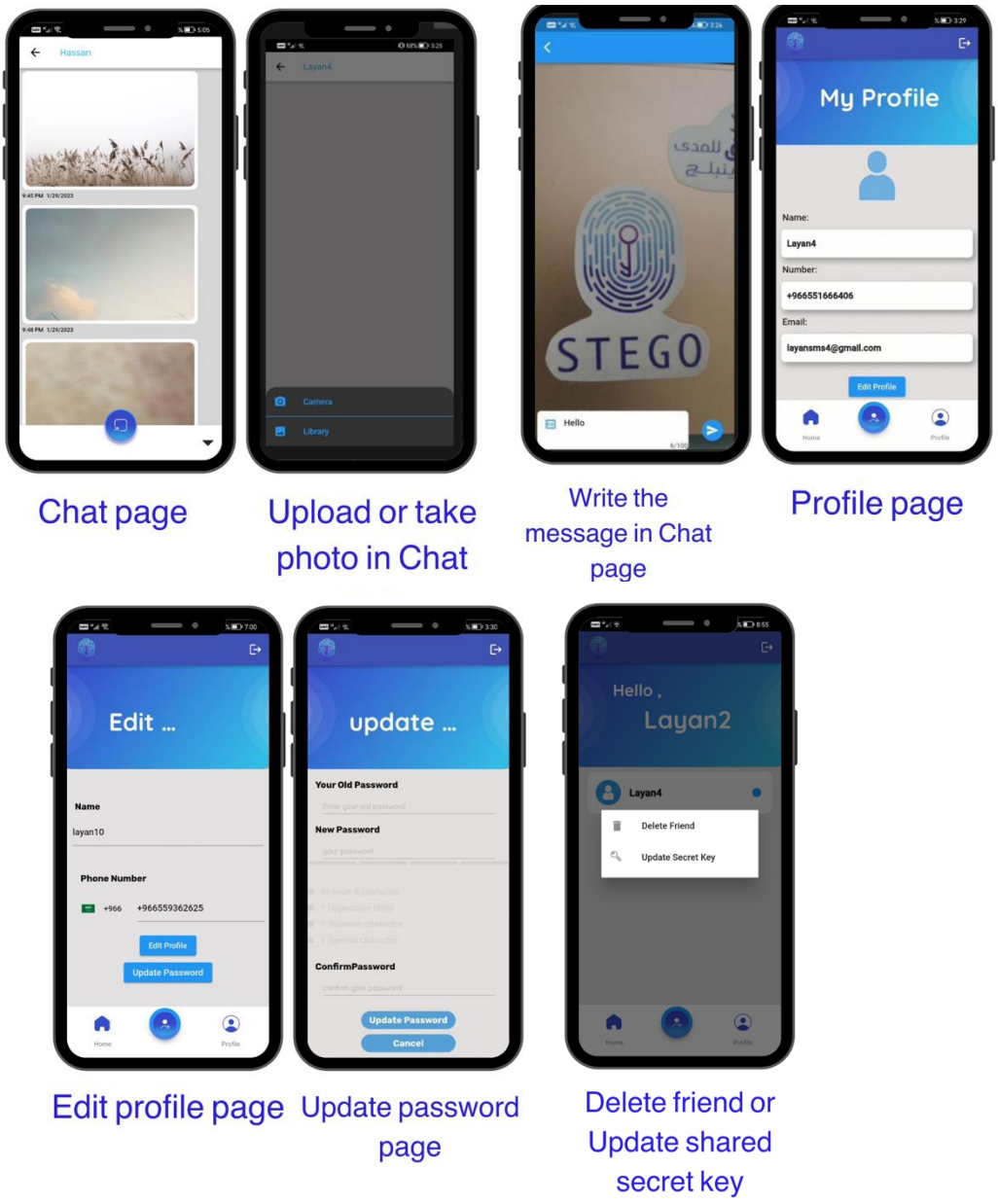


Figure 24: Interfaces part 3

## 4.6 Implementation

This section covers the system implementation in detail. It discusses the software tools used to develop the application, the system implementation, and the challenges.

After selecting the user stories that will be implemented during this release and then breaking those user stories down into tasks. We downloaded the necessary tools and programs to implement the system and then started working on it. We developed the front ends using Android Studio, Flutter and Figma and then created the database for the system using Firebase.

Table 3, table 4 and table 5 show our three major functions' description and flow in STEGO app.

1. Registration Function

Table 3: Regesrtaion Function's Description

| | |
|---|---|
| **Function** | Register a new account |
| **Description** | This function provides the users with the ability to create an account in our application, that provides security by allowing the users to send and receive secret messages that are hidden within an image, by adding friends to exchange with, and so much more features, such as, view and edit profile and receive notification. |
| **Function's Flow** | After the user fill in the fields, the application will validate it to make sure that it abides by the conditions, if it is not validated, then an error message will be displayed, but if it is validated, then a successful message will be displayed and a public and a private keys will be generated, then the user's information will be stored in the database, except for the privet key it will be stored in the local storage for the user's device. After that an email verification link will be sent to the user's email to verify from the email. |

```
      alignment: Alignment.center,
      validator: (val) {
        if (val.isEmpty)
          return 'the field is empty';
        if (val !=
            framePasswordController.text)
          return "Password doesn't match";
        return null;
      },
      value: framePasswordController.text), // CustomTextFormField
  CustomButton(
      onTap: () async {
        if (_formKey.currentState!
                .validate() &&
            phonNumberContorller
                    .text.length >=
                7) {
          showDialog(
              context: context,
              barrierDismissible: false,
              builder: (context) => Center(
                    child:
                        CircularProgressIndicator(),
                  )); // Center

          Email = frameEmailController.text;
          Password =
              framePasswordController.text;
          ConfirmPassword =
              frameConfirmPasswordController
                  .text;
          Name = frameNameController.text;
          phoneNumber =
              '$countryCode${phonNumberContorller.text}';
          print("+++++++++++++++");
          print(phoneNumber);
          final rsaKeypair =
              RSAKeypair.fromRandom();
          publicKey =
              rsaKeypair.publicKey.toString() ;
          privateKey =
              rsaKeypair.privateKey.toString();
          print(
              "The public key is $publicKey");
          print(
              "The private key is $privateKey");
          //Send PK to local storage
          await userSecureStorage
              .setPrivateKey(privateKey);
          print(phoneNumber);
          if (Password == ConfirmPassword) {
            try {
              final newuser = await _auth
                  .createUserWithEmailAndPassword(
                email: Email.trim(),
                password: Password.trim(),
              );
              String? token =
                  await FirebaseMessaging
                      .instance
                      .getToken();
              _firestore
                  .collection("user")
                  .doc(newuser.user?.uid)
                  .set({
                "uid": newuser.user?.uid,
                "Email": Email,
                "Name": Name,
                "token": token,
                "PhoneNumber": phoneNumber,
                "PublicKey": publicKey,
                "friends": [],
                "requests": [],
              });
              newuser.user!
                  .sendEmailVerification();
              Navigator.pushNamed(context,
                  AppRoutes.verfiy);
            } on FirebaseAuthException catch (e) {
              Navigator.pushNamed(context,
                  AppRoutes.registerScreen);
              String msg =
                  e.message.toString();
              ScaffoldMessenger.of(context)
                  .showSnackBar(
```

Figure 25: Registration Function's Implementation

## 2- Encode Image Function

Table 4: Encode Image Function's Description

| | |
|---|---|
| **Function** | Encode a secret message into an image |
| **Description** | This function provides users with the ability to exchange secret messages with friends in a secure way by encrypting the secret message with the shared secret key and then encode the message within an image and then send that image to the intended friend. |
| **Function's Flow** | After the user writes the secret message, the application then will retrieve the secret key shared between the two friends, and then it will retrieve the user's private key to decrypt the secret key, since the secret key is encrypted when it was stored in the database. After that the application will encrypt the secret message using the shared secret key, and then will encode the encrypted secret message inside the image. |

```dart
Future encrypt() async {
  String SKey="";
  String msg = _title.text;
  print("orginal msg"+msg+"the secret key"+widget.Secretkey);

  try {
    EncryptedSecretKey=widget.Secretkey;
    // decrypt shared secret key :
    final myPrivateKeyString = await userSecureStorage.getPrivateKey() ?? "";
    MyPrivateKey=RSAPrivateKey.fromString(myPrivateKeyString);
    print("my private key from secure storage ");
    print(MyPrivateKey.toString());
    print("decrypted Secret Key");
     SKey = MyPrivateKey!.decrypt(EncryptedSecretKey!);

  }
  catch(e) {
    print(e);
  }
  print("original msg"+msg+"the secret key"+SKey);

  // encrypt the message using shared secret key :
  encryptedmsg =
  (await FlutterAesEcbPkcs5
      .encryptString(
      msg,SKey))!;

  print("the original msg");
  print(_title.text);
  print("the encrypted msg");
  print(encryptedmsg);

  setState(() {
    isLoading = true;
  });
  Directory tempDir = await getTemporaryDirectory();
  File? StegoImag = await Steganograph.encode(
    image: File(widget.path),
    message: encryptedmsg,
    outputFilePath: tempDir.path + '/result.png',
  );
  log(StegoImag.toString());
  print(StegoImag!.path);
  log(StegoImag.path);
  if (StegoImag != null) {
    setState(() {
      bytes = StegoImag.readAsBytesSync();
    });
    return StegoImag.path;
  }
}
```

Figure 26: Encode Image Function's Implementation

2. Add New Friend Function

Table 5: Add New Friend Function's Description

| | |
|---|---|
| **Function** | Add new friend to friend list |
| **Description** | This function provides users with the ability to add new friends to exchange secret messages with them and it generates shared secret key with each two users to use it in encrypting the messages before steganography encode process. |
| **Function's Flow** | The user must first accept the friendship request received from their friend in order to add them as friends, after accepting the request, the application will generate a shared secret key between those two friends, and then encrypt it using the public key for each user and then store it in the database. |

```dart
Future<int> addFrind(String email) async {
  String Friendpub ='';
  String Mypub ='';
  String MyEmail=FirebaseAuth.instance.currentUser!.email!;

  await _firestore
      .collection('user')
      .where('Email', isEqualTo: MyEmail)
      .get()
      .then(
          (value) {
            Mypub = value.docs[0].data()["PublicKey"];
      });
  await _firestore
      .collection('user')
      .where('Email', isEqualTo: email)
      .get()
      .then(
          (value) {
        Friendpub = value.docs[0].data()["PublicKey"];
      });
  await getUserData(MyEmail);

  await getUserData(email);

  final myPrivateKeyString = await userSecureStorage.getPrivateKey() ?? "";


  List<dynamic> friends = [];
  List<dynamic> requests = [];
  String name = '';
  String token = '';
  if (await getUserData(email)) {

    await _firestore.collection('user').doc(_currentUser).get().then((value) {
      friends = value.data()!["friends"];
      requests = value.data()!["requests"];
      name = value.data()!["Name"];
      log(value.data()!["friends"].toString());
    });
    if (friends.contains(email)) {
      Utils.toastMessage("Already your friend");
    } else {
      requests.remove(email);
      await _firestore.collection('user').doc(_currentUser).update({
        "friends": FieldValue.arrayUnion([email]),
        "requests": FieldValue.arrayRemove([email]),
      });
      //friend public key from database
      FriendPublicKey= RSAPublicKey.fromString(Friendpub);
      //my public key from database
      MyPublicKey= RSAPublicKey.fromString(Mypub);
      //generate sheared secret key
      var Secretkey = await FlutterAesEcbPkcs5.generateDesKey(128);
      //my encrypted key
      MyEncreptedPublicKey= MyPublicKey!.encrypt(Secretkey!);
      //friend encrypted key
      FriendEncreptedPublicKey= FriendPublicKey!.encrypt(Secretkey!);
```

Figure 27: Add New Friend Function's Implementation part 1

```
      //add friendship data to database for the two users:
      await _firestore
          .collection('user')
          .doc(_currentUser)
          .collection("chats")
          .doc(friendModel.uid)
          .set({
        "status": "true",
        'name': friendModel.name,
        "key": MyEncreptedPublicKey,
        "email": email,
        "read": false,
        "uid": friendModel.uid,
        "time": DateTime.now(),
      });
      await _firestore
          .collection('user')
          .doc(friendModel.uid)
          .collection("chats")
          .doc(_currentUser)
          .set({
        "status": "true",
        'name': name,
        "email": FirebaseAuth.instance.currentUser!.email,
        "key": FriendEncreptedPublicKey,
        "read": false,
        "uid": FirebaseAuth.instance.currentUser!.uid,
        "time": DateTime.now(),
      });
      await _firestore
          .collection('user')
          .doc(await _firestore
              .collection('user')
              .where('Email', isEqualTo: email)
              .get()
              .then((value) {
            return value.docs.first.id;
          }))
          .update({
        "friends": FieldValue.arrayUnion(
            [FirebaseAuth.instance.currentUser!.email!]),
      });
      await _firestore
          .collection('user')
          .doc(await _firestore
              .collection('user')
              .where('Email', isEqualTo: email)
              .get()
              .then((value) {
            return value.docs.first.id;
          }))
          .get()
          .then((value) {
        token = value.data()!["token"];
      });
      //send notification
      LocalNotificationService.sendPushMessage(
          "$name accept your friend request", "Request Accepted", token);
      await _firestore
          .collection('user')
          .doc(_currentUser)
          .get()
          .then((value) {
        log(value.data()!["friends"].toString());
      });
      Utils.toastMessage("Added");
      log("added");
    }
  }
  return 1;
}
```

Figure 28: Add New Friend Function's Implementation part 2

- The Software Tools

Table 6 shows the software tools used to implement STEGO.

Table 6: Software Tools

| Software Type | Version | Description |
|---|---|---|
| Flutter | 3.5.0 | Google's UI toolkit is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase. Flutter code is compiled using Dart's native compilers. |
| Dart | 2.19.0 | Dart is a client-optimized programming language for apps on multiple platforms. It is developed by Google and is used to build mobile, desktop, server, and web applications. |
| Android Studio | 4.0.1 | Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools |
| Figma | - | Prototyping tool used to design the user interface. |
| Firebase | - | Cloud database service used to store data. |
| GitHub | 2.37.0 | A code hosting platform used to manage and integrate source code. |

- Connecting to Firebase

A few steps were required to connect our application to Firebase. First Go to Firebase's website and create an account to create and download the required files and write the required dependencies. Then we downloaded firebase_auth for authentication by running this in command line with flutter "$ flutter pub add firebase_auth". Finally, by clicking continue we finish setting up Firebase.

- EmailJS

EmailJS helped us to send invite emails using client-side technologies to users that don't have an account in our application. We first create an account in EmailJS, after that we connect EmailJS to one of the supported email services (Gmail), and then we create an email template that will be sent to the users. Next we add required dependency (http:), and to send the email we use json.encode() and assign the required parameter (service_id, template_id, user_id, template_params).

- Challenges

The challenge of running the code in Android Studio, each time it takes a long time to run. One of the difficulties we had in application programming was learning the application development language, as well as connecting the cloud firestore to our application. Furthermore, the implementing of two authentication factors (OTP) code and some interfaces code. Moreover, one of the big challenges we have encountered is the procedure of generate asymmetric keys to every user and the process of encode and store the shared secret key for each two users.

Our STEGO application's link to GitHub: https://github.com/LayanSM1/2022-GP1-22


System
Evaluation

# 5 System Evaluation

To make sure that Stego application works as needed, In the is chapter we conducted system testing in this chapter by applying UAT. We provided the results and held a discussion about it.

## 5.1 User Acceptance Testing

During the user acceptance testing stage of software development, the intended audience tests the software. Twenty-two users were chosen for the test. This kind of testing is performed to confirm that the application will satisfy end users' needs and, if necessary, to improve the application.

### 5.1.1 Demographics of Participants

Table 7 shows the demographics of the users that participated in the user acceptance testing of the application.

Table 7: Demographics of the UAT Testers

| Variable | Value | Number of participants = 22 | Percentage |
|---|---|---|---|
| Age | 16-19 | 5 | 22.7% |
| | 20-26 | 7 | 31.8% |
| | 27-39 | 9 | 40.9% |
| | 40 and above | 1 | 4.5% |
| Gender | Female | 11 | 50% |
| | Male | 11 | 50% |
| Educational level | High schools | 5 | 22.7% |
| | University graduate | 7 | 31.8% |
| | University undergraduates | 5 | 22.7% |
| | Upper degrees | 5 | 22.7% |

### 5.1.2 Questionnaire/Interview Results

As we did before in system requirements, we created the survey using Google Forms to survey the users, the survey consists of 13 questions (attached in Appendix C) and was distributed via social media channels, 22 responses were received to gather their feedback and make sure that the acceptance criteria were met.

We asked them how easy was to navigate through the application, was it easy to read and write a secret message, were the error/confirmation messages and the notifications in our application helpful to them, how easy was it to add a friend and edit their profiles. Also, we asked them what they found best about our application and how likely they are to recommend this application to a friend, how would they rate their overall experience with the application, their age, gender, and educational level.

Overall, the questionnaire produced a good result, The navigation within the app and applying the features were easy, the users were very likely to recommend the application to their friends and the experience of it was very good, the age range of people who responded to the survey were male and female and ranged from the 27 to 39 age range, and for the educational most of them were university graduated.

According to our findings, we can better comprehend the many viewpoints' users have expressed through this questionnaire. Moreover, the application satisfies its criteria and it's simple to use and navigate, resulting in a better user experience.

Table 8 shows the responses of the user acceptance testing that was conducted by the targeted audience.

Table 8: Responses of the User Acceptance Testing

| Question Number | Question | Value | Response |
|---|---|---|---|
| 1 | What's your age? | 16-19 | 22.7% |
| | | 20-26 | 31.8% |
| | | 27-39 | 40.9% |
| | | 40 and above | 4.5% |
| 2 | Gender? | Female | 50% |
| | | Male | 50% |

| | | | |
|---|---|---|---|
| 3 | What's your educational level? | High schools | 22.7% |
| | | University graduate | 31.8% |
| | | University undergraduates | 22.7% |
| | | Upper degrees | 22.7% |
| 4 | On a scale from 1 to 5 (1=very difficult, 5=very easy), How easy was it to navigate through the application | Very easy | 36.4% |
| | | Easy | 59.1% |
| | | Neutral | 4.5% |
| | | Difficult | 0% |
| | | Very difficult | 0% |
| 5 | Was it easy to read a secret message that you received? | Yes | 90.9% |
| | | No | 0% |
| | | Maybe | 9.1% |
| 6 | Were the error/confirmation messages in our application helpful to you? | Yes | 90.9% |
| | | No | 0% |
| | | Maybe | 9.1% |
| 7 | On a scale from 1 to 5 (1=very difficult, 5=very easy), How easy was it to add a friend? | Very easy | 27.3% |
| | | Easy | 59.1% |
| | | Neutral | 13.6% |
| | | Difficult | 0% |
| | | Very difficult | 0% |
| 8 | On a scale from 1 to 5 (1=very difficult, 5=very easy), How easy was it to write a secret massage? | Very easy | 59.1% |
| | | Easy | 40.9% |
| | | Neutral | 0% |
| | | Difficult | 0% |
| | | Very difficult | 0% |
| 9 | Were all the notifications received helpful to you? | Yes | 95.5% |
| | | No | 0% |

| | | | |
|---|---|---|---|
| | | Maybe | 4.5% |
| 10 | On a scale from 1 to 5 (1=very difficult, 5=very easy), Was it easy to edit your profile information? | Very easy | 27.3% |
| | | Easy | 45.5% |
| | | Neutral | 27.3% |
| | | Difficult | 0% |
| | | Very difficult | 0% |
| 11 | What do you find best about our application? | Interface design | 81.8% |
| | | Features provided | 68.2% |
| | | User friendly | 36.4% |
| 12 | On a scale from 1 to 5 (1=very bad, 5=very good), How would you rate your overall experience with our application? | Very good | 50% |
| | | Good | 45.5% |
| | | Neutral | 4.5% |
| | | Bad | 0% |
| | | Very bad | 0% |
| 13 | On a scale from 1 to 5 (1= Not at all likely, 5=very likely), How likely are you to recommend our application to a friend? | Very Likely | 59.1% |
| | | Likely | 40.9% |
| | | Neutral | 0% |
| | | Unlikely | 0% |
| | | Not at all likely | 0% |

## 5.2 Quality Attributes (NFR testing)

Table 9 shows the results of the non-functional requirements testing.

Table 9: Quality Attributes

| **User story** | **Quality Attribute** | **Measure** | **Results** |
|---|---|---|---|
| As a user, I want the application to be available 95% of the time I try to access it, so that I don't get frustrated | **Availability:** The degree to which an application is operational, useful, and | Compute that the application is available 95% of the time. | We evaluated the application's overall effectiveness and identified its operational metrics in relation to its |

| | | | |
|---|---|---|---|
| and find another application to use. | functional in order to satisfy the needs of a user or a company is referred to as its availability. | | capacity to provide the desired results. |
| As a user, I want the application's response time to be ranged between 3 to 30 seconds given a good Internet connection so that I don't become annoyed and try another application. | **Performance:** Application functionality and user responsiveness are reflected in an app's performance. | Compute the response time to be ranged between 3 to 30 seconds. | 90% of users reported being satisfied with the application's overall performance, according to our testing. |
| As a user, I want to use all the application's features without having any training so that it does not have to take me a lot of time to learn how to use it. | **Usability:** How well a user can utilize an app to accomplish a goal is known as usability. | All program functionalities must be used by users without any training. | Users generally agreed that using the app and its features was simple and easy. |

## 5.3 Discussion

This section discusses the results of user feedback and surveys and how the application will improve after collecting user feedback from testing our application. According to the users, the application will solve the problem of security for the message and protect its confidentiality. We found that navigating through the app and the features provided by our application were easy to do. All the users found error/confirmation messages to be quite useful, as well as the notifications. Our application was highly rated by users and almost every one of them would recommend it to a friend based on their experience.

# Conclusions and Future Work

# 6 Conclusions and Future Work

- Global impact

Many people send and receive daily messages in this technological age, but what about secret messages? It is challenging to share these messages, whether they come from two different countries or from the same country and they might contain a variety of confidential data of different kinds, whether they are for work or may be personal to the user in many aspects of their lives. Many of the challenges and issues users run into when sending this kind of secret message anytime, anywhere, will be resolved by an application like STEGO.

- Local impact

Stego will help the Kingdom of Saudi Arabia in digital transformation to ensure the security of the message between the sender and receiver to prevent third parties from accessing the content of the message.

- Problems and challenges encountered during software development.

The Flutter framework and the dart programming language were new to the team members at the beginning of the implementation, and we ran across a few exceptions that were difficult to figure out how to fix. Additionally, it was difficult to find resources that clearly explained the flutter framework and dart language. Furthermore, we had difficulty figuring libraries related to the steganography science since some of them were outdated, so we couldn't add those packages, and we were forced to replace some of the libraries we used at the start of the program in order to move forward with the work. We also struggled with time management as we learned a new framework and studying for other university courses.

- The main contribution of the project

The main contribution of STEGO application is to facilitate communication between users in sending and receiving secret messages by sharing a secret key using asymmetric encryption, as the private key is stored locally on the user’s device, which increases a high degree of protection for the sent secret messages, in addition to the fact that the user's information is stored in a highly confidential manner.

- Limitations and future work:

STEGO is an English mobile application designed mainly to help users to send secret messages, by upload/take photo and encrypt the secret message by using symmetric encryption and then encode it within the image and for more security STEGO shares the secret key by using asymmetric encryption. STEGO users should use public, private and secret keys to use our system. STEGO users can view and edit their profile information, add/delete a friend, and receive notifications. STEGO at this point does not support uploading/taking audio or videos, although this functionality could be added in the future. Since STEGO is an android application. So as a future work, STEGO will be generalized for IOS devices. Also, STEGO will provide more languages. From the user perspective, STEGO will enable the user to adjust periodically and customize how to receive notification (once per day, twice per day).

- Conclusion

This document depicts our experience working with STEGO, from the idea's inception as a seed into its development through every stage we underwent. Beginning with the introduction chapter, which clarifies the idea and gives a general overview of STEGO. The introduction chapter is followed by the background chapter, which is important in preparing the reader to understand STEGO details by providing a brief explanation of knowledge aspects in which STEGO falls. To deliver an application that fills a gap in applications, and to specify STEGO features, we examined and discussed academic papers and mobile applications within the same field as STEGO, which were represented in the literature review chapter. After achieving a clear understanding of STEGO features, we began the system design and development chapter, which converts STEGO features into a form used to facilitate the implementation of STEGO application and support the understanding of STEGO components. After that, we began developing STEGO by using flutter framework and lastly in system evaluation chapter we tested the application to ensure that all the futures were implanted correctly and it’s bug-free.

# Acknowledgement

# 7 Acknowledgement

We thank Allah for providing us with the abilities and strength needed to successfully complete the project. Without the assistance of those who guided and encouraged us along the way, it couldn't have been finished. Our project reflects our supervisor Dr. Kholoud Al-saleh, support and valuable advice, and we are grateful for it. We would also like to thank Dr. Abeer Aldrees and Dr. Alia Alabdulkarim for cooperating with us. Finally, we should admit that we couldn't have finished this without the endless help of our family and friends.


References

# 8 References

[1] A. Choudary, “Steganography Tutorial – A Complete Guide For Beginners.” Edureka. https://www.edureka.co/blog/steganography-tutorial (accessed Sep. 05, 2022).

[2] M. Semilof and C. Clark, “What is steganography?” Techtarget. https://www.techtarget.com/searchsecurity/definition/steganography (accessed Sep. 05, 2022).

[3] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, “Digital image steganography: Survey and analysis of current methods,” Signal Processing, vol. 90, no. 3, pp. 727–752, 2010, doi: 10.1016/j.sigpro.2009.08.010.

[4] N. F. Johnson and S. Jajodia, “Exploring steganography: Seeing the unseen,” Computer (Long. Beach. Calif)., vol. 31, no. 2, pp. 26–34, 1998, doi: 10.1109/MC.1998.4655281

[5] P. Kumar and V. K. Sharma, “Information Security Based on Steganography & Cryptography Techniques: A Review,” Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 4, no. 10, pp. 246–250, 2014.

[6] T. Morkel, M. S. Olivier, and J. H. Eloff, “an Overview of Image Steganography,” Africa (Lond)., vol. 83, no. July, pp. 51–107, 2005, [Online]. Available: http://martinolivier.com/open/stegoverview.pdf.

[7] S. Iqbal, S. Singh, and A. Jaiswal, “Symmetric Key Cryptography: Technological Developments in the Field,” Int. J. Comput. Appl., vol. 117, no. 15, pp. 23–26, 2015, doi: 10.5120/20631-3248.

[8] N. Subramanian, O. Elharrouss, S. Al-Maadeed, and A. Bouridane, “Image Steganography: A Review of the Recent Advances,” IEEE Access, vol. 9, pp. 23409–23423, 2021, doi: 10.1109/ACCESS.2021.3053998.

[9] J. T. Harmening, Virtual Private Networks. Elsevier Inc., 2017.

[10] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, “A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish,” Procedia Comput. Sci., vol. 78, no. December 2015, pp. 617–624, 2016, doi: 10.1016/j.procs.2016.02.108.

[11] E. Conard, S. Misenar, and J. Feldman, “CISSP Study Guide,” Second Edi. 2012.

[12] E. Conard, S. Misenar, and J. Feldman, “Eleventh Hour CISSP: Study Group,” Third Edit. 2017

[13] S. M. Wadi and N. Zainal, “Rapid Encryption Method based on AES Algorithm for Grey Scale HD Image Encryption,” Procedia Technol., vol. 11, no. Iceei, pp. 51–56, 2013, doi: 10.1016/j.protcy.2013.12.161.

[14] N. Aleisa, “A comparison of the 3DES and AES encryption standards,” Int. J. Secur. its Appl., vol. 9, no. 7, pp. 241–246, 2015, doi: 10.14257/ijsia.2015.9.7.21.

[15] A. Muhammad Abdullah, “Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data View project Call for papers View project Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt,” no. June, 2017, [Online]. Available: https://www.researchgate.net/publication/317615794.

[16] “RSA Algorithm in Cryptography.” Geeksforgeeks. https://www.geeksforgeeks.org/rsa-algorithm-cryptography/ (accessed Dec. 11, 2022).

[17] M. Cobb. “RSA algorithm (Rivest-Shamir-Adleman)” Techtarget. https://www.techtarget.com/searchsecurity/definition/RSA#:~:text=The%20RSA%20algorithm%20(Rivest%2DShamir%2DAdleman)%20is%20the,an%20insecure%20network%20such%20as (accessed Dec. 11, 2022).

# Appendix A: Requirements Elicitation's Interviews

# 9 Appendix A: Requirements Elicitation's Interviews

- Interviews' Questions
  - What is your understanding of steganography?
  - Do you think steganography and cryptography work better with each other?
  - How important is the field of steganography?
  - How can you protect sensitive information from being accessed by unauthorized parties?
  - What criteria do you keep in mind when deciding whether to use the app, especially in terms of security?

- Interviews' Analysis

The founder of the cyber security firm Shield IT, Mr. Hmoud Alhalmani, was the first person we spoke with during our interview. The interview, which comprises of 5 questions (appended as Appendix A), was done through Zoom on September 25, 2022. We started by asking him What is his understanding of steganography, and he answered that he does not have a deep vision of the steganography, but he has knowledge of the ideas of the Blockchain and knowledge in Web3 technologies that how the data is preserved so that no one can view it except the responsible persons. When we asked if he thought steganography and cryptography worked better together, he said that, of course, they did. However, he added that there will be a big challenge in mastering both technologies at once. He said, the two technologies are now in high demand in the job market. We asked him how important the field of steganography is, and he responded that it is extremely essential, but it targets a specific category of users who want to convey messages and sensitive information to the other party, and if it is used effectively and competitively for comparable applications, I believe it will be utilized by parties that want to share their data within the party or with other parties. We asked him how to safeguard sensitive information from being accessed by unauthorized persons, and he replied that encryption can be used, as in the case of e-mail, where the responsible person sends mail to a specified person, and the message is not opened unless a key is exchanged between them. Additionally, you can design a unique algorithm for your application to ensure the security and confidentiality of the data so that hackers cannot understand this algorithm or access the data

being transferred. Finally, we asked Mr. Hmoud what criteria he examines when determining whether to use the app, particularly in terms of security, and he said that the most essential factors that must be considered are privacy and security, as well as a strong and secure encryption system, so that I can use the program while ensuring that no unauthorized person has access to my data.

The second interview was with Dr. Alia Alabdulkarim, an assistant professor in the department of information technology at King Saud University's College of Computer Information and Sciences. On September 27, 2022, the interview was conducted through email. She answered in response to the first question that Steganography means hiding the existence of a message (or malware) using a host file. Steganography has recently used tweets. She responded that it depends on the sensitivity of the message being sent when asked her view on how well steganography and cryptography function together. Because cryptography can create additional processing burden. In response to the third question, which asked how important the field of data steganography is, she stated that it is very important. When we questioned her about how to prevent unauthorized parties from accessing sensitive information, she replied that it depended on the type of information and where it is kept: Databases are typically secured in organizations by policies. Acls can be used to safeguard files. Hashing is a method for protecting passwords. Cryptography can be used to protect files in extremely sensitive situations, but it comes with the additional responsibility of safeguarding the keys. Finally, when asked what considerations she makes regarding the app's security, specifically when selecting whether to use it or not, she said I would be checking for secure transmission, secure key generation and storage, and most importantly, secure, and strong algorithm.

Mr. Fahad Alduraibi was the third interviewee. Holds master’s degree in electrical and computer engineering from Southern Illinois University Carbondale (SIUC), with interests in information security, IoT, and embedded systems. The interview took place on September 29, 2022, through Twitter. When we questioned him about what steganography is, he said that it is the process of hiding a message inside another message, photo, music, etc. Then we asked him if he thought steganography and cryptography worked better together, and he responded that he thought they did because there are ways to detect steganography and encryption can be a protection to further conceal the information. And in response to the third question, about how important the field of steganography is, he stated that it is important for specific practices but not for all. We questioned him about how he can guard against unauthorized individuals

getting access to sensitive information. He stated that the simplest and most usual method is to use cryptography. Finally, we questioned him about the considerations he makes when choosing whether to use an app or not, particularly regarding security. How simple it is to detect if something is hidden inside an object, he said. And how strong is the encryption used.

Finally, Dr. Arwa Alsultan was the last interviewee for an assistant professor in the department of information technology at King Saud University's College of Computer Information and Sciences. On September 29, 2022, the interview was conducted through email. She responded to the first question by stating that steganography is a method for concealing a message in a photo. When asked if she thought steganography and cryptography worked well together, she said yes. The third question addressed how significant the field of data steganography is, and she responded that it is important, yet overshadowed by encryption. When we asked her how she prevents unauthorized parties from accessing sensitive information, she said she used encryption & strict access control policies. Finally, when asked what criteria she considers when deciding whether to use an application in terms of security, she mentioned the manufacturer's reputation, online reviews.

# Appendix B: Requirements Elicitation's Questionnaires

# 10 Appendix B: Requirements Elicitation's Questionnaires


Figure 29: Elicitation's Questionnaires

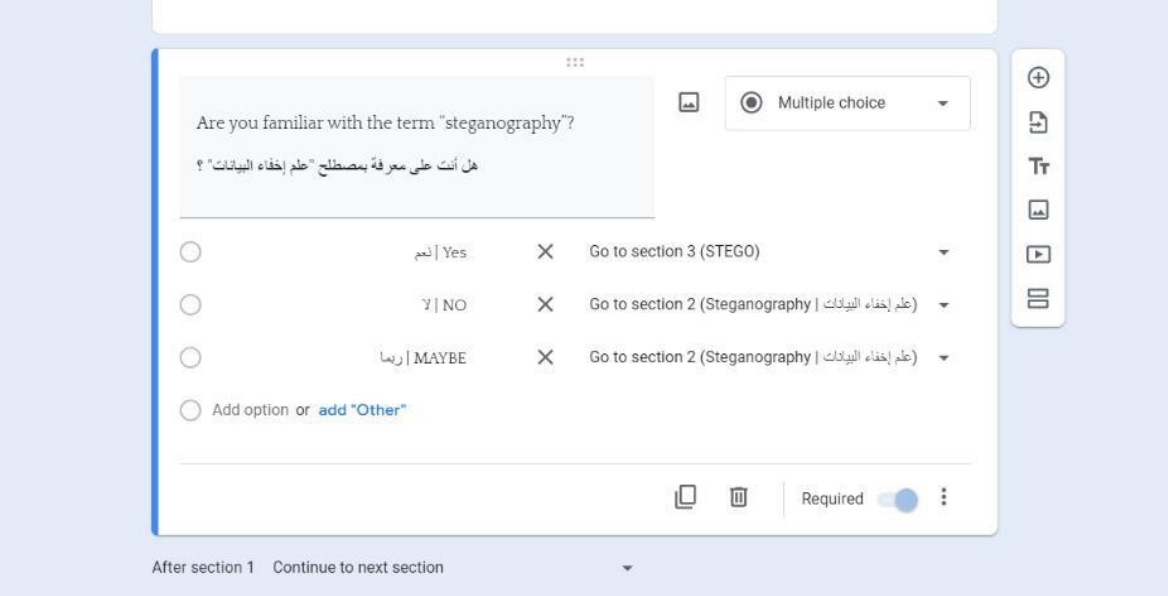


Figure 30: Elicitation's Questionnaires

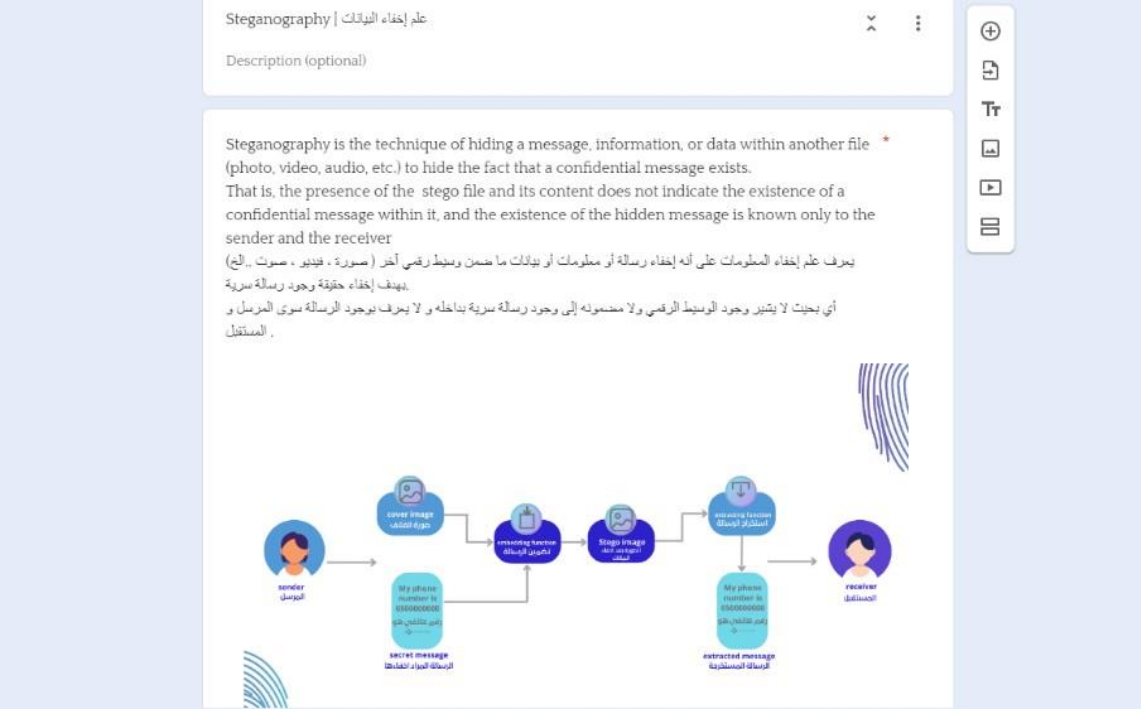


Figure 31: Elicitation's Questionnaires


Figure 32: Elicitation's Questionnaires


Figure 33: Elicitation's Questionnaires


Figure 34: Elicitation's Questionnaires

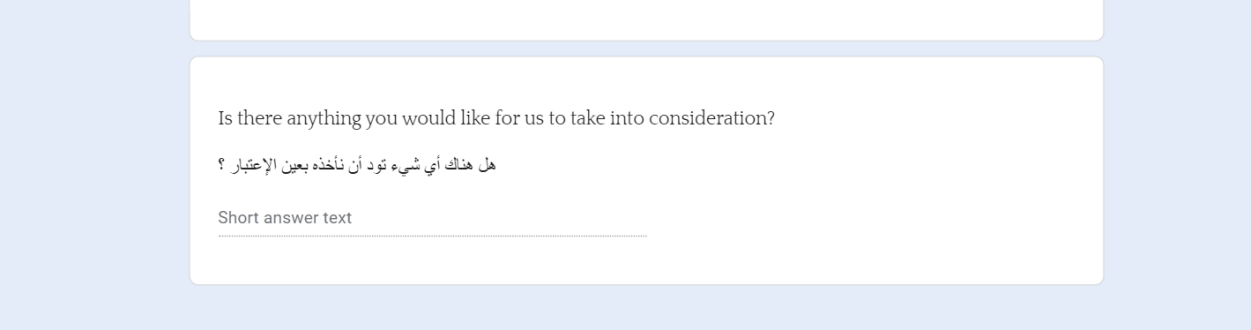


Figure 35: Elicitation's Questionnaires

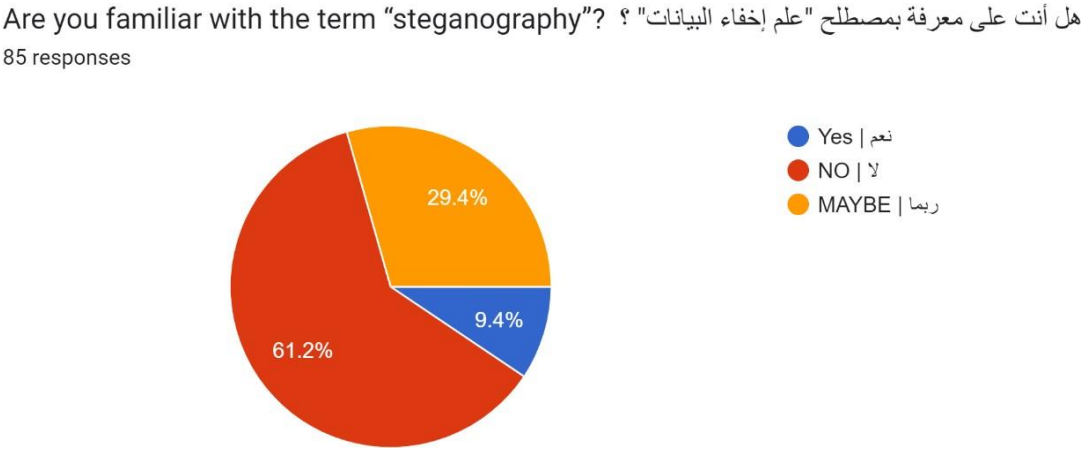


Figure 36: Elicitation's Questionnaire's Response

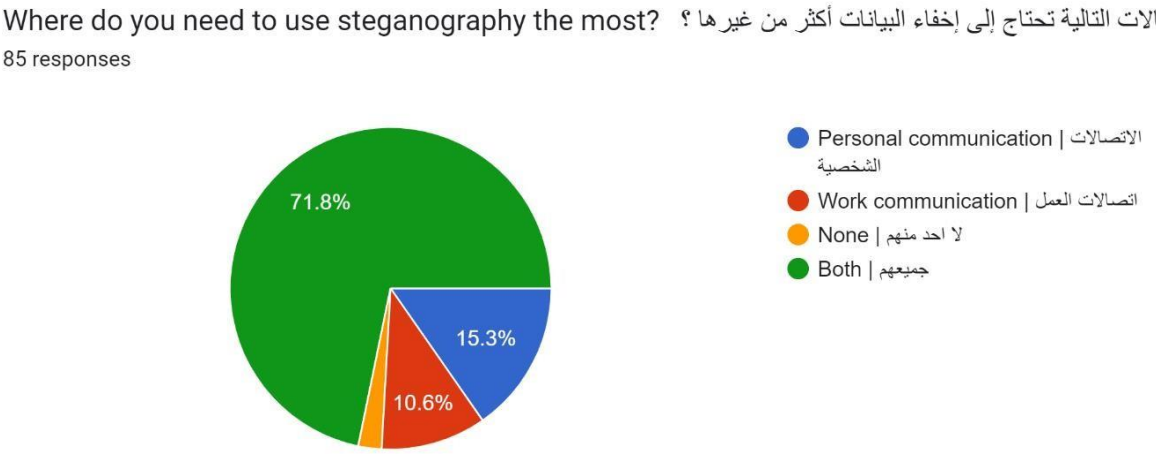


Figure 37: Elicitation's Questionnaire's Response

Do you face difficulty when hiding your secret messages or information? هل تواجه صعوبة عند إخفاء رسائلك أو معلوماتك السرية ؟

85 responses

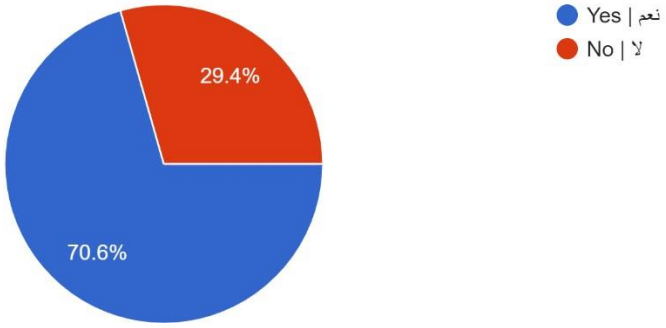


Figure 38: Elicitation's Questionnaire's Response

How often do you need to hide your secret messages /information? كم مرة تحتاج لإخفاء رسائلك/معلوماتك السرية ؟

85 responses


Figure 39: Elicitation's Questionnaire's Response

Do you believe that making a steganography application will make the communication and exchange of secret messages better ? هل تعتقد أن إنشاء ... المعلومات سيجعل الإتصال وتبادل الرسائل السرية بشكل أفضل ؟

85 responses

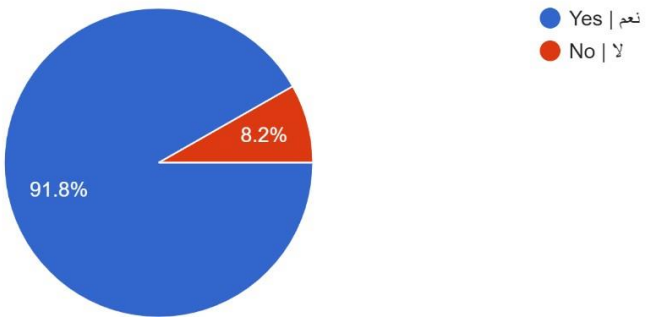


Figure 40: Elicitation's Questionnaire's Response

Would you use our application to send your messages that contain confidential information? هل سوف تستخدم تطبيقنا لإرسال الرسائل التي تحتوي على معلومات سرية ؟

85 responses

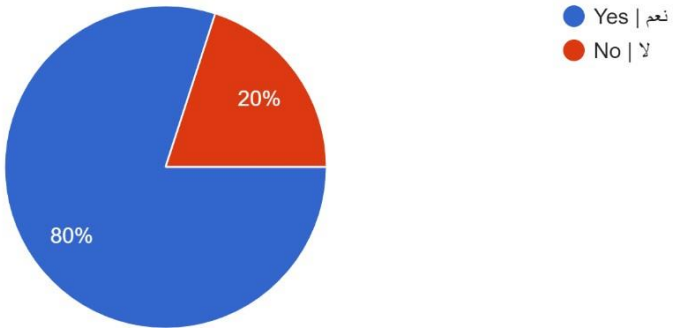


Figure 41: Elicitation's Questionnaire's Response

| Is there anything you would like for us to take into consideration? | 1 |
|---|---|
| لا | 12 |
| لا، الله يوفقكم | 1 |
| البيانات تكون مخفية بشكل كامل وتكون متاحه فقط للشخص المسؤول عنها | 1 |
| سووه بسرعه | 1 |
| اعطاء صلاحية، بحيث ان المرسل يمكنه اتاحة صلاحية اظهار الرسائل لاكثر من مستقبل | 1 |
| بالتوفيق الله ييسر لكم | 1 |
| الفكرة حلوة لكن كيف راح يثقون الناس في أنكم ما راح تشاركون المعلومات السرية تبعهم؟ | 1 |
| اخفاء هوية منشئ الملف المرئي | 1 |
| بالتوفيق | 1 |
| Apps in general is an invitation to someone's private information no matter how "trusted " it is | 1 |
| لا. | 1 |
| الله يوفقكم وينفع فيكم الإسلام و المسلمين | 1 |
| No thing more | 1 |
| ان لا يتمكن مسؤولين التطبيق من الوصول الى الرسائل الخاصة | 1 |
| لا شكرا | 1 |
| التطبيق امن بسبب كثره المهكرين | 1 |
| لايوجد | 1 |
| يكون سهل التعامل معه | 1 |
| ،،، | 1 |
| الله يوفقكم ونشوف ابداعكم | 1 |
| لايوجد شي | 2 |
| اتمنى ان يكون فتره تجريبيه وان يكون سعره معقول | 1 |
| نتوضيح سرية الرسائل حتى للتطبيق وعدم جمعه لاي معلومات خاصة | 1 |

Figure 42: Elicitation's Questionnaire's Response

# Appendix C: User Acceptance Testing's Questionaries

# 11 Appendix C: User Acceptance Testing's Questionaries

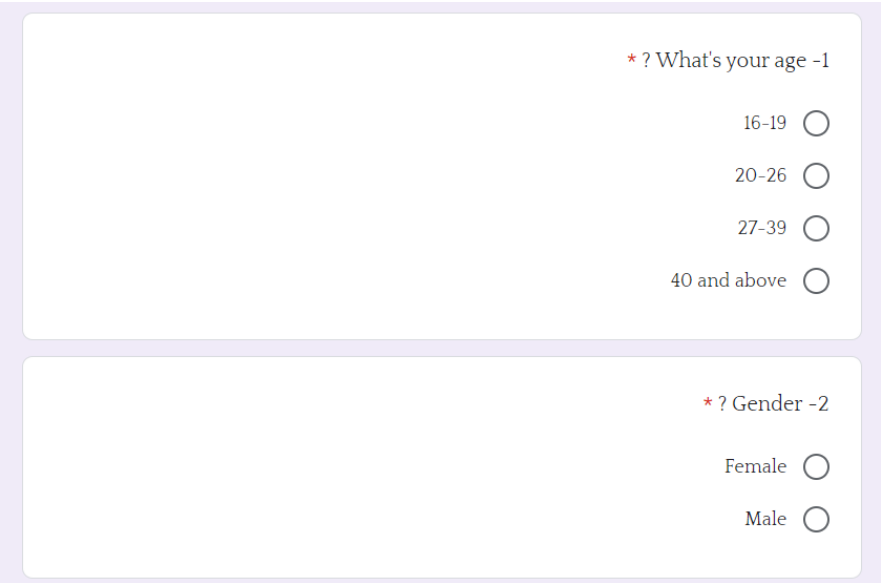


Figure 43: Testing's Questionaries

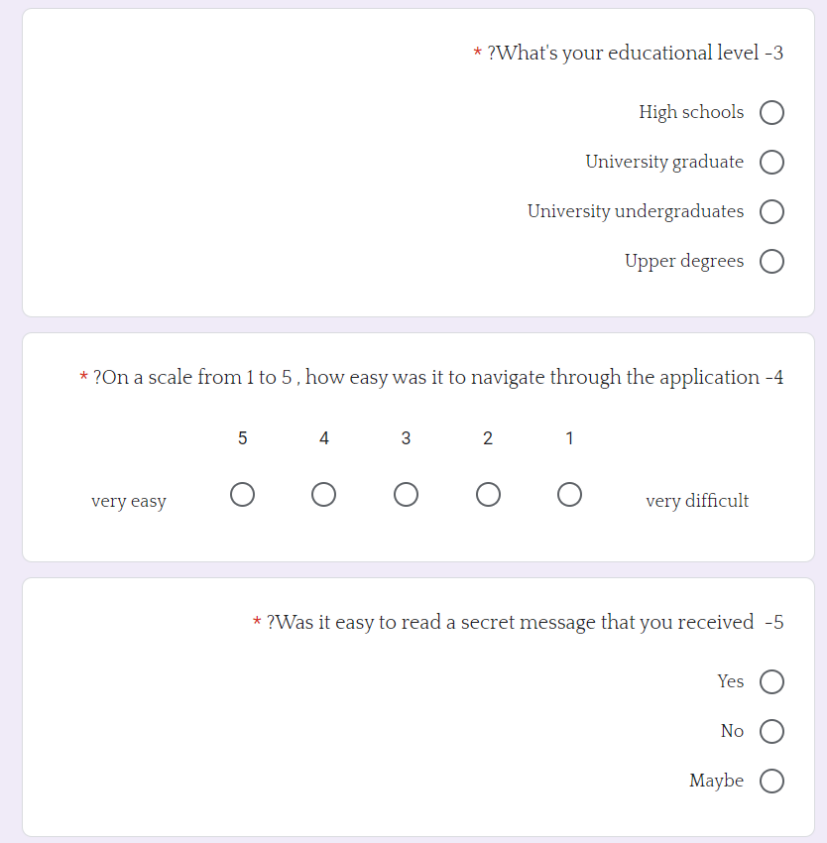


Figure 44: Testing's Questionaries

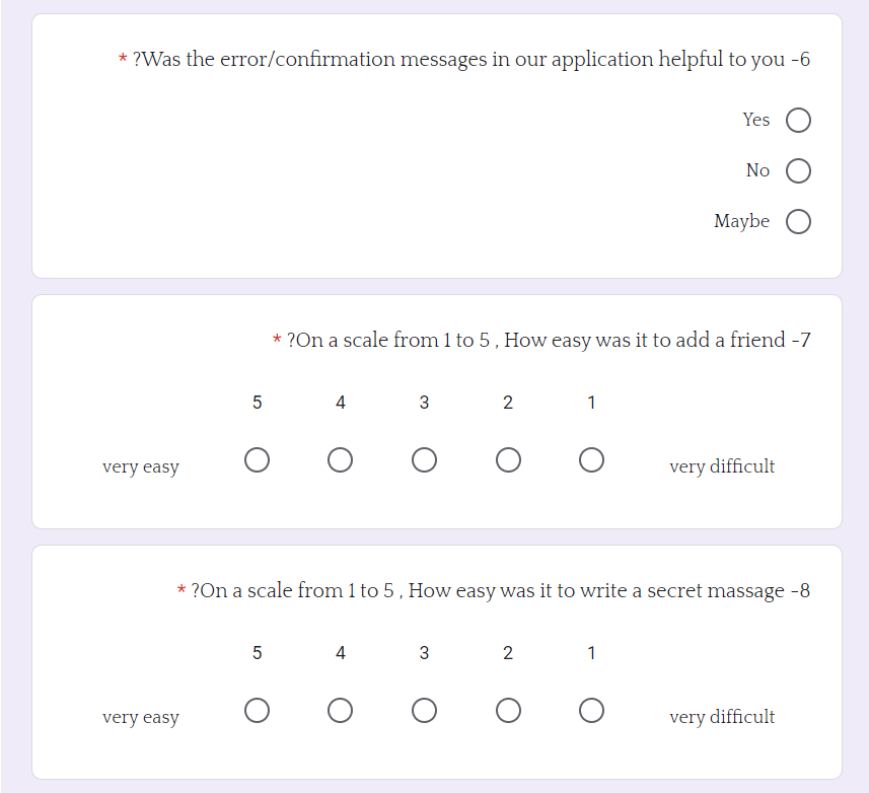


Figure 45: Testing's Questionaries

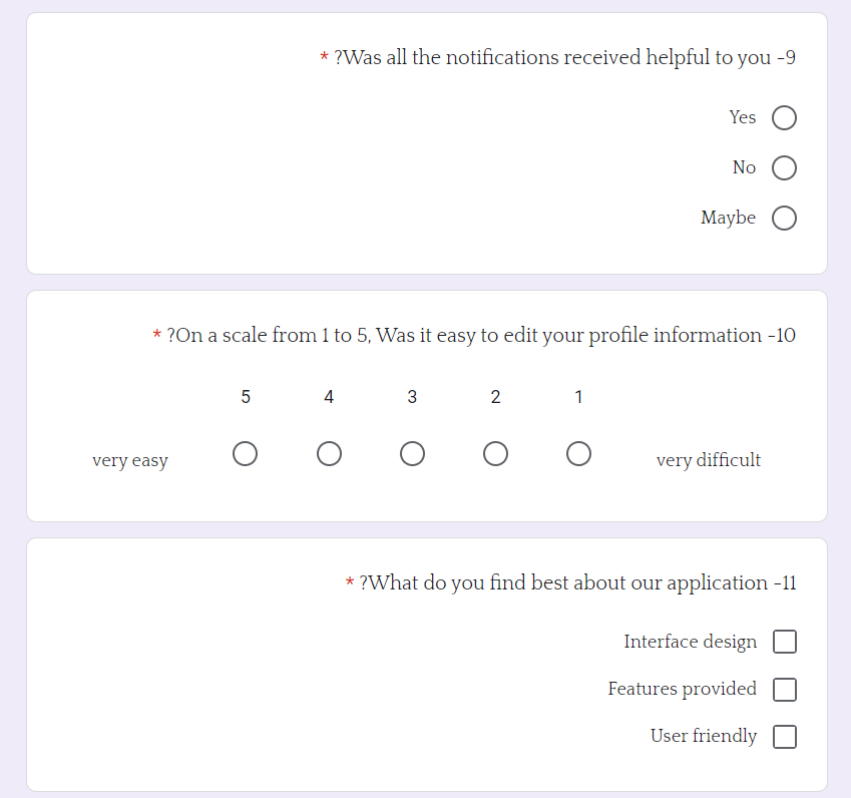


Figure 46: Testing's Questionaries

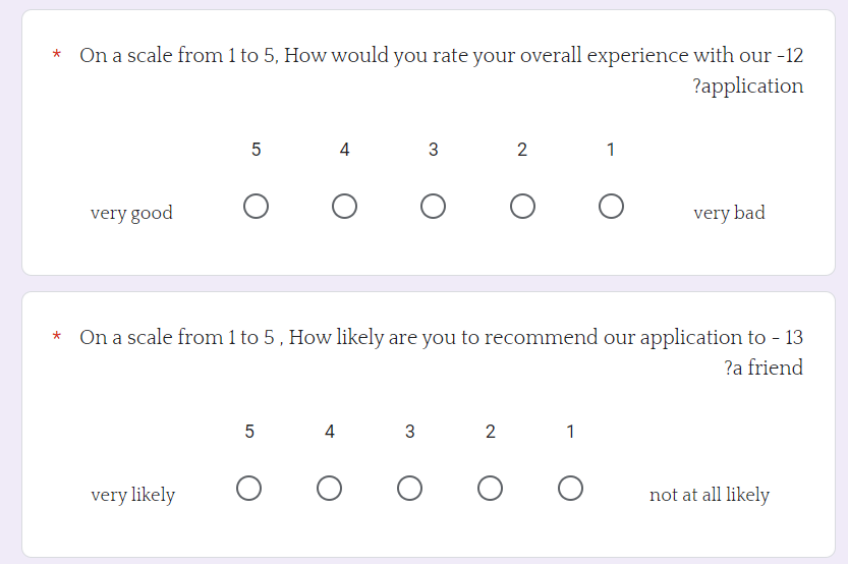


Figure 47: Testing's Questionaries