Introduction: -

Nowadays, real time communication is increasingly rapidly. The popularity of this communication is mostly seen through messages and chats, whether be it personal messages or confidential emails. Thus, there is a need of some protection system which could prevent other party from intruding the privacy of the users. Encrypted Chatbot is such project where two or more authorized users can have chats and messages with no invasion of external party. The chats are encrypted end to end with use of different encryption methods and keys. This is done so as to protect the privacy of people and their messages. For securing of information, it is encrypted before it leaves a user's device and can only be decrypted by the intended recipient. The encrypted message is called as cipher text. For encryption and decryption both the users will require a Key, which might be public or private. The key needs to be created, stored, and offered robust end-to-end secure encryption. Therefore, this project will create ciphertexts and keys taking plain texts as input. Have a safe space for your chats!

Research: -

https://www.scientificamerican.com/article/crack-the-code-make-a-caesar-cipher/

https://www.comparitech.com/blog/information-security/rsa-encryption/

In cryptography, two techniques are used for encryption and decryption: 1. Caesar Cipher 2. RSA algorithm.

Caesar Cipher: It shifts the entire alphabet by the number picked by the user. For example, if the number chosen by the user is 2, then A would be replaced by C, B would become D, and so on.

RSA algorithm: It is based on two prime numbers. These numbers are used for generating encryption and decryption keys.

Swot Analysis: -

STRENGHTS

The ultimate strength is privacy and security of our plain texts.

WEAKNESS
If the unauthorized user gets to know the key somehow then the text can go in wrong hands.

OPPORTUNITIES

The newer technologies can provide stronger encryption methods and keys.

THREAT

Exploitation of safety of public and private keys is one of the threats in this field.

REQUIREMENTS: -

HIGH LEVEL REQUIREMENTS: -

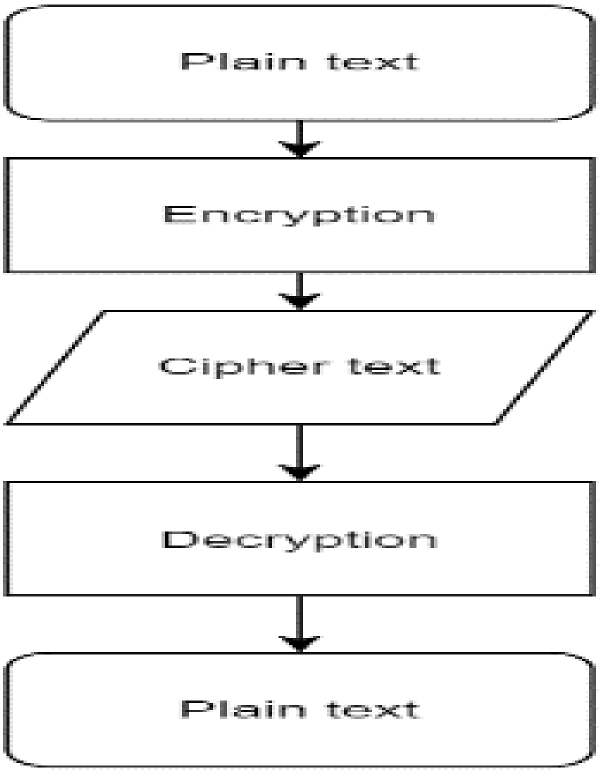
ID	DESCRIPTION	CATEGORY
HLR01	User should be able to encrypt message.	Technical
HLR02	User should be able to decrypt message.	Technical
HLR03	The key should be produced & matched.	Mathematical

LOW LEVEL REQUIREMENTS: -

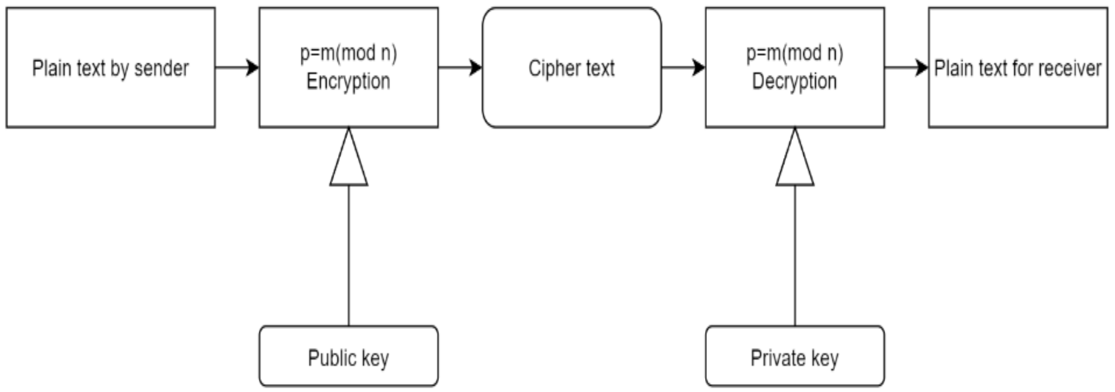
ID	DESCRIPTION	STATUS
LLR01	The plain text for encryption and decryption.	IMPLEMENTED
LLR02	The prime numbers for generation of key.	IMPLEMENTED
LLR03	Formulae for public and private keys.	IMPLEMENTED

Architecture

Behavioural Diagram: -



Structural Diagram: -



Tools: -

- draw.io (https://app.diagrams.net/)
- Microsoft Word
- or any other free tool

Implementation:

The project is designed for encrypting and decrypting messages between people. It uses two different types of algorithms for different types of messages. While considering numbers as plain text we use RSA Algorithm. It involves calculating key pairs with use of formulae. The first step of encrypting a message with RSA is to generate the keys. The following steps are involved in RSA algorithm:

Step 1: Generate the RSA modulus - N=p*q

Step 2: Derived Number (e) - The primary condition will be that there should be no common factor of (p-1) and (q-1) except 1

Step 3: Public key-The specified pair of numbers n and e forms the RSA public key and it is made public.

Step 4: Private Key **d** is calculated from the numbers p, q and e. The mathematical relationship between the numbers is as follows −

ed = 1 mod (p-1) (q-1)

Encryption Formula:-

C = Pe mod n

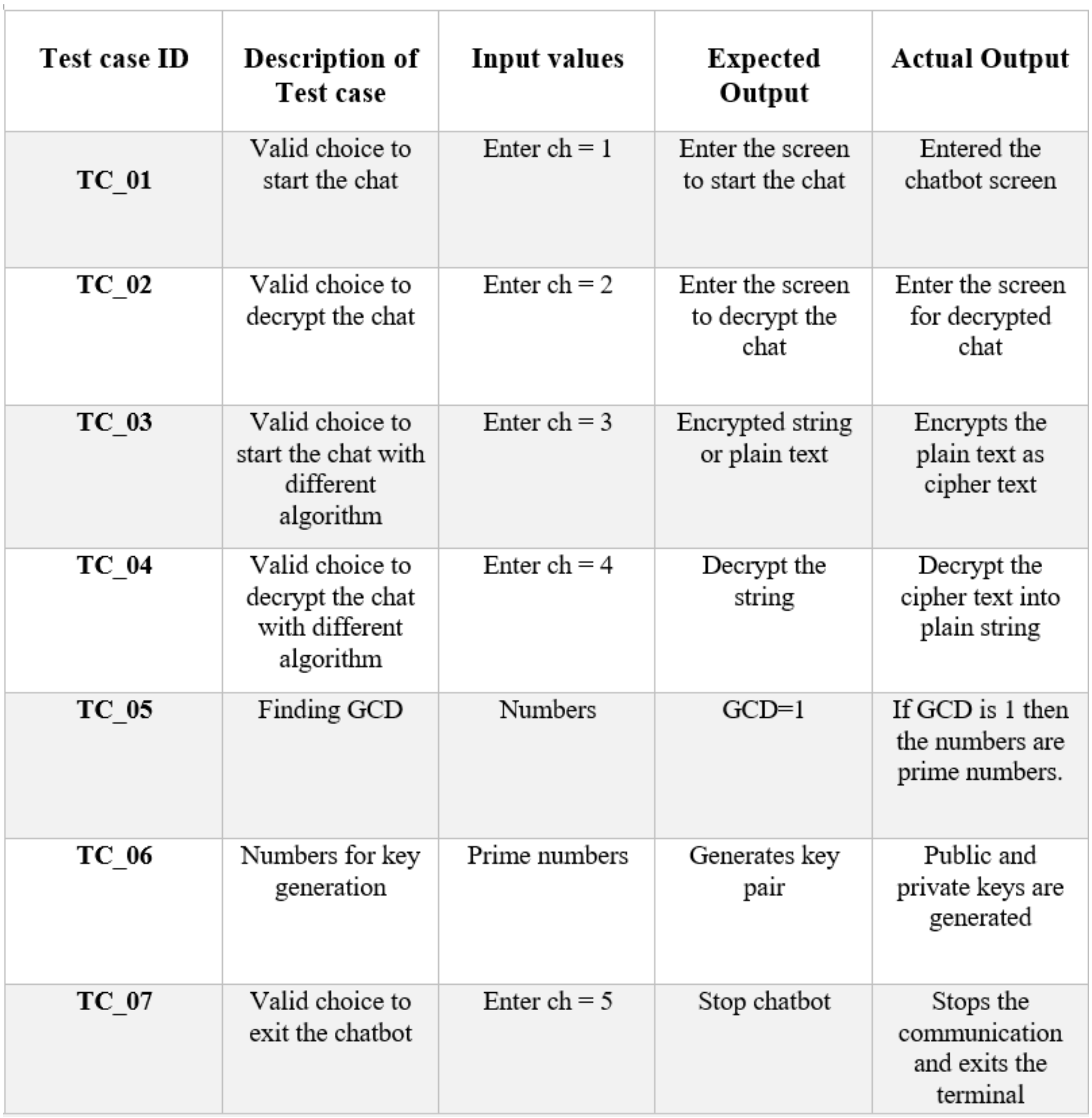
Decryption Formula

P = Cd mod n

For encrypting strings like messages, we can use Caesar cipher. The Caesar cipher, named after Roman Emperor Julius Caesar is one of the earliest and most widely known ciphers. In this you shift each letter by three spaces ahead. Thus, getting a new word or string.

The whole project is based on this above idea. After compiling the code without errors, we can start the chatbot. During execution you get to choose between different options of encryption or decryption. Once you choose the option you can start the conversation. The program will provide you with cipher texts and keys while encryption and vice-versa. After having a meaningful conversation, you can exit the console.

TestPlanAndOutput:



Test case ID	Description of Test case	Input values	Expected Output	Actual Output
TC_01	Valid choice to start the chat	Enter ch = 1	Enter the screen to start the chat	Entered the chatbot screen
TC_02	Valid choice to decrypt the chat	Enter ch = 2	Enter the screen to decrypt the chat	Enter the screen for decrypted chat
TC_03	Valid choice to start the chat with different algorithm	Enter ch = 3	Encrypted string or plain text	Encrypts the plain text as cipher text
TC_04	Valid choice to decrypt the chat with different algorithm	Enter ch = 4	Decrypt the string	Decrypt the cipher text into plain string
TC_05	Finding GCD	Numbers	GCD=1	If GCD is 1 then the numbers are prime numbers.
TC_06	Numbers for key generation	Prime numbers	Generates key pair	Public and private keys are generated
TC_07	Valid choice to exit the chatbot	Enter ch = 5	Stop chatbot	Stops the communication and exits the terminal