

Programación de Computadores

Pycrypt

Presentado por:

Nicolás Osorio Guarín

nosoriog@unal.edu.co

Pablo Jose Florez Gomez

pafllorezg@unal.edu.co

Sergio Andrés Gordillo Gomez

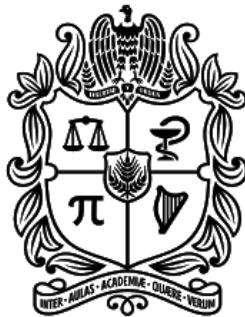
sgordillo@unal.edu.co

Profesora:

Stephanie Torres Jimenez

sttorresji@unal.edu.co

Fecha: 11/12/2020



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Facultad de Ingeniería Electrónica

2020 - II

Tabla de Contenido

Tabla de Contenido	2
INTRODUCCIÓN	3
CAMPO DE ACCIÓN	3
DEFINICIÓN GENERAL	4
OBJETIVOS	4
Objetivo General	4
Objetivos Específicos	4
RESULTADO ESPERADO	4
DATOS EXTRAÍDOS DE LA WEB	5
RESULTADO OBTENIDO TOTAL	5
DEMOSTRACIÓN DEL PROYECTO	5
Opción de registrarse	7
métodos de encriptación	8
TRABAJO A FUTURO	10
REFERENCIAS BIBLIOGRÁFICAS	10
CONCLUSIONES	11

INTRODUCCIÓN

La encriptación es un proceso que consiste en transformar mensajes de tal forma que se vuelven incomprensibles para otras personas, con el fin de garantizar la confidencialidad de estos mensajes, y a su vez, permitir su vinculación con otras personas por medio de una clave de desencriptación. Su funcionamiento consiste en una serie de algoritmos que procesan el mensaje, lo encriptan volviéndolo indescifrable y generan una clave dada por el usuario, que le permitirá a la persona que la posea desencriptar el mensaje para que en el futuro pueda leerlo.

En este proyecto se crearán diversos algoritmos de encriptación que le permitan al usuario ingresar un mensaje para encriptarlo y posteriormente que el usuario cree una clave de desencriptación, con la cual, pueda acceder a ese mensaje en el futuro, todo este proceso se realizará mediante una interfaz de usuario, es decir, el menú y todos los elementos con los que interactúa el usuario que le permitirá interactuar con los diferentes métodos de encriptación creados por los algoritmos. Del mismo modo, en el proceso de creación de los algoritmos se utilizarán diferentes técnicas de encriptación que se explicarán más adelante.

La principal problemática a resolver en este proyecto es crear y usar algoritmos que generen mensajes encriptados difíciles de descifrar, y que la interacción del usuario con el programa y la interfaz de usuario sea fácil y sencilla de entender y que le brinde al usuario diversas opciones de encriptación.

CAMPO DE ACCIÓN

La encriptación es utilizada principalmente en algunos tipos de datos, tales como:

- Información de identificación personal: Cualquier tipo de información que se use para identificar a una persona de manera única tales como: Números de Cédulas, Licencias de conducción entre otros.
 - Información empresarial: En las empresas es de vital importancia mantener segura información como: Los datos de los clientes de la empresa, Planes y propuestas de negocios, Campañas y estrategias de marketing, etc..
- y toda la información o datos que a nuestro concepto se tengan que mantener seguros.
- Empresas como :
- Google: con servicios como (AES), (TLS), (S/MIME)
 - Amazon: con servicios como (SSL), (TLS)
 - Facebook: con servicios como (SSL)
 - Dropbox : con servicios como (CAS)

Ofrecen servicios de encriptación, en base a los diversos métodos que tiene cada empresa para encriptar la información.

DEFINICIÓN GENERAL

El proyecto usará distintos algoritmos de encriptación que le permita al usuario encriptar algún mensaje que requiere de una mayor seguridad durante el viaje a su destinatario. También, se busca la personalización del programa y de los mensajes del usuario mediante algoritmos propios que le permitan una encriptación más segura y única en sus mensajes.

En base a esto también se busca crear una interfaz gráfica agradable visualmente para el usuario y que sea fácil de usar. Con el uso de librerías como: (tkinter, wxpython , etc.).En la que el usuario pueda ver los diferentes métodos de encriptación y elegir el de su preferencia.

OBJETIVOS

Objetivo General

Crear métodos y algoritmos que encripten de diversas formas la información suministrada por el usuario. Con el fin de brindarle una mayor seguridad a sus datos y permitiendo llegar a su destinatario de manera segura y confidencial. Por medio de un menú de selección de algoritmos de encriptación ofrecido al usuario.

Objetivos Específicos

- Explicar cómo funcionan algunos algoritmos de encriptación
- Sugerir algunos algoritmos propios de encriptación al usuario
- Establecer un menú visualmente atractivo de algoritmos que le permitan al usuario encriptar su información de diferentes maneras para obtener mayor seguridad.
- Demostrar la fiabilidad de los algoritmos que se usarán para efectuar la encriptación
- Registrar de manera organizada, algunos algoritmos de encriptación que se le ofrece al usuario.

RESULTADO ESPERADO

El resultado que se espera con este proyecto consiste en tener un programa que le pueda brindar al usuario diferentes métodos de encriptación; y que a su vez garantice la confidencialidad del mensaje y la opción de introducir una clave de descifrado para poder visualizar el mensaje, otra de las características que debe incluir el programa, será la opción de compartir el mensaje cifrado con otra persona y que esta lo pueda abrir con la clave seleccionada. Finalmente la interfaz del usuario, es decir, el menú y todos los elementos con los que el usuario interactúa, deben ser fáciles, sencillos de entender y manejar. A su vez, se integren de manera eficaz con los algoritmos de encriptación que se le brinda al usuario.

DATOS EXTRAÍDOS DE LA WEB

De la web utilizamos los datos acerca de que son 4 de los algoritmos usados en nuestro proyecto. Es decir, en la página web de Wikipedia, extrajimos información pertinente a las definiciones de los algoritmos Cesar, Vigenere, Vernam y Solitario. Estos datos, se extrajeron mediante el uso del web scraping; este proceso se lleva a cabo con la librería BeautifulSoup para obtener los datos relevantes para nuestro proyecto. Estos datos relevantes, se pueden usar como información encriptable en el proyecto, accediendo a ella por medio de un botón de información.

RESULTADO OBTENIDO TOTAL

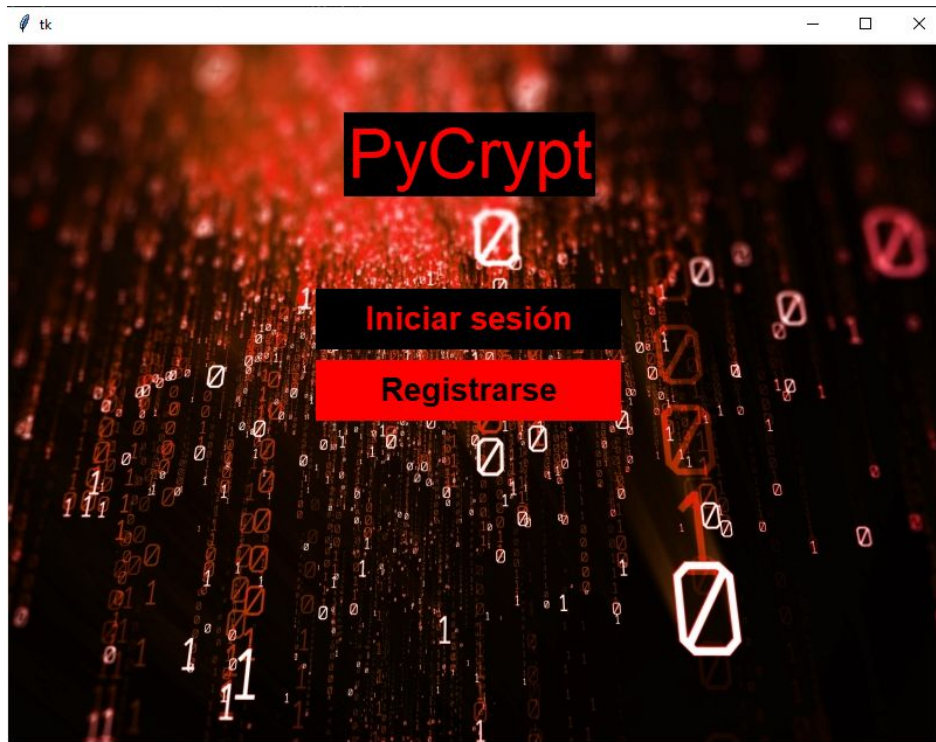
Después de haber realizado en totalidad el proyecto de PyCrypt, podemos evidenciar que el resultado obtenido del proyecto cumplen con las mejoras esperadas después, de la primera entrega, es decir, cuenta con un módulo de registro, el uso de archivos para guardar mensajes, la posibilidad encriptar imágenes y un nuevo método de encriptación, además, estéticamente también llegamos al resultado esperado, ya que, utilizamos estilos para los fondos de las ventanas y para los botones de el login y el menú. En comparación con la versión anterior que no tenía un login, esta versión utiliza una base de datos sqlite para el registro de usuarios y su respectiva contraseña, y para el inicio de sesión de los usuarios registrados, asimismo, al programa de le adicionó el cifrado RSA, el cual, a diferencia de los métodos anteriores es un cifrado de tipo asimétrico, es decir, cuenta con dos claves en vez de una, finalmente cuenta con algoritmo de encriptación de imágenes, el cual, le permite al usuario esconder una imagen dentro de otra con la posibilidad de guardarla, y al mismo tiempo tiene la opción de extraer la imagen escondida y guardarla en la ruta que el usuario desee.

DEMOSTRACIÓN DEL PROYECTO

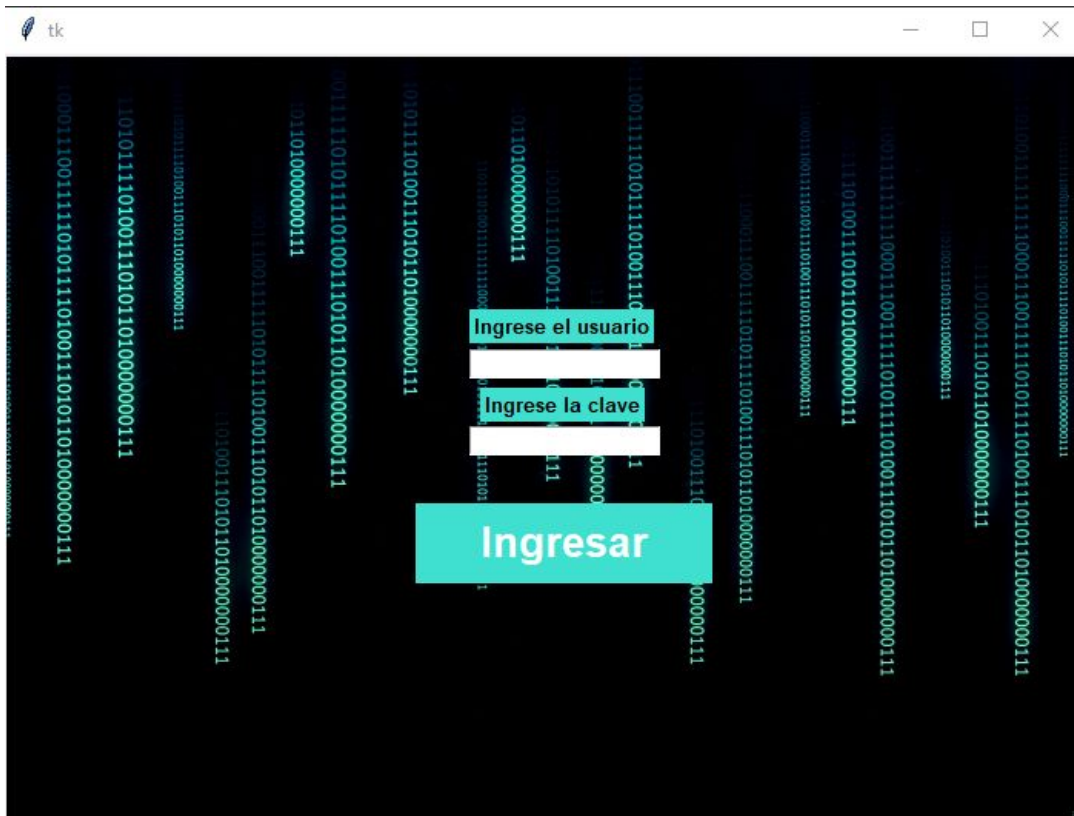
En primer lugar el programa cuenta con un módulo de registro en el que el usuario tiene la opción de iniciar sesión o registrarse en la base de datos creada en sqlite3, una vez iniciada la sesión se abrirá la ventana del menú de métodos donde se encontrar los botones que dirigirán al usuario a cada uno de los métodos, asimismo, esta ventana tiene dos botones adicionales, el botón de información en el que por medio de web scrapping se muestra en la ventana información pertinente de algunos métodos, el segundo botón permite al usuario visualizar el historial de mensajes que ha utilizado en los diferentes métodos, como también la hora en que realizó la encriptación, en cuanto a los algoritmos de encriptación, el usuario tiene la posibilidad de ingresar un mensaje más la clave con la que lo va encriptar o desencriptar, si este mensaje es mayor a 40 caracteres el mensaje encriptado o desencriptado, se guardará en una archivo de texto donde posteriormente puede visualizar el mensaje, su clave, el encriptado y la hora en la que utilizó el método. Finalmente para el algoritmo de encriptación de imágenes, el usuario tiene la posibilidad de utilizar dos imágenes formato png y de dimensiones 300x300, una de las imágenes se esconderá dentro de la otra y la imagen resultante se guardará en la ruta indicada por el usuario,

asimismo, el proceso de extracción de la imagen le permite al usuario abrir una imagen desde sus archivos extraer la imagen que estaba escondida, para posteriormente guardarla.

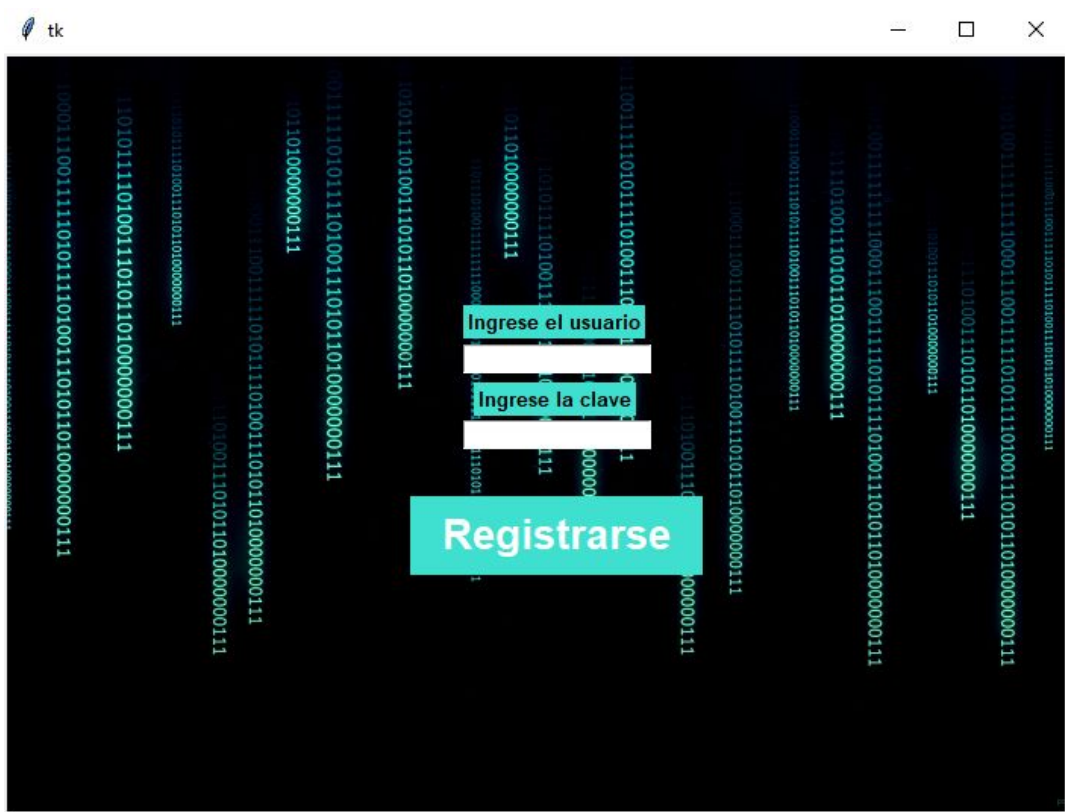
Módulo de registro



Opción de iniciar sesión



Opción de registrarse



A registration window titled "tk" with a black background featuring a green binary code pattern. In the center, there is a registration form with a light blue border. The form contains two input fields labeled "Ingrese el usuario" and "Ingrese la clave", both with white text. Below these fields is a large blue button with the text "Registrarse" in white. The window has standard macOS-style window controls (red, yellow, and green buttons) in the top-left corner.

Ventana del menú



A menu window titled "tk" with a black background featuring a green binary code pattern. The window displays the title "PyCrypt" in large red letters at the top. Below the title, the text "Escoja el método de encriptación" is written in red. A vertical stack of seven red buttons with white text lists the encryption methods: "Metodo Cesar", "Metodo Solitario", "Metodo P-I", "Metodo Vigenere", "Metodo Vernan", "Encriptar imagenes", and "Metodo RSA". At the bottom left, there is a blue button labeled "Info", and at the bottom right, there is a blue button labeled "Historial". The window has standard macOS-style window controls (red, yellow, and green buttons) in the top-left corner.

A continuación, algunas de las ventanas de los métodos de encriptación

Método Vigenere



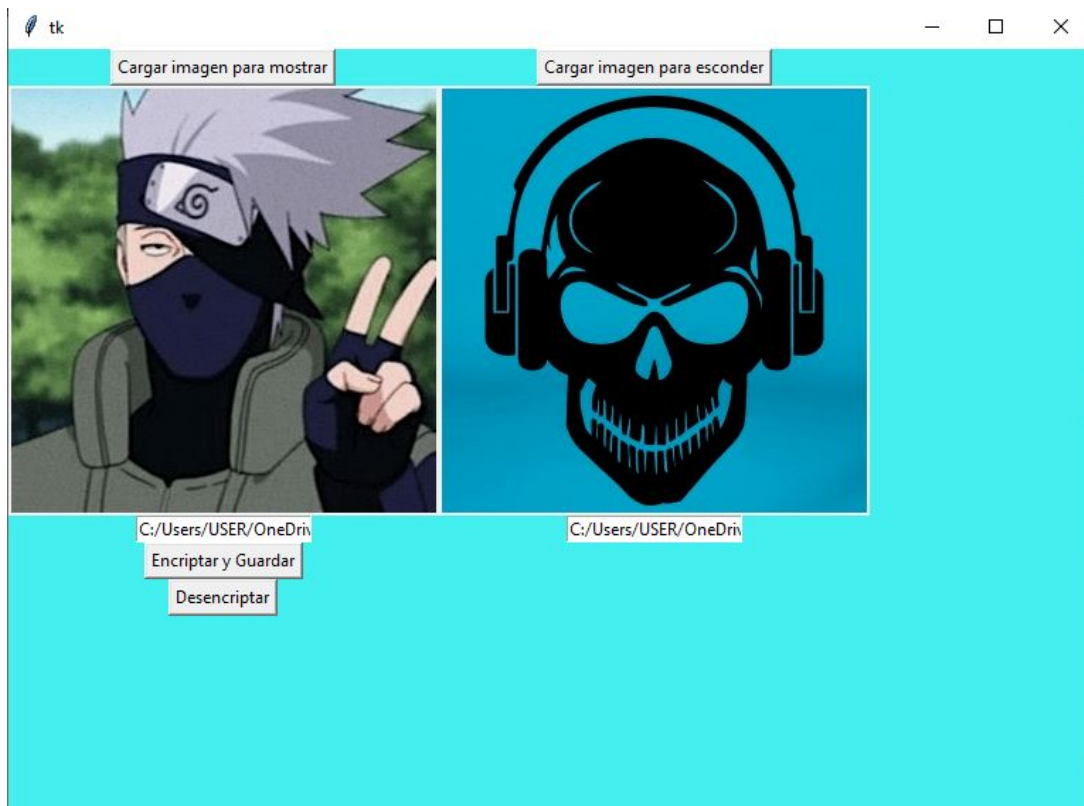
The screenshot shows a window titled "Vigener" with a red background. It contains two sections for encryption and decryption. The top section has labels "Ingrese el mensaje:" and "Ingrese la clave:" followed by input fields and an "Encriptar" button. The bottom section has labels "El mensaje es:", "Ingrese el mensaje encriptado:", and "Ingrese la clave:" followed by input fields and a "Desencriptar" button.

Método RSA

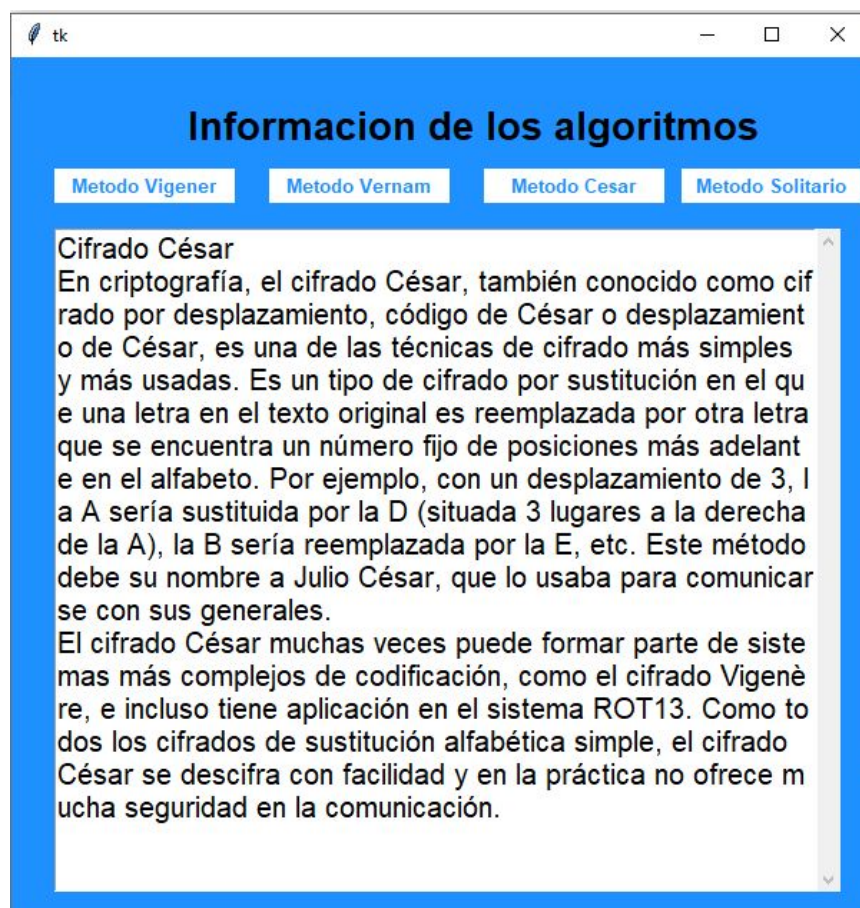


The screenshot shows a window titled "tk" with a blue background. It features a "Generar Claves" button at the top left. Below it, a text area displays "Clave Pública: 3,447" and "Clave Privada: 99,447". To the right, there are labels "Ingrese el mensaje" and "Ingrese la clave" with corresponding input fields. At the bottom right, there are "Encriptar" and "Desencriptar" buttons. A large white rectangular area is visible at the bottom of the window.

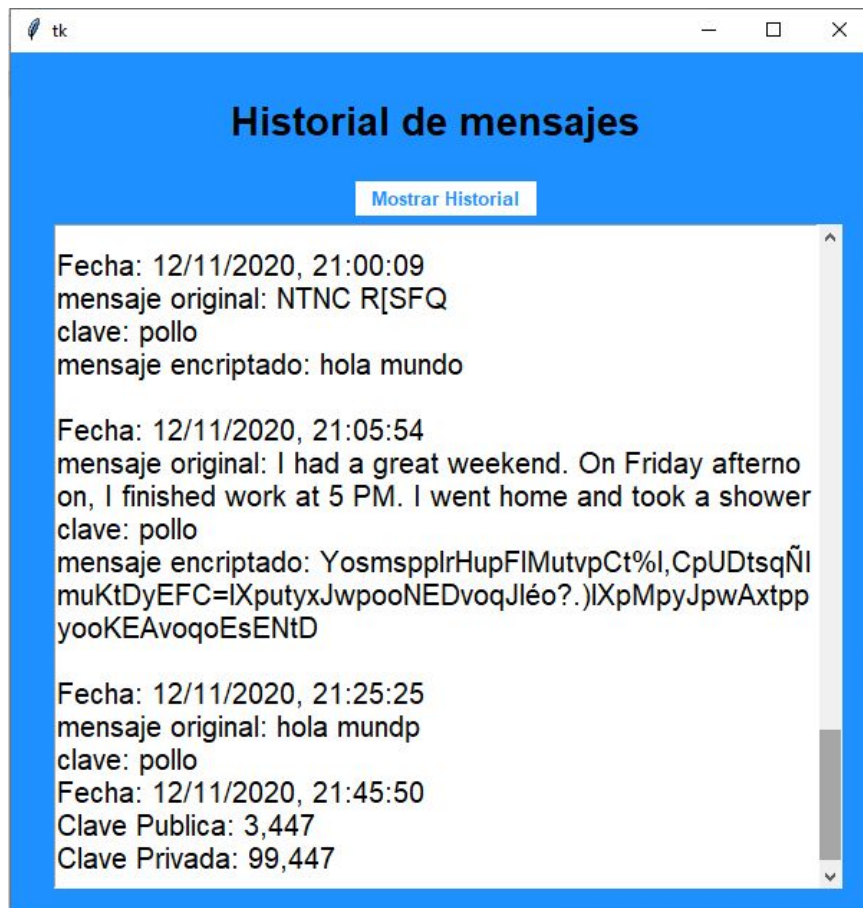
Método Imágenes



Botón de información



Botón Historial



TRABAJO A FUTURO

Para el trabajo de PyCrypt a futuro se puede realizar la implementación de archivos para uno de los algoritmos que presentan unas excepciones a la hora de crearlos. Este es un algoritmo realizado por nosotros llamado Par - Impar como su nombre lo indica trabajamos modificando los valores en números pares e impares para llegar a la encriptación. Sin embargo, los caracteres que se usan son UNICODE y al momento de generar un archivo con estos salen de manera errónea. Por lo tanto, se busca a futuro que esta salida sea la correcta y se pueda almacenar en los archivos como se hizo con los demás algoritmos de encriptación. Además, en el algoritmo de encriptación de imágenes se planea mejorarlo para optimizar su tiempo de ejecución y también permitirle al usuario ingresar imágenes de diferentes dimensiones, ya que, en esta versión, solo permite imágenes png de 300x300 pixeles. Finalmente, las ventanas de los algoritmos de encriptación, tienen la capacidad de ser mejoradas estéticamente en los estilos de los botones y ventanas.

REFERENCIAS BIBLIOGRÁFICAS

<https://juncotic.com/rsa-como-funciona-este-algoritmo/>
<https://sindominio.net/biblioweb/telematica/solitario.html>
<https://www.ugr.es/~anillos/textos/pdf/2011/EXPO-1.Criptografia/02a11.htm>
https://es.wikipedia.org/wiki/Cifrado_de_Vigenère

https://es.wikipedia.org/wiki/Cifrado_César

http://www.dma.fi.upm.es/recursos/aplicaciones/matematica_discreta/web/aritmetica_modular/cesar.html

https://es.wikipedia.org/wiki/Cifrado_Vernam

[https://es.gaz.wiki/wiki/Solitaire_\(cipher\)](https://es.gaz.wiki/wiki/Solitaire_(cipher))

CONCLUSIONES

La complejidad de un algoritmo de encriptación está dada por los métodos que maneja. Es decir, entre más complejo sea el sistema que utiliza de base para encriptar un mensaje es más seguro. Por ejemplo, un algoritmo que trabaja con secuencias numéricas conocidas es menos seguro que un algoritmo que trabaja con números que no tienen ninguna secuencia como lo es el RSA, estos algoritmos como el RSA se denominan algoritmos asimétricos y su complejidad a la hora de encriptar hace que los vuelva más seguros que otros algoritmos simétricos.

Aprendimos a manejar la cantidad de imprevistos que salen a la hora de programar, ya que muchas veces algo está funcionando bien y se le agrega algo nuevo (en nuestro caso guardar los mensajes en el historial) y empiezan a surgir muchos problemas que a lo largo del semestre pudimos ir arreglando, pero al final queda de lección de tener copias del código cuando sirve , ya sea de manera local o en plataformas en la nube como github entre otras con servicios parecidos.