**A Mini Project Report**

**on**

**“Message- Encoder and Decoder”**

Submitted to the

Savitribai Phule Pune University

In partial fulfillment for the award of the Degree of

Bachelor of Engineering

in

**Information Technology**

by

**Viraj Adhav [UIT20M1001]**

**Saurabh Hon [UIT20M1027]**

**Dipjyot Jape [UIT20M1031]**

**Kiran Lungade [UIT20M1040]**

**S.Y. B. Tech Information Technology**

Under the guidance of

**Prof. D. P. Bhamare**



**Department of Information Technology**

**SRES’s Sanjivani College of Engineering, Kopargaon 423603**

**2021-2022**

**SRES’s Sanjivani College of Engineering, Kopargaon 423603**

**Department of Information Technology**



**CERTIFICATE**

This is to certify that the Mini Project report entitled **“Message- Encoder and Decoder”** being submitted by- **Adhav Viraj Vikram (UIT20M1001)**

**Hon Saurabh Kalyan (UIT20M1027)**

**Jape Dipjyot Dilip (UIT20M1031)**

**Lungade Kiran Sanjay (UIT20M1040)**

is a record of bonafide work carried out by them under the supervision and guidance of **Prof. D. P. Bhamare** in partial fulfillment of the requirement for **S.Y (Information Technology)** of Savitribai Phule Pune University, Pune in the academic year 2021-2022

Date: /06/2022

Place: Kopargaon

**Prof. D. P. Bhamare Dr. M. A. Jawale Dr. A. G. Thakur**

Guide H. O. D Principal

This Mini Project report has been examined by us as per the Savitribai Phule Pune University, Pune requirements at **Sanjivani College of Engineering, Kopargaon-423603** on ……………......

Internal Examiner External Examiner

I

**ACKNOWLEDGEMENT**

The entire session of Mini- Project completion phase so far was a great experience providing us with great insight and innovation into learning various data security concepts and achievement of it. As is rightly said, for the successful completion of any work, people are the most important asset our project would not be materialized without the cooperation of the people involved.

First and foremost, we are very thankful to our respected project guide Prof. D. P. Bhamare for their leading guidance and sincere efforts in finalizing this topic. They took deep interest in correcting the minor mistakes and guided us through our journey so far. Also, they have been persistent source of inspiration for us.

We are also very thankful of Dr. M. A. Jawale, Head of Department of Information Technology for the symmetric guidance and providing necessary facilities and we express deep gratitude to all the faculty members and our department’s technical Staff for providing us needed help.

Adhav Viraj Vikram (RN- 01)

Hon Saurabh Kalyan (RN- 26)

Jape Dipjyot Dilip (RN- 30)

Lungade Kiran Sanjay (RN- 39)

II

**Abstract**

Secret, private, and confidential message usually sent by embedding them in a covering media such as digital color image. The principal attraction to project is that it offers equal security for a smaller bit, size, thereby reducing processing overhead. In simpler terms, encoding/decoding is the translation of a message that is easily understood. When you decode a message, you extract the meaning of that message in ways that make sense to you. Decoding has both verbal and non- verbal forms of communication: Decoding behavior without using words means observing body language and its associated emotions. The encoding of the message is the production of the message. It is a system of coded meanings, and to create that, the sender needs to understand how the word is comprehensible to the members of the audience.

Base64 encoding is the process of converting binary data to an ASCII string format by converting that binary data into 6-bit character representations. The Base64 method of encoding is used when binary data, such as image or video needs to be transmitted over systems that are designed only to transmit data in a plain text (ASCII) format. Web developers use base64 formats to include images, fonts, sounds and other resources directly inside files. Due to wide use of base64 encoding a decoder software is required.

III

**Contents**

Acknowledgement I

Abstract II

Abbreviations III

List of Figures IV

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | **Sr.** | | **Chapter** | **Page No** | | **1.** |  | **Introduction to Text Based Input** |  | |  | 1.1 | Introduction to Message- Encoder and Decoder ………...……. | 07 | |  | 1.2 | Motivation behind Message- Encoder and Decoder.………….. | 07 | |  | 1.3 | Aim and Objective(s) of the work ……….……………………. | 07 | | **2.** |  | **Algorithm** …………………………………………………….. | 08 | | **3.** |  | **Library Files** …………………………………………………. | 09 | | **4** |  | **Project File Structure and Functions** |  | |  | 4.1 | Import Libraries ………………………………………………. | 10 | |  | 4.2 | Initialize Window ……………………………………………... | 10 | |  | 4.3 | Define Variables ………………………………………………. | 10 | |  | 4.4 | Function to encode ……………………………………………. | 10 | |  | 4.5 | Function to decode ……………………………………………. | 11 | |  | 4.6 | Function to set mode ………………………………………….. | 11 | |  | 4.7 | Function to exit window ………………………………………. | 11 | | **5.** |  | **Result Analysis** ……………………………………………….. | 12 | | **6.** |  | **Advantages and uses** ………………………………………… | 14 | | **7.** |  | **Future Scope and Conclusion** ……………………………….. | 15 | |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| IV |  |  |  |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Name** | **Page No.** |
| 1. | Encoding…………………………………………. | 12 |
| 2. | Decoding…………………………………………. | 13 |
| 3. | Reset……………………………………………… | 13 |

**CHAPTER 1**

**INTRODUCTION TO Message- Encoder and Decoder**

* 1. **Introduction to Project**

Python is a high\_ level, interpreted, interactive and object-oriented scripted language. It is designed to be highly readable. It provides various options for developing graphical interfaces (GUI’s).

This is python project which help to encode and decode of message which created using python and tkinter, base64 library. Which is part of Message encoding and decoding is the process to first convert the original text to the random and meaningless text called ciphertext. This process is called encoding. Decoding is the process to convert that ciphertext to the original text. This process is called the Encryption\_ Decryption process.

* 1. **Motivation behind project**

Encoding and Decoding message is the heart of communication. Humans encode thoughts using language. One way of looking at programming is to see it as a way of communicating human instructions to computers, which then carry out the tasks we want to perform.

* 1. **Aim and Objective(s) of the work**

**Project aims**

The aim of the project is to transform data so that it can be properly and safely consumed by a different type of the system. The goal is not to keep information secret, but rather to ensure that it’s able to be properly consumed.

**Project objectives**

* The project is to encode and decode message using a common key.
* This project will be built using the Tkinter and base64 library.
* In this project, users must enter the message to encode or decode.
* Users must select the mode to choose the encoding and decoding process, the same key must be used to process the encoding and decoding for the same message.
* To ensure integrity and usability of data and is commonly used when data cannot be transferred in its current format between systems or applications.

**CHAPTER 2**

**Algorithm**

1. Hashing- Maps data of arbitrary size to a hash of fixed size.
2. Lossless- Used for compressing text/video/audio, creating zip file.
3. Huffman- A variable length code is assigned to input different characters.
4. Lempel-Ziv-Welch (LZW)- is a popular compression algorithm used by number of formats, including GIF, TIFF, PDF, Unix Compress, PostScript.
5. Discrete Cosine Transform (DCT)- express a finite sequence of data points in terms of sum of cosine functions oscillating at different frequencies.

**CHAPTER 3**

**Library File**

To build this project we will use basic concept of python, Tkinter and base64 library.

* **Tkinter:** It is a standard GUI python library used to create a display window.
* **base64:** base64 module provides a function to encode the binary data to ASCII characters and decode that ASCII characters back to binary data.

To install the library, we use pip install command on the command prompt

*pip install tkinter*

*pip install base64*

These are the step to build message encode- decode python project

* Import module
* Create display window
* Define function
* Define labels and buttons

**CHAPTER 4**

**Project File Structure and Functions**

**4.1. Import Libraries-**

*From tkinter import \**

*Import base64*

The first step is to import tkinter and base64 libraries.

4.2. **Initialize Window-**

* **Tk ()–** initialized tkinter which means window created.
* **geometry ()-** set the width and height of the window.
* **resizable (0,0)-** set the fixed size of the window
* **title ()-** set the title of the window

**Label ()-** widget use to display one or more than one line of the text than users aren’t able to modify.

* **root** is the name which we refer to our window
* **text** which we display on the label
* **font** in which the text is written
* **pack** organized widget in block

**4.3. Define variables-**

* **Text** variable stores the message to encode or decode
* **private\_key** variable stores the private key used to encode and decode
* **mode** is used to select that is either encoding or decoding
* **Result** store the result

**4.4. Function to encode-**

* **Enc** = [] is a empty list, we run the loop till the length of the message
* **i% len(key)** gives the remainder of division between i and len(key) and that remainder used as an index of key the value of key at that index is stored in **key\_c**
* **ord ()** function takes string argument of a single unicode character and return its integer unicode value
* **chr ()** function takes an integer argument and returns the string.
* **ord (message[i])** convert the value of message at index i into the integer value
* **ord(key\_c)** converts the key\_c value to integer value
* **ord(message[i]) + ord(key\_c)) % 256** gives the remainder of division of addition of ord(message[i]) and ord(key\_c) with 256 and passes that remainder to chr () function
* **chr ()** function converts that integer value to string and store to enc
* **base64.urlsafe\_b64encode** encode a string.
* **join ()** method joins each element of list, string, and tuple by a string separator and returns the concatenated string.
* **encode ()** method returns utf-8 encoded message of the string.
* **decode ()** method decodes the string.
* **return** gives the result of the encoded string.

**4.5. Function to decode-**

* **dec= []** is an empty list
* Decode the content from input and write the result in binary to the output
* We ran the loop till the length of the message
* **256 + ord(message[i]) – ord(key\_c)) % 256** gives the remainder of addition of 256 with subtraction of ord(message[i]) – ord(key\_c) and then division with 256 and passes that remainder to chr () function
* **chr ()** function convert integer value to string and store to dec
* **return “”. join(dec)** return the result

**4.6. Function to set mode-**

* If the mode set by the user is ‘**e**’ then the **Encode ()** function will be called
* If the mode set to ‘**d**’ then the **Decode ()** function will be called
* Else print ‘**invalid mode**’
* **private\_key.get ()** and **Text.get ()** values are pass to the arguments of Encode () and Decode () function

**4.7. Function to exit window-**

* **root. destroy ()** will quit the program by stopping the mainloop

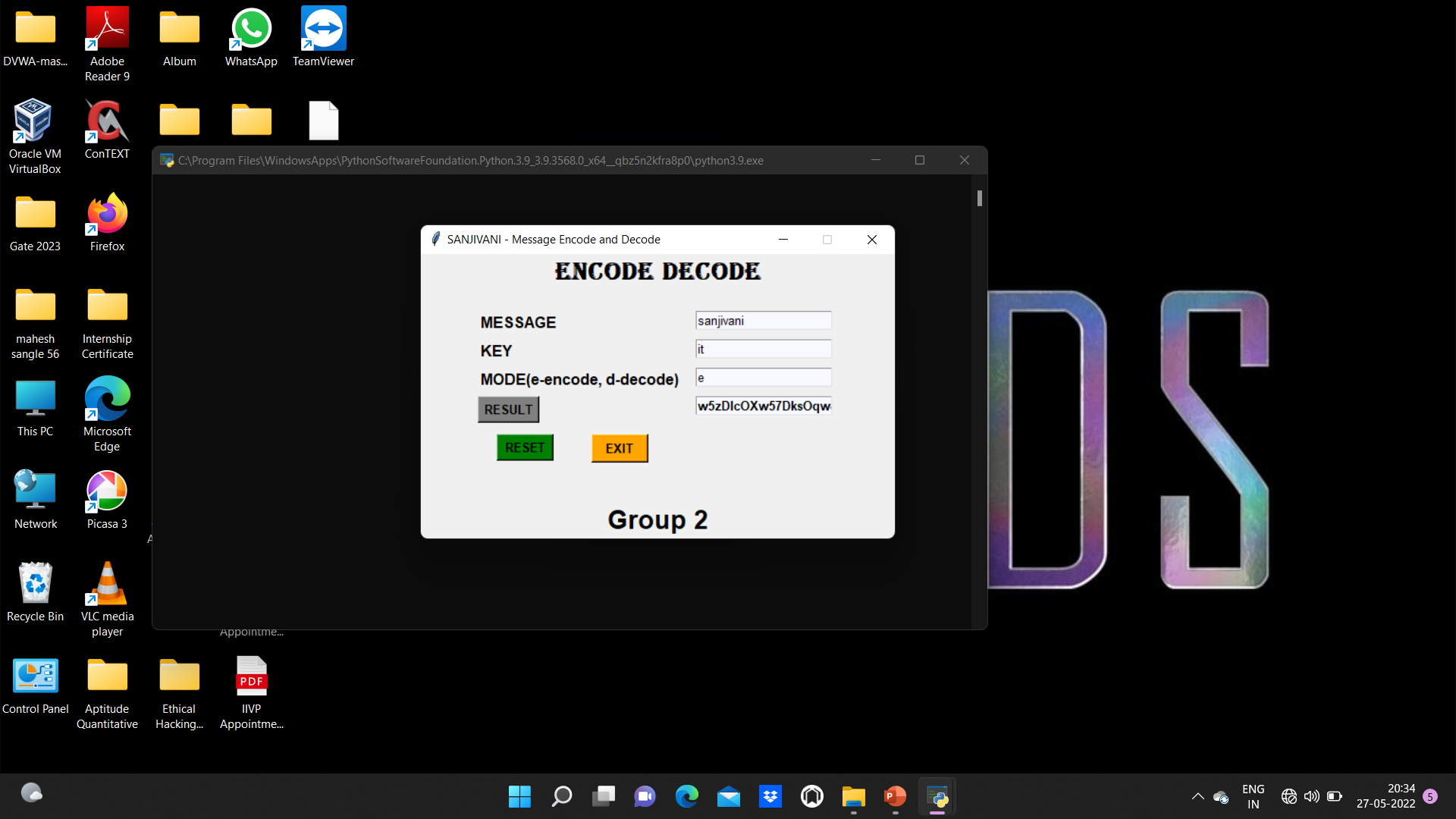
**CHAPTER 5**

**Result Analysis**

* 1. **Encode**
* Open terminal, enter the command

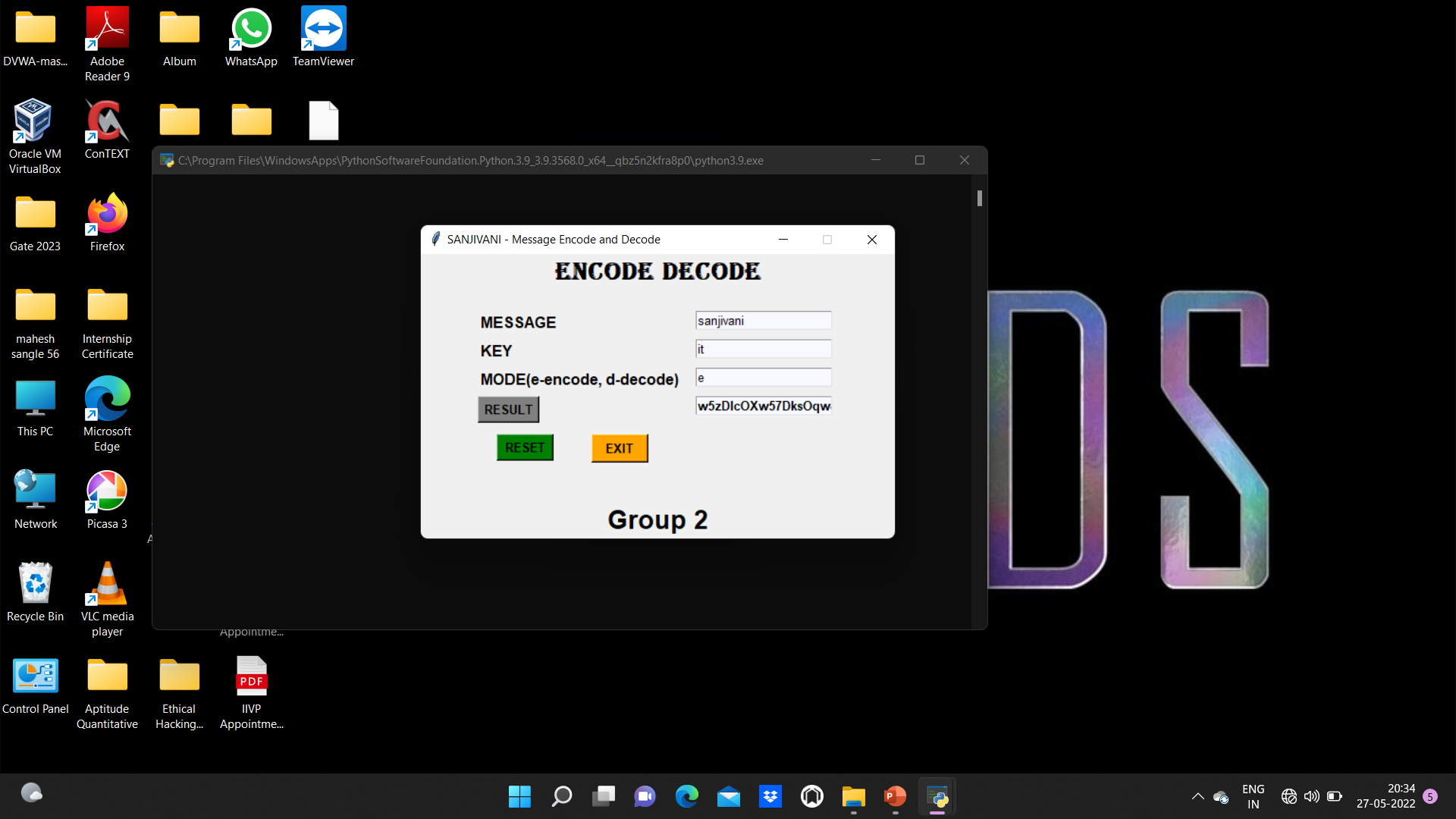
*python encoder.py*

* Empty window will be opened
* Then enter the message, key
* While doing encoding, type mode as ‘e’.
* Click on Result button to get encoded output
* For example:

****

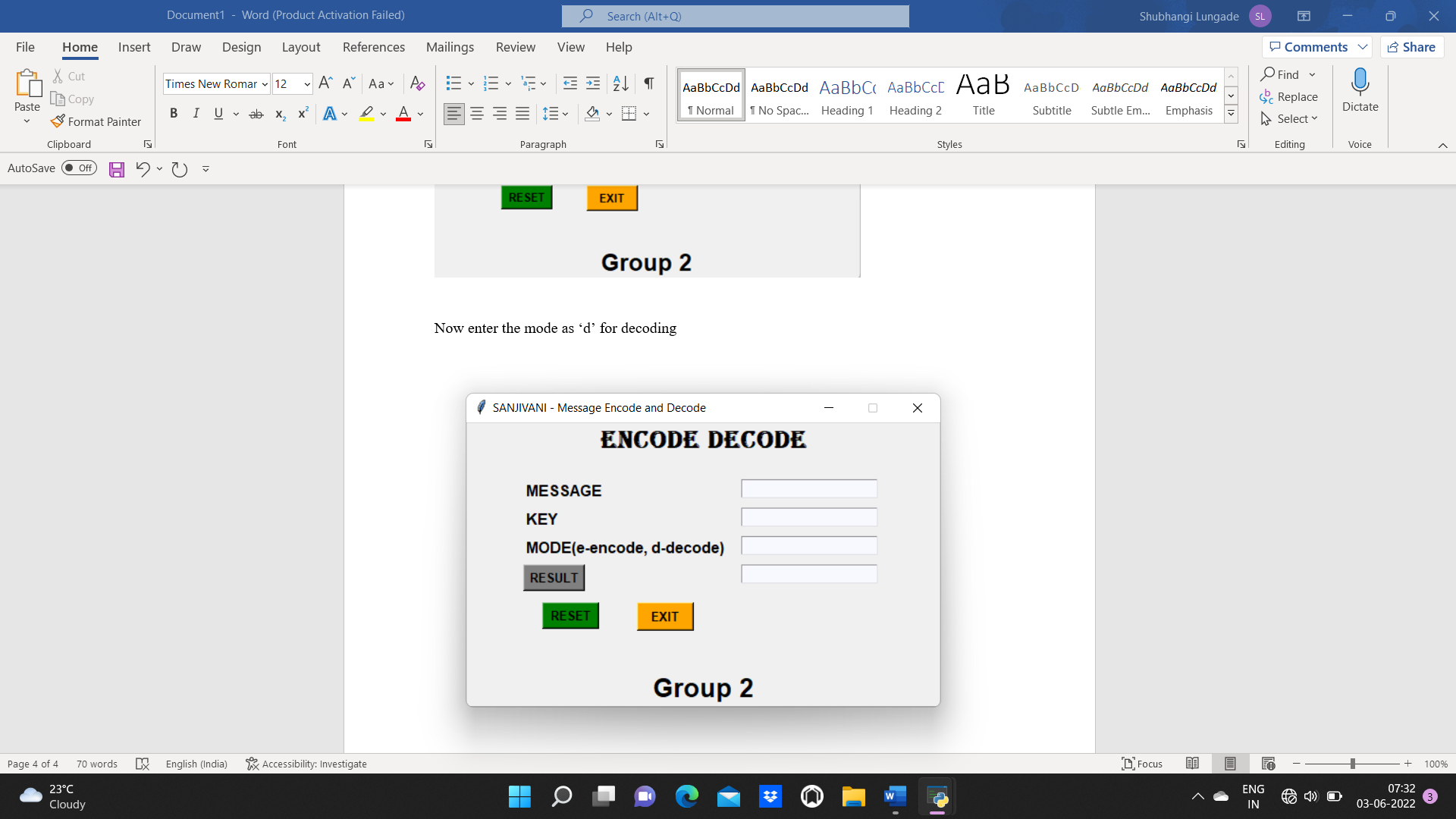
**Figure 1: Encoding**

* 1. **Decode**
* Copy the encoded result, and paste in the message as an input, and keep the key same
* Now enter the mode as ‘d’ for decoding
* Again, click on the Result to get the decoded output
* For example,

****

**Figure 2: Decoding**

* 1. **Reset**
* Click on reset button for resetting the window



**Figure 3: Reset**

**CHAPTER 6**

**Advantages**

* Highly reliable and accurate
* Higher resolution
* Incorporated into existing applications
* Low-cost feedback
* Integrated electronics
* Fuses optical and digital technology
* Security of message

**Uses**

* Cyber Security
* Online shopping
* Online payments
* Communication
* ATM
* Data exchange
* Send Secure E-mail
* Data storage (like- photo, video, files, etc...)

**CHAPTER 7**

**Future Scope**

* The tool can be made more user friendly by adding more buttons like clear, color, etc.
* Other encode decode feature can be added like zip/ unzip and other things.
* Python other platforms can be used for increasing speed of decode.
* The tool can be developed to run another platform like Linux and Mac os.

**CONCLUSION**

We have successfully developed Message encode – decode project in Python. We used the popular tkinter library for rendering graphics on a display window and base64 to encode & decode. We learned how to encode and decode the string, how to create button, widget, and pass the function to the button. In this way, we can encode our message and decode the encoded message in a secure way by using the key.