Encrypted Messaging Project

Analysis 1. and 2. Introduction:

The project idea is to make an app that allows users to send messages through an encrypted channel, this addresses the problem of unsecure messaging and untrustworthy messaging applications. I am also very interested in and want to learn more about cyber security and cryptography which will both be very relevant to this project. The project could be used by anyone who want to be able to communicate without the worry of their private messages being monitored. Although I imagine that it will most likely be used by other people who have an interest in cyber security and digital privacy most likely hobbyists between the ages of 16 and 40.

The main goal of the project is to give people who are interested in cyber security and encryption a way to communicate without having to worry about data privacy, this goal lends itself well to a computational approach because only with a computational approach can people communicate almost instantly with anyone from around the world. This makes a computational approach not just best practice but a necessity.

To reinforce the security side of the project, messages will be deleted from the database as soon as it is confirmed that the message has been received. As well as this, messages will be encrypted on the device before the database even sees it. All the I will be able to see is a set of random characters connected to a single use address which will be decrypted by the server to get an Ip address which it will send to the intended recipient. This is important because it allows the user to send and receive messages without me or anyone else being able to see it.

The challenges of this project are that I do not have the ability to make a very complex encryption algorithm therefore I will be focused on making a simple algorithm that only does the essentials, I also intend to learn as much as I can to better understand how encryption works there are many online tutorials I can use for this. Another challenge is that building a remote database that does not keep messages (or who is sending messages to who) for longer than it needs to be something that I have never attempted before. A third is the fact that I have never developed an app before. For all these problems there are resources I can use to learn more, for example universities like Stanford have free online courses for encryption.

For this project to be successful I will need to a. learn about how databases, encryption and instant messaging work b. find a way to implement these ideas in a way that works in the given time I must finish this project and c. get feedback from people who are interested in my project so I can customize the project to their ideas/ needs.

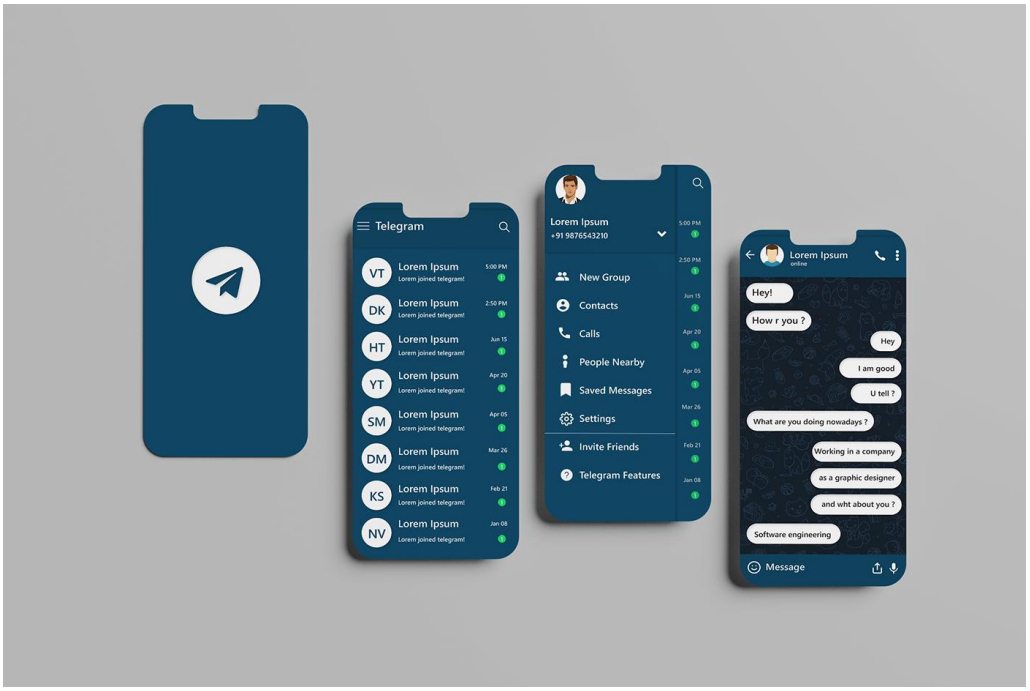
Other technical details:

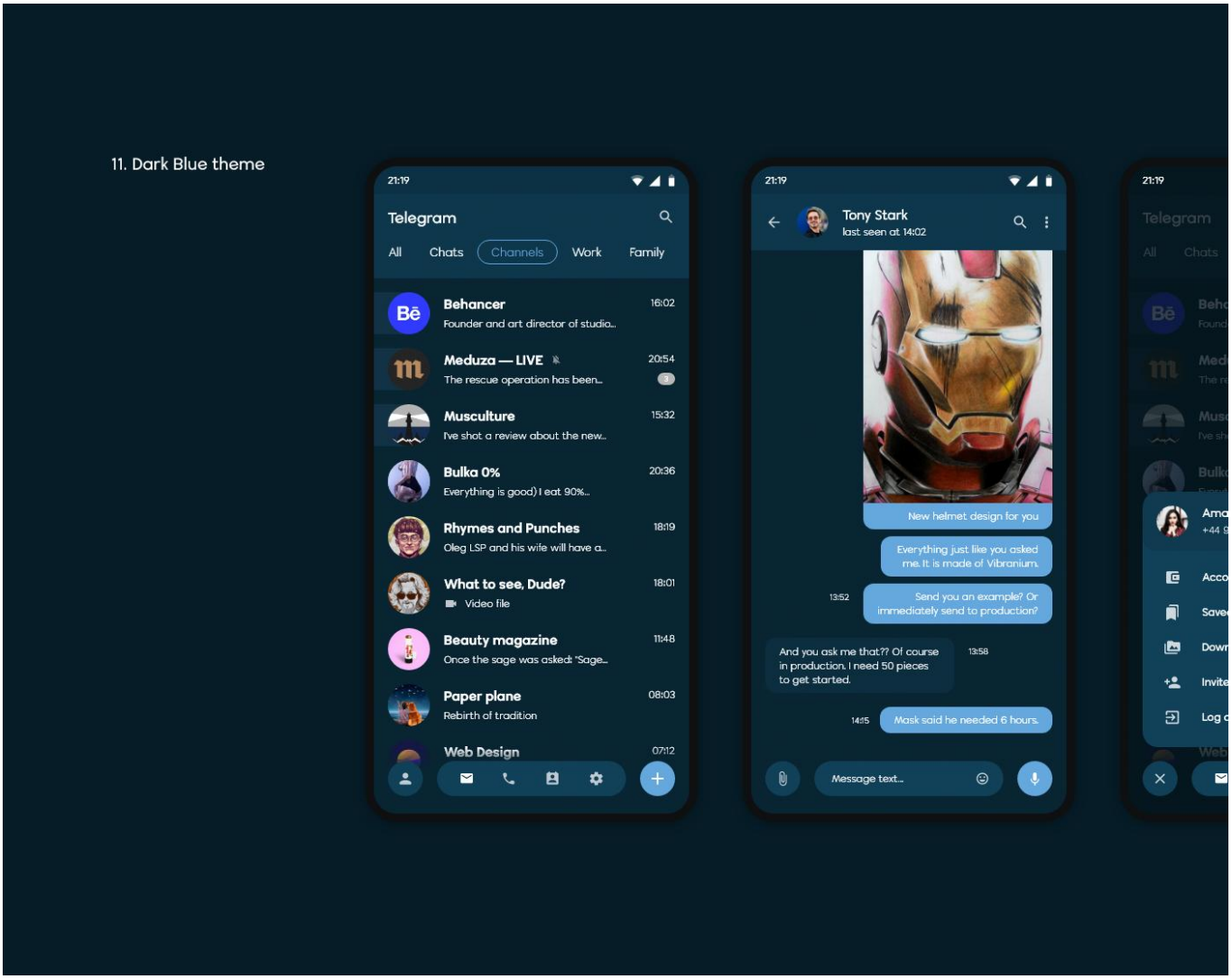
- The app will be able to send text messages.
- The app will be able to send images.
- The app will be written in Java.
- The app will be deigned to be used on android.
- The app will use client-side encryption.
- I will develop a simple messaging system.
- I will use an existing encryption algorithm

Analysis 3. research existing solutions:

Example 1, Telegram:

The first example Telegram is very good as it uses an end-to-end encryption algorithm and has an intuitive Ui design that is easy to navigate even for new users. It also allows users to customise their accounts and have multiple accounts. Telegram also can create group chats and allows users to customise who can and can't send messages on the group chats. However, registration requires a phone number and end-to-end encryption isn’t used for all messaging.



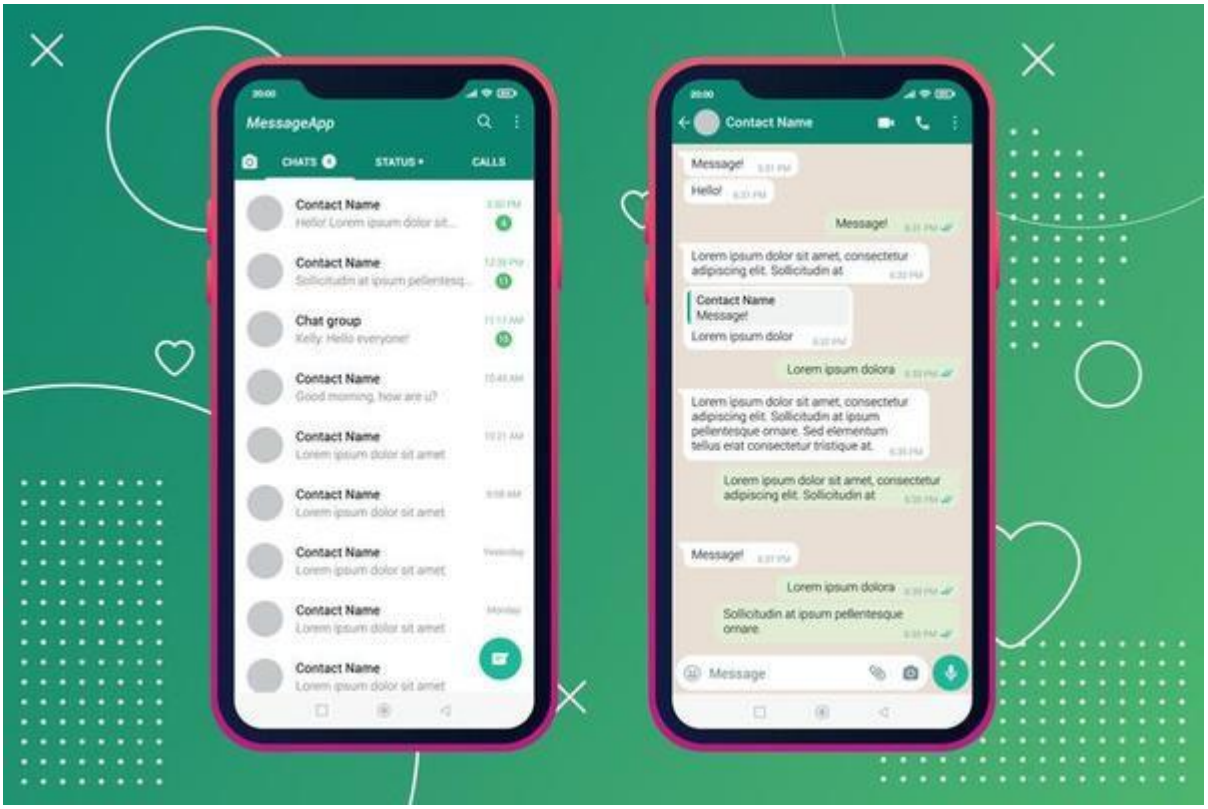


Positives:	**Negatives:**
Very clear and easy to read Ui design. All the colours allow for better readability meaning that people with worse eyesight can still read text on the app.	To make an account users need to give their phone number.
End-to-end encryption is used as well as a custom encryption protocol.	End-to-end encryption isn't used for most chats, for private chats client-server encryption is used.
The app is available on android, windows, iOS and Linux. This allows more people to communicate on the application.	Users don’t get the option to reject messages before they are sent.

Example 2. WhatsApp:

My second example of WhatsApp is necessary because the app is the most used messaging app in the world at a reported 2 billion users beating out WeChat by 700 million. It has also been accused of sharing and selling user data which most people would say is a privacy violation, whether this is true I don’t know. WhatsApp is also considered a template for how a messaging app should be many of its features like stickers have been copied and popularised through WhatsApp showing its influence on other messaging apps.

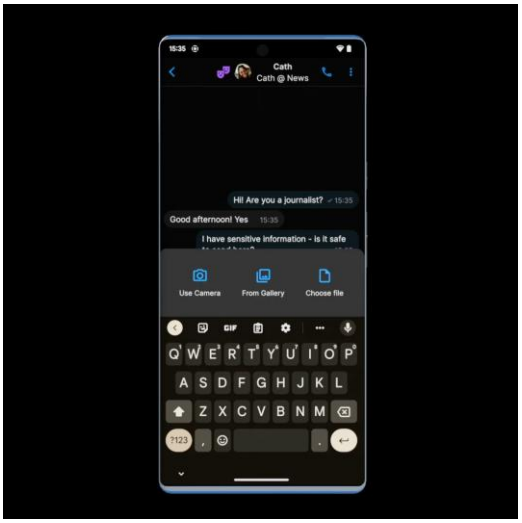

WhatsApp
Chats 6
Updates
Calls
Whitman Chat
11:26 AM
Sarah: For tn: or ?
Jack Whitman
9:28 AM
Video
Jane Pearson
8:15 AM
Dinner soon?
Thomas Stewart
8:03 AM
GIF
Francis Whitman
Yesterday
pls tell me you follow SingleCa...
Alice Whitman
Yesterday
Mom: How was this 10 yrs ago??
Jane Whitman
Did you watch the game?
Victor Whitman
Let's chat about it another tim
WhatsApp
Ask Meta AI or Search
All
Unread
Groups
Whitman Chat
11:26 AM
Sarah: For tn: or ?
Jack Whitman
9:28 AM
Sticker
Jane Pearson
8:15 AM
Dinner soon?
Thomas Stewart
8:03 AM
GIF
Francis Whitman
Yesterday
pls tell me you follow SingleCatClu...
Alice Whitman
Yesterday
Mom: How was this 10 yrs a
Jane Whitman
Chats
Updates
Communities
Calls



Positives	Negatives
Very intuitive Ui design. The Ui is simple and easy for new users to understand even if they haven't used an app like WhatsApp before.	Limited customization features. Apart from the background and a few other things you can't customize much on WhatsApp, although this might suit the userbase more.
End to end encryption. All data is encrypted on the phone and is not decrypted until it reaches the recipient, this means that data is much less likely to be seen by other people.	Data is often shared with other companies connected to WhatsApp like Facebook which many people see as a breach of privacy.
Cross-platform access. You can log into your account on many devices this allows for	Historically WhatsApp has been vulnerable to malware attacks. Although this could just be because of how many people use the site.

Example 3. SimpleX

I have added this example because it is considered by many to be the most secure messaging app in existence. This is because all data is stored on the user's phone and the app is designed in a way that makes it hard for the company to collect much data on the users. But it has the problem of messages being slow and taking much longer to send than the other examples, this is because of the privacy features which slow the app down.



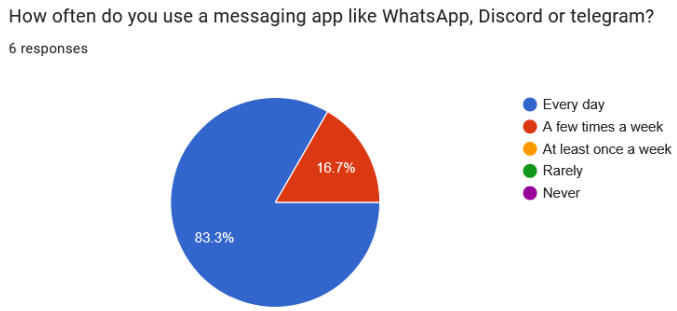
Positives	Negatives
You don't need to give your phone number to use the app so it completely anonymous.	Can have problems sending messages quickly because of no phone number collection. Messages are slow.
The Ui is simple and easy to use. It takes design elements from other apps like WhatsApp giving new users some sense of familiarity.	No disappearing messages like on the other 2 examples I have.
You can create multiple accounts unrelated to each other with ease. This allows you to appear as someone else if you want.	No automatic discovery of contacts like on the other 2 examples making it harder to find your contacts.

Analysis 4. Stakeholders:

The stakeholders for my application would most likely be people who have an interest in cyber security. Most likely between the ages 18 and 30. They could use my application for private messaging and group announcement channels. The stakeholders are a small group because it isn't realistic for anyone who isn't interested in cyber security to be interested in a cyber security focused application. The requirements for this group would be a well encrypted messaging protocol and the knowledge that I wouldn't be able to see their messages. I am planning on achieving this by using an in-memory database and end to end encryption.

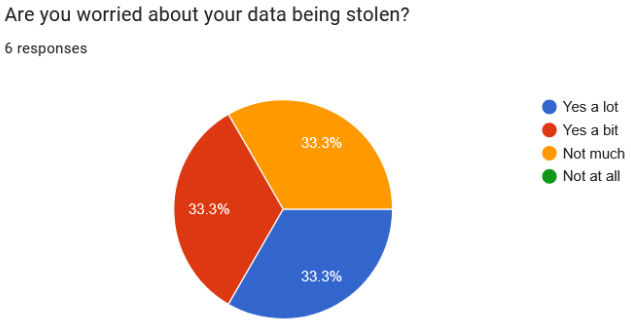
Results of the servery:

Question 1: How often do you use a messaging app like WhatsApp, Discord or Telegram?



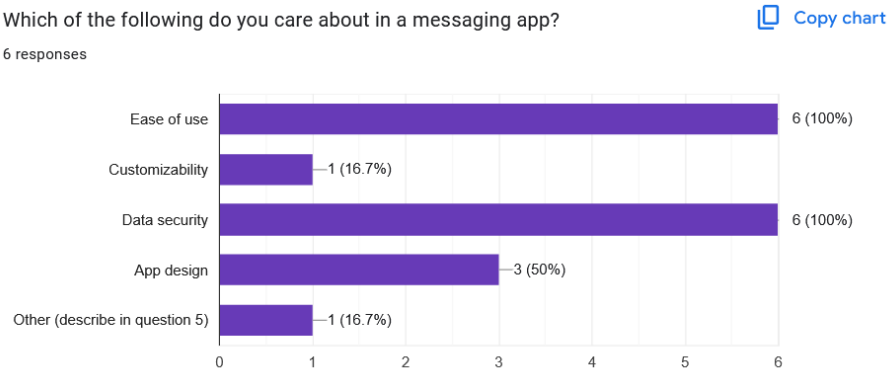
This shows that the stakeholders already use a messaging app like what I am planning on making this means that there is some limited interest in the application.

Question 2: Are you worried about your data being stolen?



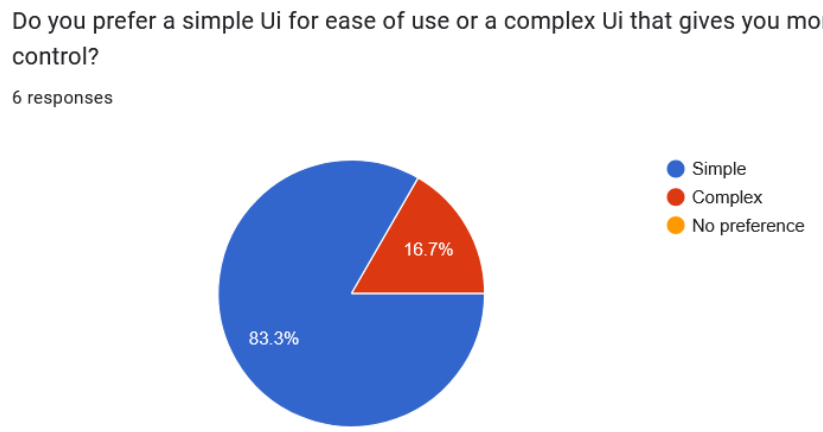
The majority said that they are worried about their data being stolen, this could mean that the stakeholders don't believe other applications care enough about their data being stolen. This shows that there is an interest in the project since it is a data security focused project.

Question 3: Which of the following do you care about in a messaging app?



The two most common answers were data security and ease of use, 100% of the stakeholder's care about data security showing the relevance of the project. The only other was gifs which although isn't a priority for the project I will look further into.

Question 5: Do you prefer a simple Ui for ease of use or a complex Ui that gives you more control?



As we can see the overwhelming majority of the stakeholders preferred a simple Ui design so I will be making a simpler Ui design than I was planning on since only 16% of the stakeholders want a more technical Ui design.

Question 6: What is the main thing you would want from a messaging app, be as specific as possible.

A unique feature that would give me a reason to use it above the competition. e.g. Discord's servers, snapchat's disappearing messages and snapmaps, whatsapp's formality and versatility, etc
Idk
Decentralisation and end-to-end encryptment
a secure and private message that you can talk to individuals or groups
An easy to use UI, quick messages

This question I think had the most useful answers, 2 of the answers are privacy related showing the usefulness of the project. The first answer is the most interesting, although I think all the ideas would be out of the scope of the project, I think disappearing messages would be doable and useful for the project.

Analysis 5 and 6 requirements:

Essential:

- End to end encryption, this is the whole point of the project. This feature would allow the user to know that the messages they are sending cannot be seen easily by me or any other unwanted person.
- A simple Ui that is easy to use and not overcomplicated, this has been clear in the survey that the stakeholders want a simple Ui that they can easily understand.
- A database that receives the message from the sender and sends it to the receiver, this is important because without this the project will not be able to send messages.

Non-essential:

- Making group as well as one to one messaging options, this has been requested by the stakeholders but is not massively important because the project would still work without it.
- Deleting messages, this has also been requested by the stakeholders and since it would be relatively easy to implement and would make the project more user friendly, I am including it here.
- Stickers, this again has been requested by the stakeholders but since it might be harder than some of the other requirements and is not the most popular request from the stakeholders, but I am still including it because almost all the other solutions that exist have this feature.

Limitations:

I would have liked to be able to allow users to create servers like in discord but I am leaving this out because it is not an essential feature that the project would need to work and because it would have taken too much time to develop given the short time and limited resources I have to crate this project.

Another limitation on the project that I won't be able to implement is decentralisation, which was requested by one of the stakeholders, I won't be able to implement this because I do not have access to the resources to have a decentralised network of servers it also is not an essential feature for the project to function.

Hardware and software requirements:

Hardware: A phone that has at least 4GB of ram and a 64-bit processor.

Software: Minimum for android is android 13 or higher. For iPhone iOS 14 or higher.

Success criteria:

1. **An app that works on android has been developed. This allows many people to access the application.**
2. **That app can send a receive messages to/ from other people with the app. This allows for communication which is the main goal.**
3. **The app encrypts all messages. This protects the users.**
4. **That app should have an easy to navigate UI. This allows new users to use the app.**
5. **The app should send the message to the recipient through a server. This means that users can't find out details about who they are communicating with since it is done through a server.**
6. **The person who owns the server shouldn't be able to access the message. This also protects the users.**