University of Windsor

A Project on Web Session Based Encrypted Messaging System

DEVELOPED BY,

Jainam Jatinbhai Shah
Master of Applied Computing
shah65@uwindsor.ca
Student ID # 110024454

Project report for COMP-8640 Security and Privacy – Internet

GUIDED BY,

Dr. Saeed Samet
School of Computer Science
University of Windsor

I. ABSTRACT

The Web session based encrypted messaging system will allow users to chat in a most secured form. In this system, the high-level discussions can be made between two ends. Basically, the main purpose of this system is to allow two parties to engage into the communication where no third party can intrude or watch. So, how it will work? It will create a temporary session which will be automatically destroyed after both the parties leave the chat. A session will work based on encryption algorithm so that chat remains confidential. Both the parties will send and receive messages based on encryption and then followed by decryption method. Primarily it is executed as a web-based program. This project is developed using ASP.NET C# technology and MySQL database. A cryptographic algorithm is be used to encode and decode the messages. Using this system one can create a room for more than two parties to indulge in the group conversation. To add more the security factor, users using the chatrooms will have to provide the PIN or Password to enter the room if there are more than two entities.

II. INTRODUCTION

This system was developed using the ASP.NET C# technology with the intention of providing the instant chat messaging service with the encryption technique to protect the data from intruders. The main goal of this project was to develop a system that provides secrecy, privacy and also to understand some of the concepts studied in the subject of COMP 8640 *Security and Privacy-Internet* at *University of Windsor*.

Basically, in this system, for an instance consider two parties wants to communicate on an urgent basis regarding some confidential topic where they can trust that the data privacy would be preserved; and no trace of communication can be found after the intention of communication is completed then they can choose this system to engage into the communication. So, our system provides a functionality where users can create a chat rooms using their email address. The cross verification of email address is done so that authenticated and trustable users only can join the chatrooms. Joining users needs to enter the passcode set by the owners of chatrooms to join the chat session so no other users can join the rooms. This passcode is provided in the email containing the link to join the chat rooms. Owners just need to provide the email address of the users they wanted to add as a member of chatrooms. This allows user to have full control, weather which person can access the conversation.

Now all the messages sent and received by the attending participants are stored and fetched from the database. Whenever any participant sends the message, the cryptographic algorithm encodes the message text and converts it into the cipher text. So, the message stored in the database is cipher text but not an original message plain-text. This enables the system with the data privacy. If an intruder tries to fetch or watch any conversation into the chatrooms, then they will receive the cipher text which can be only decoded by the users attending the chat conversation. Whenever participants receive any message then it is automatically decoded before displaying to the user using its private key.

Since this system has a passcode authentication, email verification and cryptographic algorithm to preserve the data security, this system can be called as an MFA (Multi-factor authentication) enabled system. Nowadays majority of messaging service providing systems are MFA enabled. The concept of MFA is based on providing as many possible multiple securities to the system to protect it from the hackers or unauthenticated users.

III. A SUITABLE CRYPTOGRAPHY

For an encryption and decryption, a suitable cryptographic algorithm is a very important factor to choose. For our system we have used symmetric key cryptography for the encryption/decryption. Let us learn some facts of symmetric key cryptography:

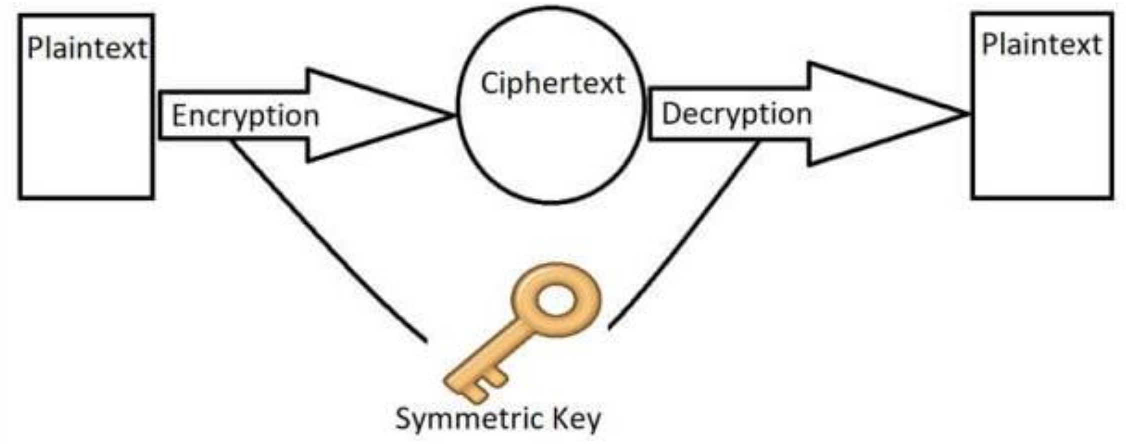


Figure 1 - Ref: csharap corner; Link: https://www.c-sharpcorner.com/

A symmetric key cryptography is a method where a single key is used to encrypt as well as decrypt the data. A Symmetric key cryptography is more reliable when using it for the faster results. It is the most secured form of communication if the encryption is strong enough and key is long enough to brute force it. For an instance, it would take a billion of years to guess the 128-bit key using same hardware.

In symmetric key cryptography, they key exchange happens over a secured channel. Both the parties will need a same secrete key to encrypt and decrypt the data. The key generation is done by on pseudo random key generator.

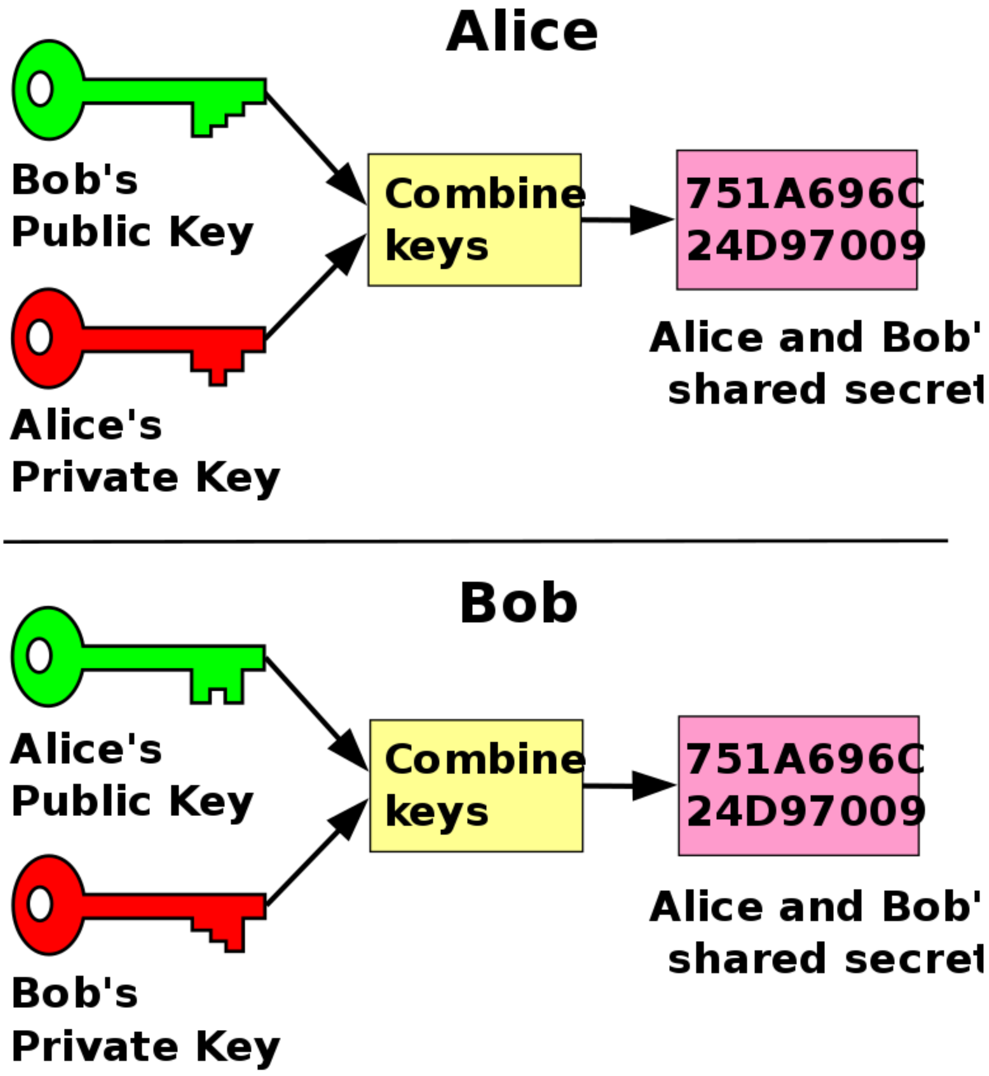


Figure 2 - Ref: Wikipedia; Link: Wikipedia/symmetric-key-cryptography/key-exchange/image

IV. ADVANCED ENCRYPTION SYSTEM (AES)

Here, we have used Advanced Encryption System (AES) to encrypt and decrypt the chat communication between parties. AES, which is formerly known as Rijndael is a symmetric key encryption algorithm developed by the two Belgian based crypt-programmers Vincent Rijmen and Joan Daemen.

How the algorithm works?

1) ***KeyExpansion*** – round keys are derived from AES key generator for each round. AES requires generates128-bit round key block for each round plus one more.

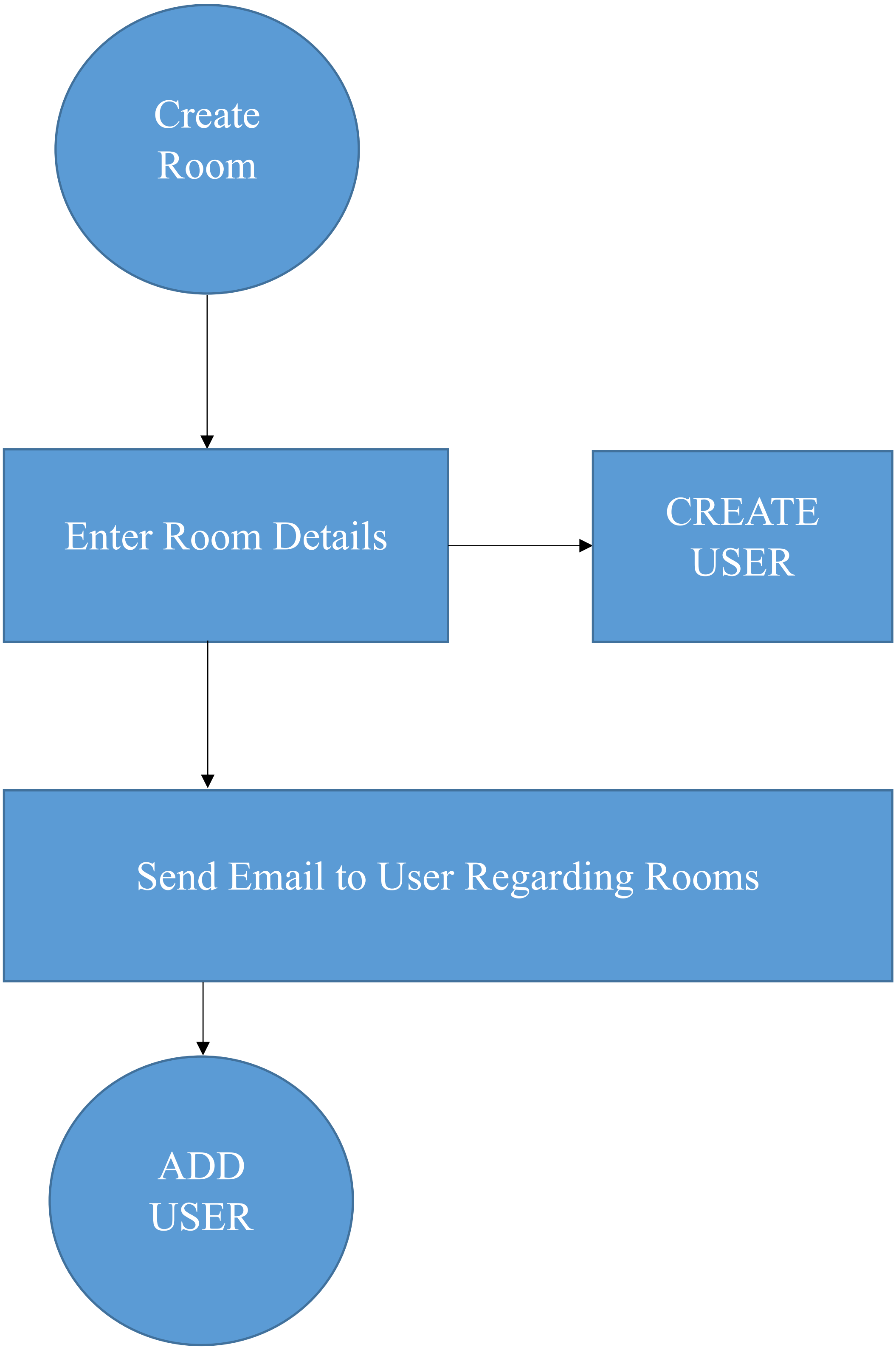
2) Initial round key addition:
    - ***AddRoundKey*** – Using bitwise XOR, each byte from state of round is merged to the byte of round key.

3) 9, 11 or 13 rounds:
    - ***SubBytes*** – According to the lookup table the byte is replaced in each step. It is a non-linear substitution step.
    - ***ShiftRows*** – a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.
    - ***MixColumns*** – a linear mixing operation which operates on the columns of the state, combining the four bytes in each column.
    - ***AddRoundKey***

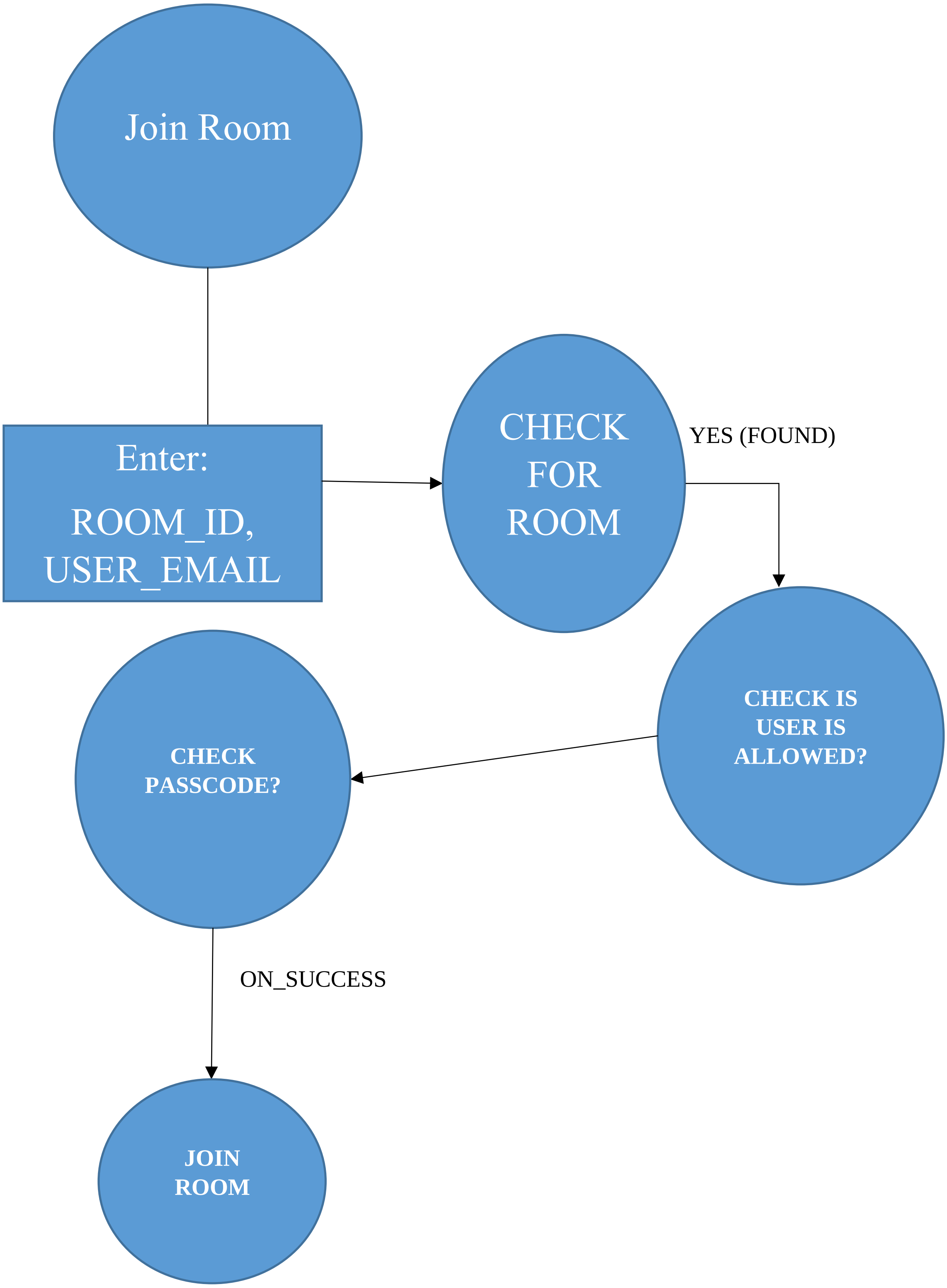
4) Final round (making 10, 12 or 14 rounds in total):
    - *SubBytes*
    - *ShiftRows*
    - *AddRoundKey*

V. SYSTEM ARCHITECTURE

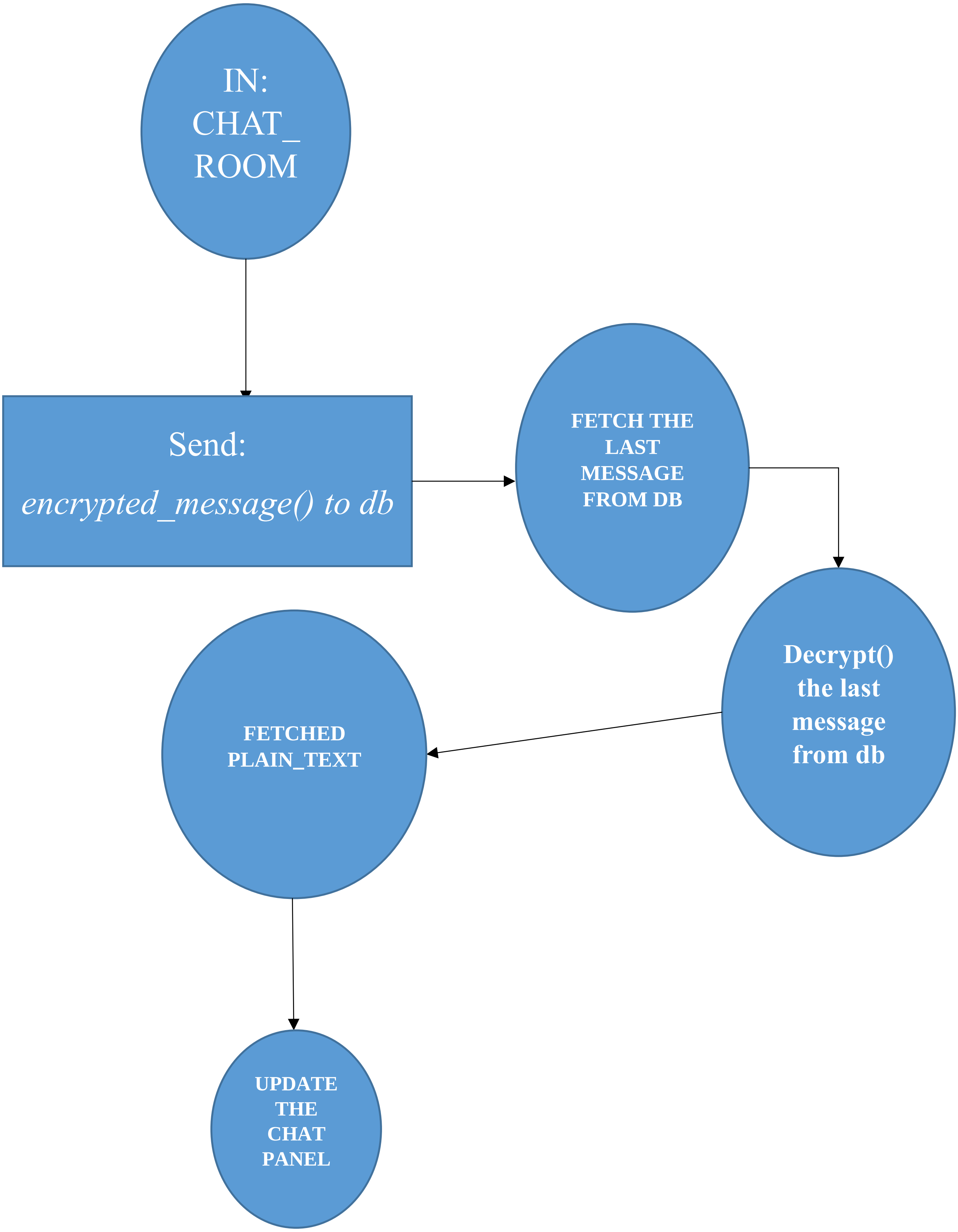
1) Create Room



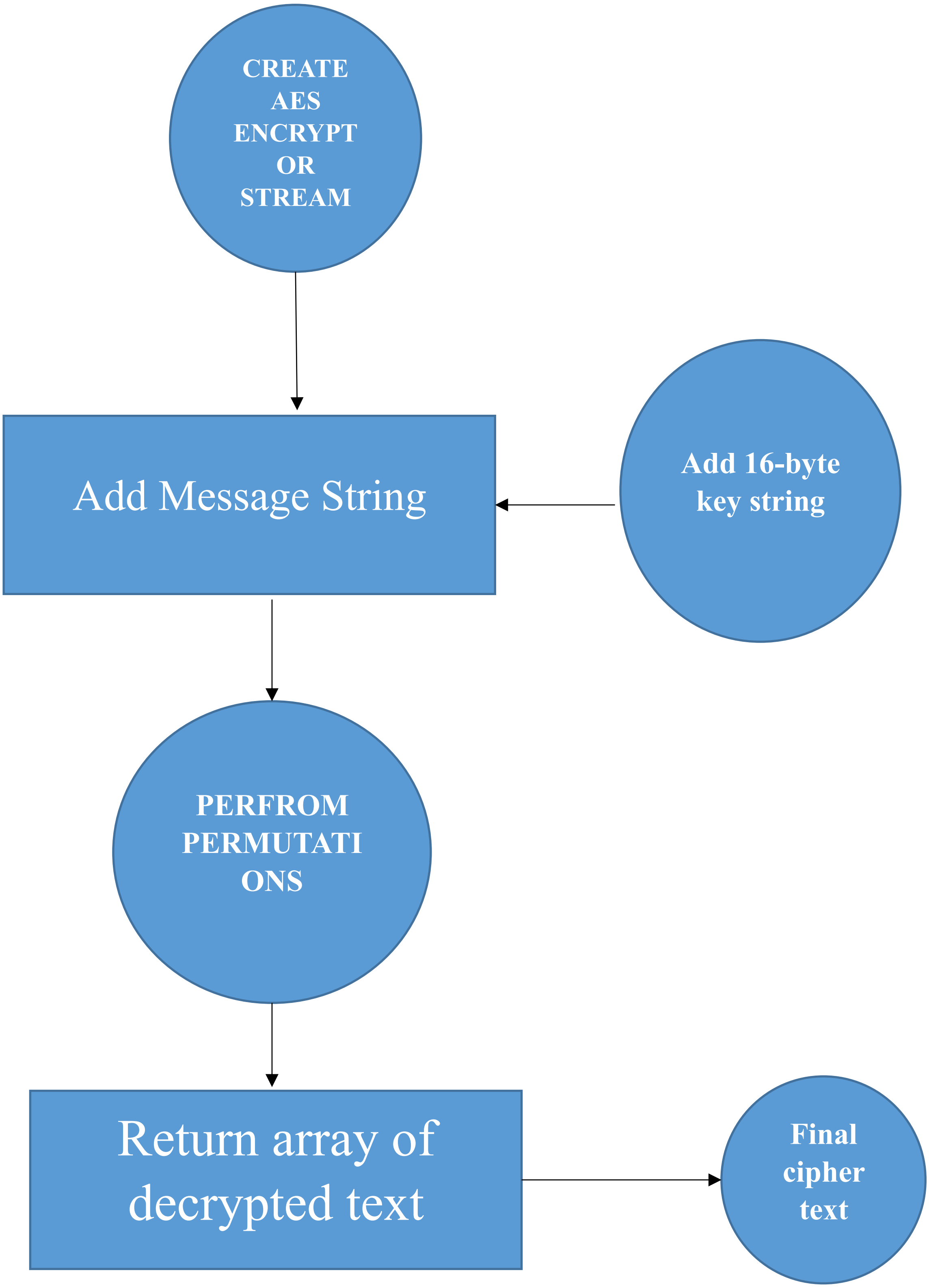
2) Join Room


Join Room
Enter:
ROOM_ID,
USER_EMAIL
CHECK FOR ROOM
YES (FOUND)
CHECK IS USER IS ALLOWED?
CHECK PASSCODE?
ON_SUCCESS
JOIN ROOM

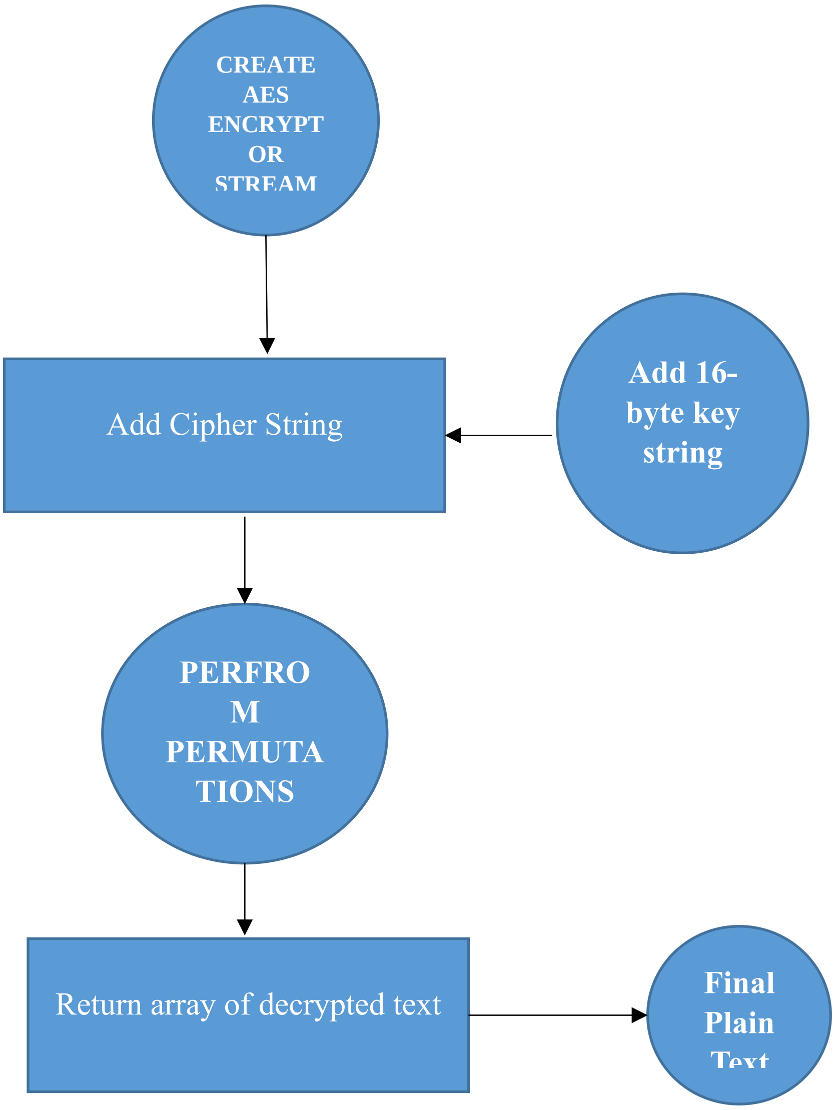
3) Chat Communication


IN:
CHAT_
ROOM
Send:
encrypted_message() to db
FETCH THE LAST MESSAGE FROM DB
Decrypt() the last message from db
FETCHED PLAIN_TEXT
UPDATE THE CHAT PANEL

4) Encryption

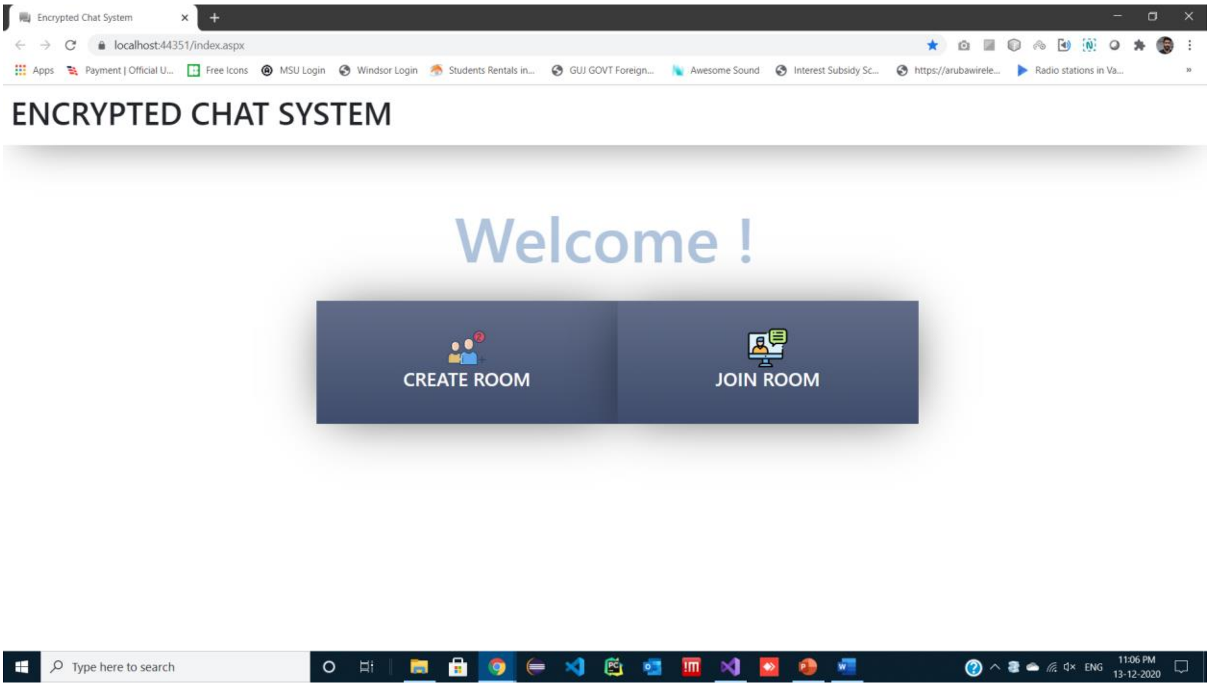


5) Decryption

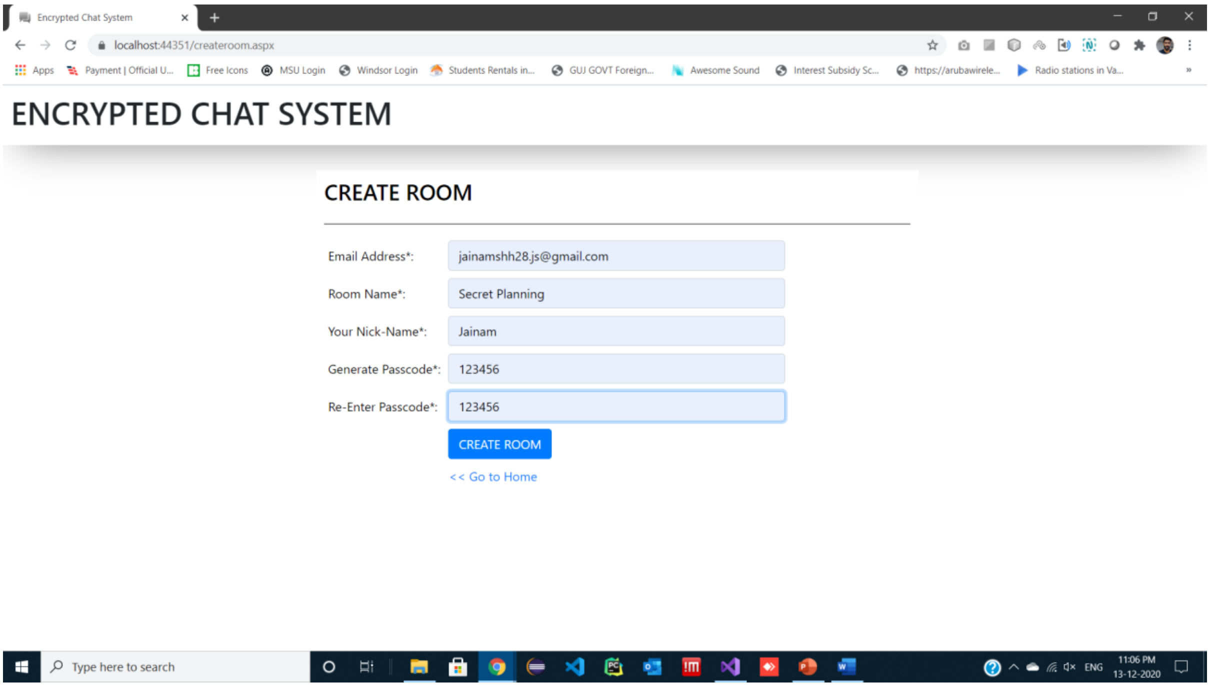


- **Technology Used:**
  - ASP.NET C#
- **For encryption / decryption**
  - ASP.NET library that enables the encryption / decryption
  - *System.Security.Cryptography;*
- **Tools and other languages:**
  - Visual Studio 2020
  - HTML5
  - BootStrap
  - JavaScript
  - CSS3
- **Database:**
  - MySQL Database Technology

VI. GRAPHICAL DEMO TO THE SYSTEM



Home Screen



Email Address*: jainamshh28.js@gmail.com

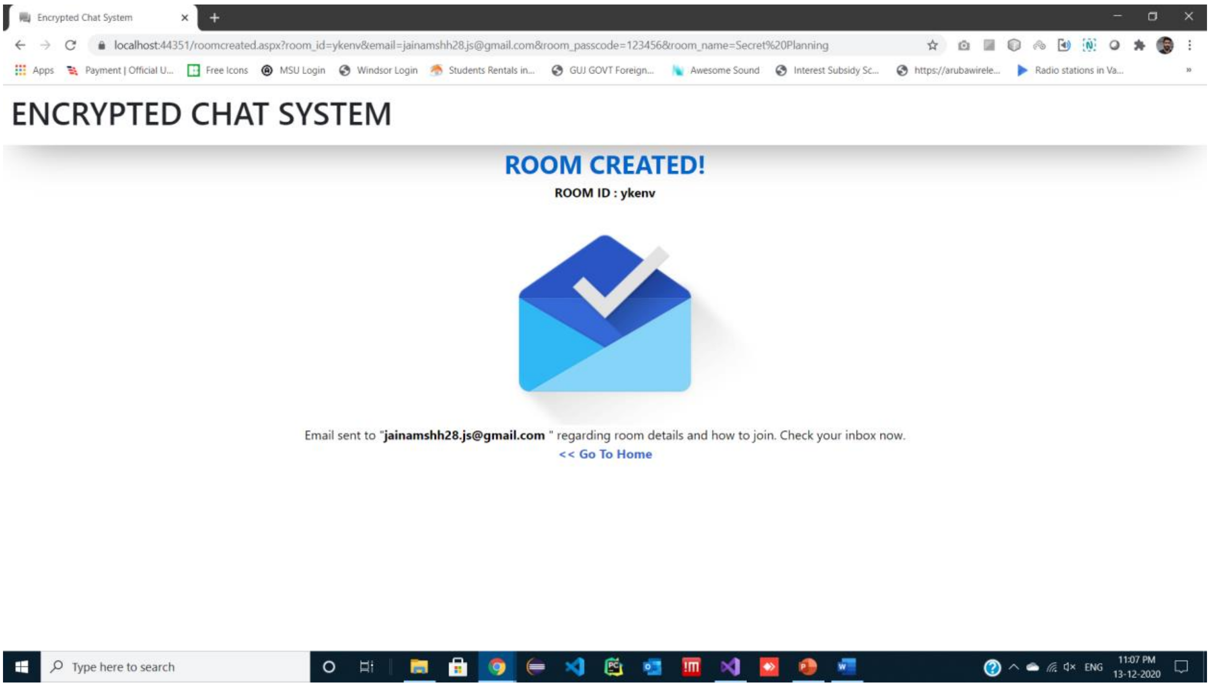
Room Name*: Secret Planning

Your Nick-Name*: Jainam

Generate Passcode*: 123456

Re-Enter Passcode*: 123456

Create Room



localhost:44351/roomcreated.aspx?room_id=ykenv&email=jainamshh28.js@gmail.com&room_passcode=123456&room_name=Secret%20Planning

ENCRYPTED CHAT SYSTEM

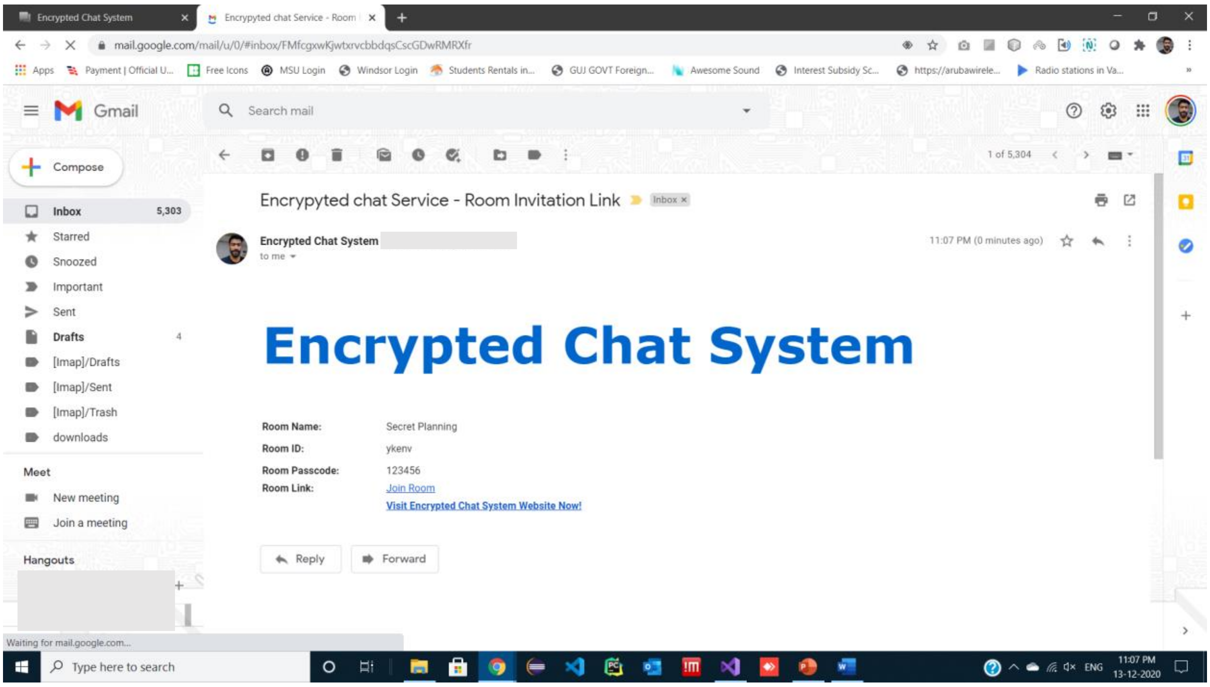
ROOM CREATED!

ROOM ID : ykenv

Email sent to "**jainamshh28.js@gmail.com** " regarding room details and how to join. Check your inbox now.

<< Go To Home

Room Creation Confirmation Page



mail.google.com/mail/u/0/#inbox/FMfcgxwKjwtxrvcbbdqsCscGDwRMRXfr

Encrypyted chat Service - Room Invitation Link

Encrypted Chat System [redacted]
to me

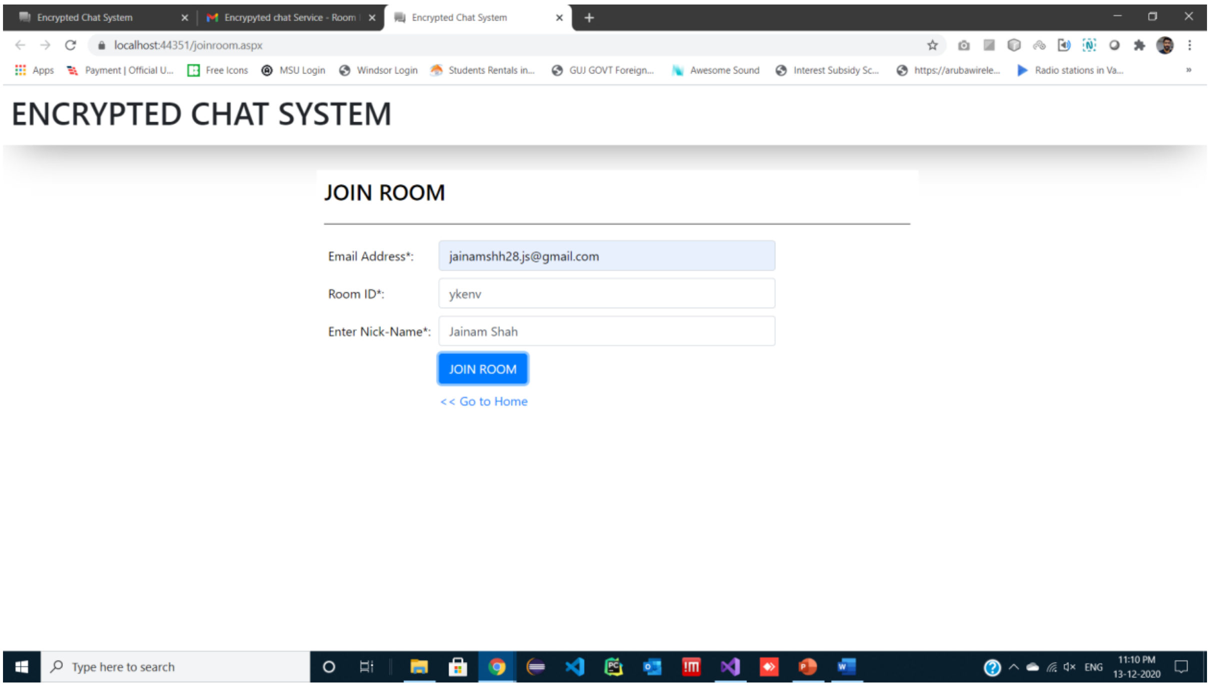
11:07 PM (0 minutes ago)

Encrypted Chat System

Room Name: Secret Planning
Room ID: ykenv
Room Passcode: 123456
Room Link: Join Room
Visit Encrypted Chat System Website Now!

[redacted]

Mail Confirmation of Room Creation and Joining Link.



localhost:44351/joinroom.aspx

ENCRYPTED CHAT SYSTEM

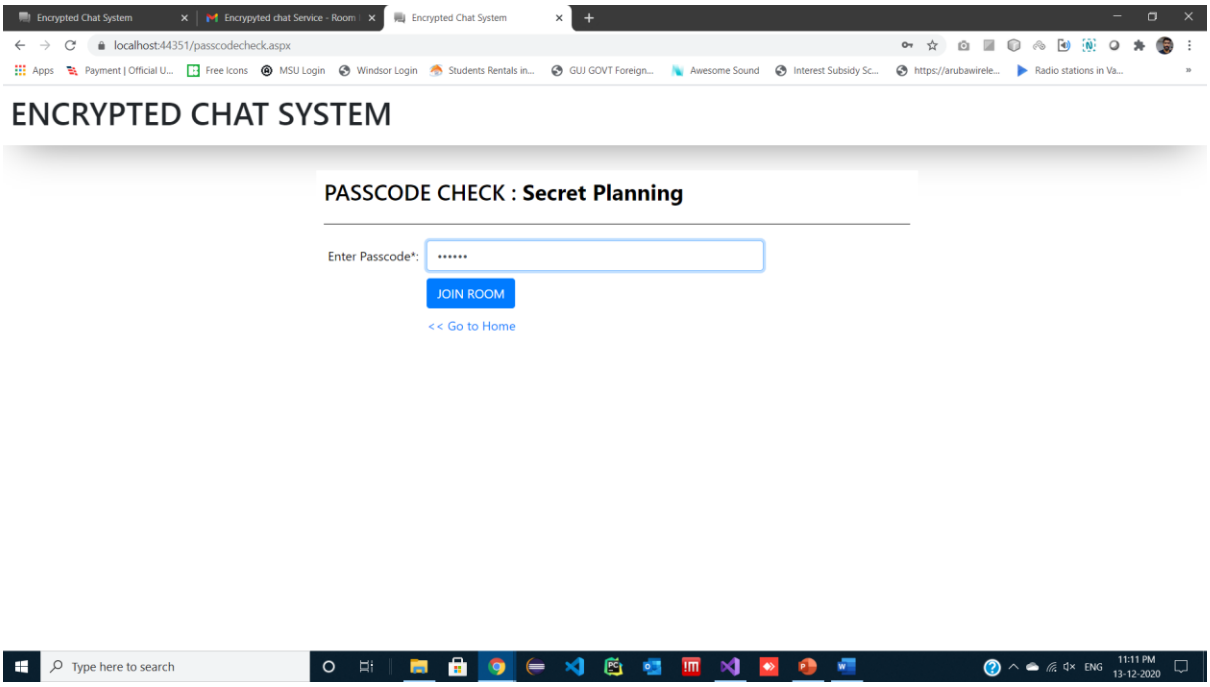
JOIN ROOM

Email Address*: jainamshh28.js@gmail.com

Room ID*: ykenv

Enter Nick-Name*: Jainam Shah

Joining a Room



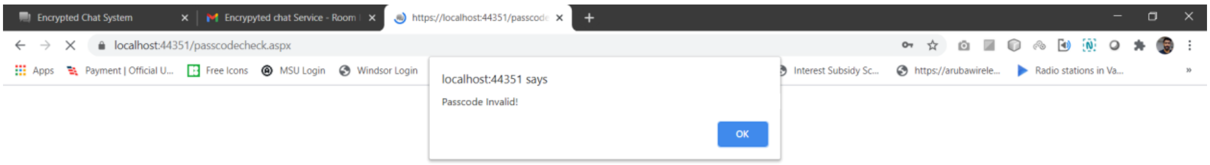
localhost:44351/passcodecheck.aspx

ENCRYPTED CHAT SYSTEM

PASSCODE CHECK : **Secret Planning**

Enter Passcode*: ••••••

Passcode verification for joining a room



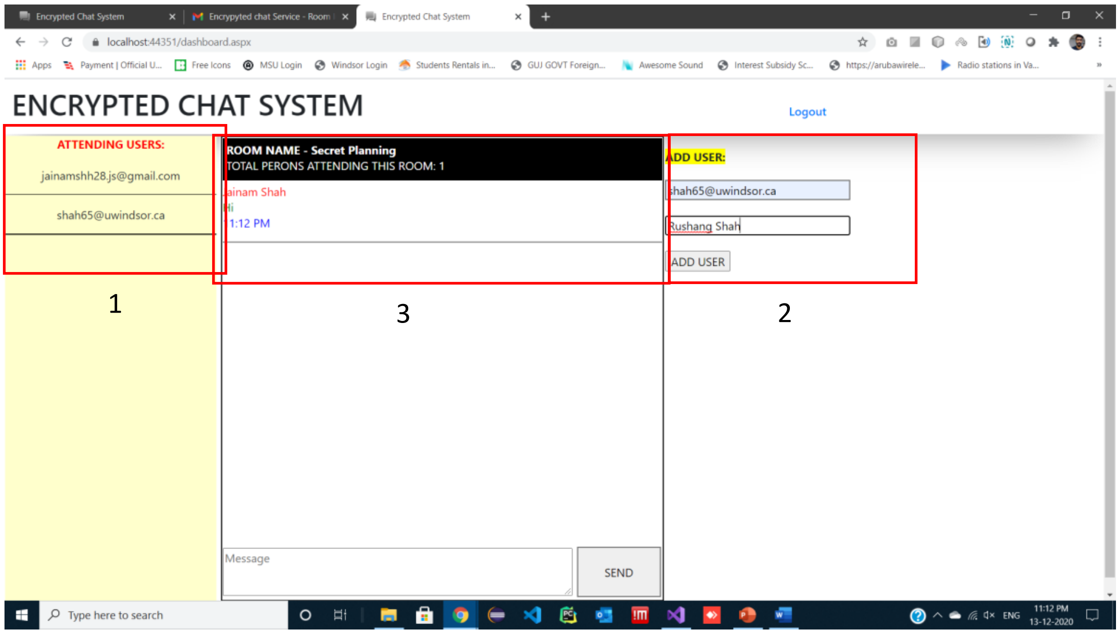
localhost:44351/passcodecheck.aspx

localhost:44351 says

Passcode Invalid!

Encrypted Chat System
Encrypypted chat Service - Room
https://localhost:44351/passcode
OK

Error on invalid passcode



ENCRYPTED CHAT SYSTEM

ATTENDING USERS:

jainamshh28.js@gmail.com

shah65@uwindsor.ca

ROOM NAME - Secret Planning
TOTAL PERONS ATTENDING THIS ROOM: 1

Jainam Shah
Hi
11:12 PM

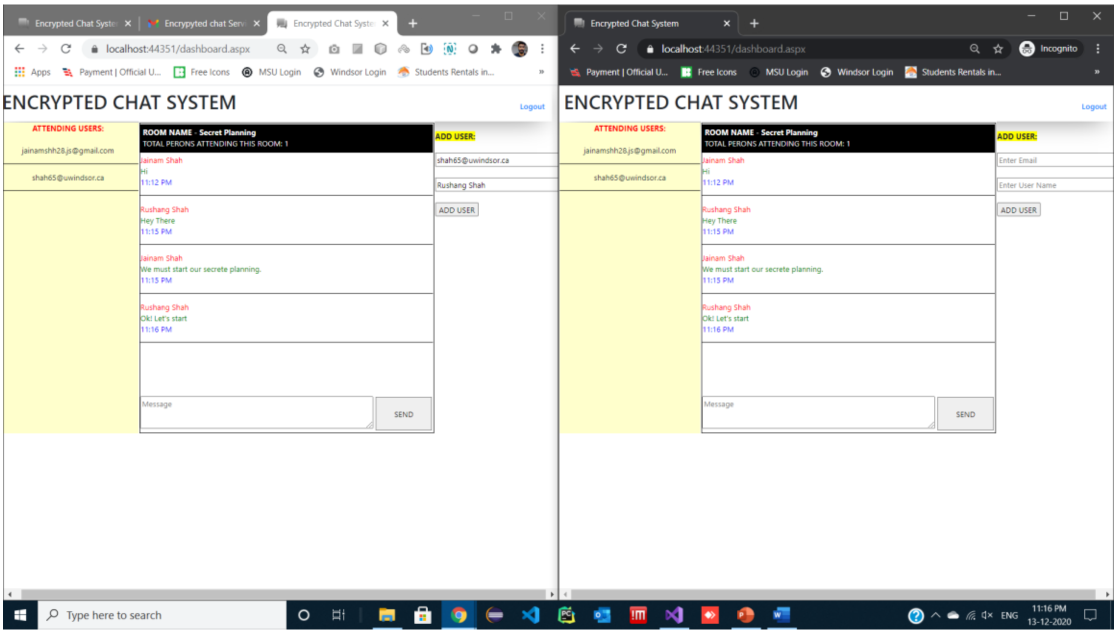
Message

ADD USER:

shah65@uwindsor.ca

Rushang Shah

Chat Room: 1) Shows who are attending the users. 2) Provides functionality to add the user. 3) A chat conversation which is automatically converted to the plain text.



ENCRYPTED CHAT SYSTEM

ATTENDING USERS:

jainamshh28.js@gmail.com

shah65@uwindsor.ca

ROOM NAME - Secret Planning
TOTAL PERONS ATTENDING THIS ROOM: 1

Jainam Shah
Hi
11:12 PM

Rushang Shah
Hey There
11:15 PM

Jainam Shah
We must start our secrete planning.
11:15 PM

Rushang Shah
Ok! Let's start
11:16 PM

Message

ADD USER:

shah65@uwindsor.ca

Rushang Shah

ENCRYPTED CHAT SYSTEM

ATTENDING USERS:

jainamshh28.js@gmail.com

shah65@uwindsor.ca

ROOM NAME - Secret Planning
TOTAL PERONS ATTENDING THIS ROOM: 1

Jainam Shah
Hi
11:12 PM

Rushang Shah
Hey There
11:15 PM

Jainam Shah
We must start our secrete planning.
11:15 PM

Rushang Shah
Ok! Let's start
11:16 PM

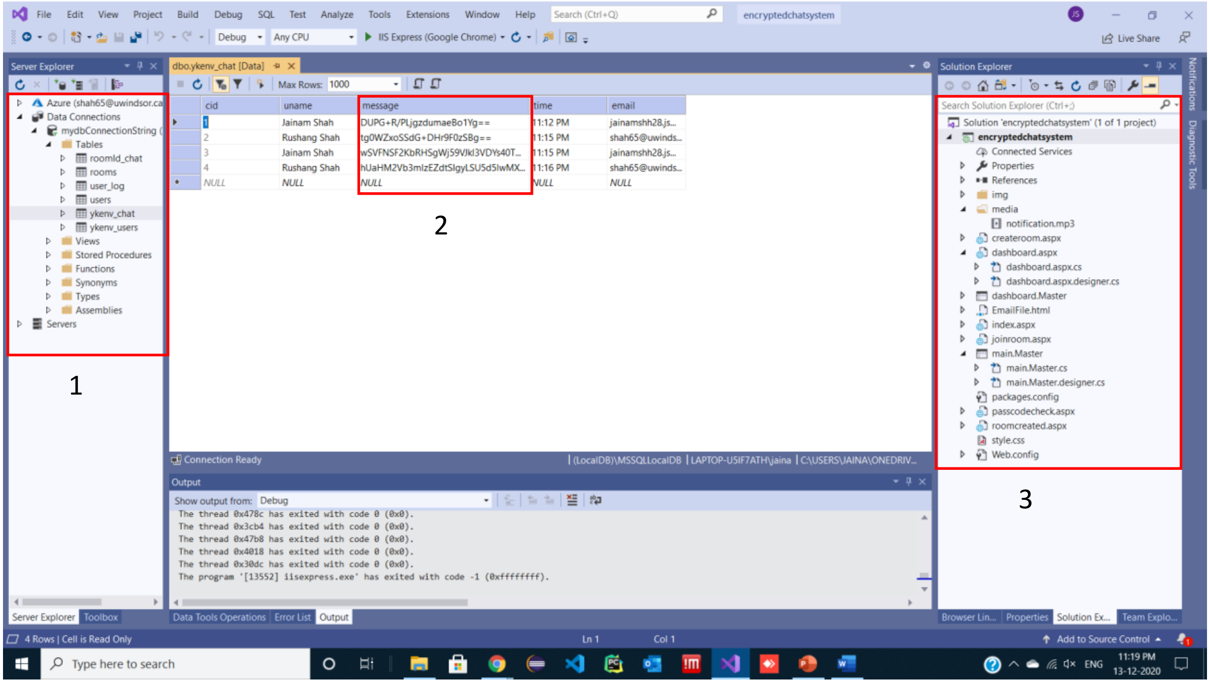
Message

ADD USER:

Enter Email

Enter User Name

Synchronous Messaging / Chat Conversation.



dbo.ykenv_chat [Data]

cid	uname	message	time	email
1	Jainam Shah	DUPG+R/PLjgzdumaeBo1Yg==	11:12 PM	jainamshh28.js...
2	Rushang Shah	tg0WZxoSSdG+DHr9F0zSBg==	11:15 PM	shah65@uwinds...
3	Jainam Shah	wSVFNSF2KbRHSgWj59VJkl3VDYs40T...	11:15 PM	jainamshh28.js...
4	Rushang Shah	hUaHM2Vb3mlzEZdtSlgyLSU5d5lwMX...	11:16 PM	shah65@uwinds...
NULL	NULL	NULL	NULL	NULL

Connection Ready | (LocalDB)\MSSQLLocalDB | LAPTOP-U5IF7ATH\jaina | C:\USERS\JAINA\ONEDRIV...

Show output from: Debug

```
The thread 0x478c has exited with code 0 (0x0).
The thread 0x3cb4 has exited with code 0 (0x0).
The thread 0x47b8 has exited with code 0 (0x0).
The thread 0x4018 has exited with code 0 (0x0).
The thread 0x30dc has exited with code 0 (0x0).
The program '[13552] iisexpress.exe' has exited with code -1 (0xffffffff).
```

4 Rows | Cell is Read Only

VS Project: 1) Database and Data Tables for handling users and chat. 2) Encrypted chat history in database. 3) Project solution files (code and design).

VII. PROS & CONS OF THIS SYSTEM

Pros:

- This system provides ultimate secrecy of data by encrypting the messages using AES algorithm every time the participants push the message text into the database.

- Passcode verification so that no other than participating members can attend the room chat.

- Automatically deletes the chat history after all the participants leave the chat so no trace of the chat is left anywhere.

- Email verification so that only attending users can have to access to room. Also, the room-owner decides who can access to rooms so that chat is in the maximum control of the chat room creator and to the limited to the confidential people only.

- All the authentication functionalities in the system together enable the system to enrol into the MFA category.

Cons:

- Here, we have used the pre-shared key in the AES based encryption. By chance if the intruder or hacker gain the key then all the messages can be decrypted but which is very crucial to gain the access to the key since it is shared on the secret channel.

- The users who get access to the chatrooms may be unfaithful to the room owners. There is still risk of getting the information to join the room by reading the mails of users containing the confidential information of the room.

VIII. FUTURE IMPLEMENTATION

In the future, we can implement this system with RSA which is asymmetric encryption technique. RSA is generally implemented with the digital signatures. Using this way, we can verify the signature so we can confirm that the users are getting the accurate and same message from the same participant. Also, we can implement the multiple encryption techniques by which we can make the system even more stronger.

In addition to that we can replace the email verification by mobile SMS verification. By doing this, it will ensure that the user we are intending to join the session is only getting the message and which eliminates the risk of emails getting read by the intruder. Also, we can implement the files sending, video and voice session which includes the one-to-one encryption. This system can be enhanced by increasing the number of capacity of users. Also, we can add provide the users to download the chat history offline feature.

IX. CONCLUSION

In the conclusion we can say that we have tried to achieve as much as stronger security for chat system application by implanting the concepts to cyber security and encryption algorithm. If any user tries to intrude any chat conversation, then the intruder will get the cipher text which is only be decrypted by the 128-bit key again, which is next to impossible to guess. In addition to the AES cryptography, we have also used the passcode and email verification. So, this adds the multi factor authentication making the system even more stronger. No systems are totally secured but the level to security of is important which makes system worth trustable.

REFERENCES

[1.] Blackboard Material for the AES - *Federal Information Processing standards Publication 197 November 26, 2001 Announcing the ADVANCED ENCRYPTION STANDARD (AES).*

[2.] Symmetric Key Encryption - why, where and how it's used in banking *by Peter Smirnoff & Dawn M. Turner (guests) on 18. January 2019. Available at: https://www.cryptomathic.com/news-events/blog/symmetric-key-encryption-why-where-and-how-its-used-in-banking*

[3.] Encryption and Decryption Using A Symmetric Key In C# *by Vivek Kumar March 11, 2019. Available at : https://www.c-sharpcorner.com/article/encryption-and-decryption-using-a-symmetric-key-in-c-sharp/*

[4.] Wikipedia - *https://en.wikipedia.org/wiki/Symmetric-key_algorithm*