Un dibujo de un animal con la boca abierta

Descripción generada automáticamente con confianza baja

**UNIVERSIDAD PERUANA DE CIENCIA APLICADAS**

**Trabajo Final**

**Curso:**

Complejidad Algorítmica

**Sección:**

CC41

**Docente:**

Silva Reyes, Patricia

**Integrantes:**

-Matthew David Tello Salas U202021443

-Joseph Ramirez Sarmiento U20211C828

-Luna Morales Gianfranco U201824343

Índice

[**Descripción del Problema**](#_heading=h.3znysh7) **3 pág.**

[**Descripción y Visualización del conjunto de datos(Dataset)**](#_heading=h.2et92p0) **3 pág.**

[**Propuesta**](#_heading=h.tyjcwt) **5 pág.**

[**Diseño del aplicativo**](#_heading=h.1t3h5sf) **6 pág.**

[**Análisis de requisitos**](#_heading=h.4d34og8) **6 pág.**

[**Diseño de sistema**](#_heading=h.2s8eyo1) **7 pág.**

[**Implementación**](#_heading=h.17dp8vu) **10 pág.**

[**Verificación**](#_heading=h.3rdcrjn) **10 pág.**

[**Mantenimiento o puesta en producción**](#_heading=h.26in1rg) **10 pág.**

[**Validación de resultados y pruebas**](#_heading=h.lnxbz9) **11 pág.**

[**Conclusiones**](#_heading=h.35nkun2) **11 pág.**

[**Bibliografía**](#_heading=h.1ksv4uv) **12 pág.**

# Descripción del Problema

Debido al conflicto ucraniano y la actual pandemia de la Covid-19, el precio del combustible se ha visto en crecimiento. Esto dio como resultado el incremento de las tarifas de los pasajes de vuelo, con ello la reducción de la cantidad de vuelos realizados (BBC News Mundo, 2022). Por esta razón, se debe optimizar el uso del combustible en los vuelos, aplicando un algoritmo que se encargue de encontrar las mejores rutas de vuelo.

“Las tarifas de transporte de [crudo](https://www.worldenergytrade.com/component/seoglossary/1-energia/petroleo) desde Rusia se han disparado debido a las sanciones impuestas al país, que aumentan el riesgo de movilizar cargas por esas rutas, mientras que la búsqueda de suministros alternativos aumenta las tarifas de otros trayectos.”  
 -*World Energy trade (marzo, 2022)*

# Descripción y Visualización del conjunto de datos (Dataset)

Se ha utilizado un dataset de OurAiport Open Data para la realización del proyecto. La cantidad de registros de los aeropuertos usados serán 1500 de los 70 000 del dataset original, discriminando aeropuertos diferentes de los ubicados en Estados Unidos y el tipo de aeropuerto. Para generar los datos, se implementó el uso de grafos para simular, con la mayor precisión posible, las rutas de vuelo. De esta manera, se logra un mejor análisis e identificación de posibilidades de simplificación de la ruta.

Finalmente, se redujo los atributos utilizados para del dataset, los cuales serán los siguientes. (El dataset se encuentra en el Anexo 1 y el código en el Anexo 2)

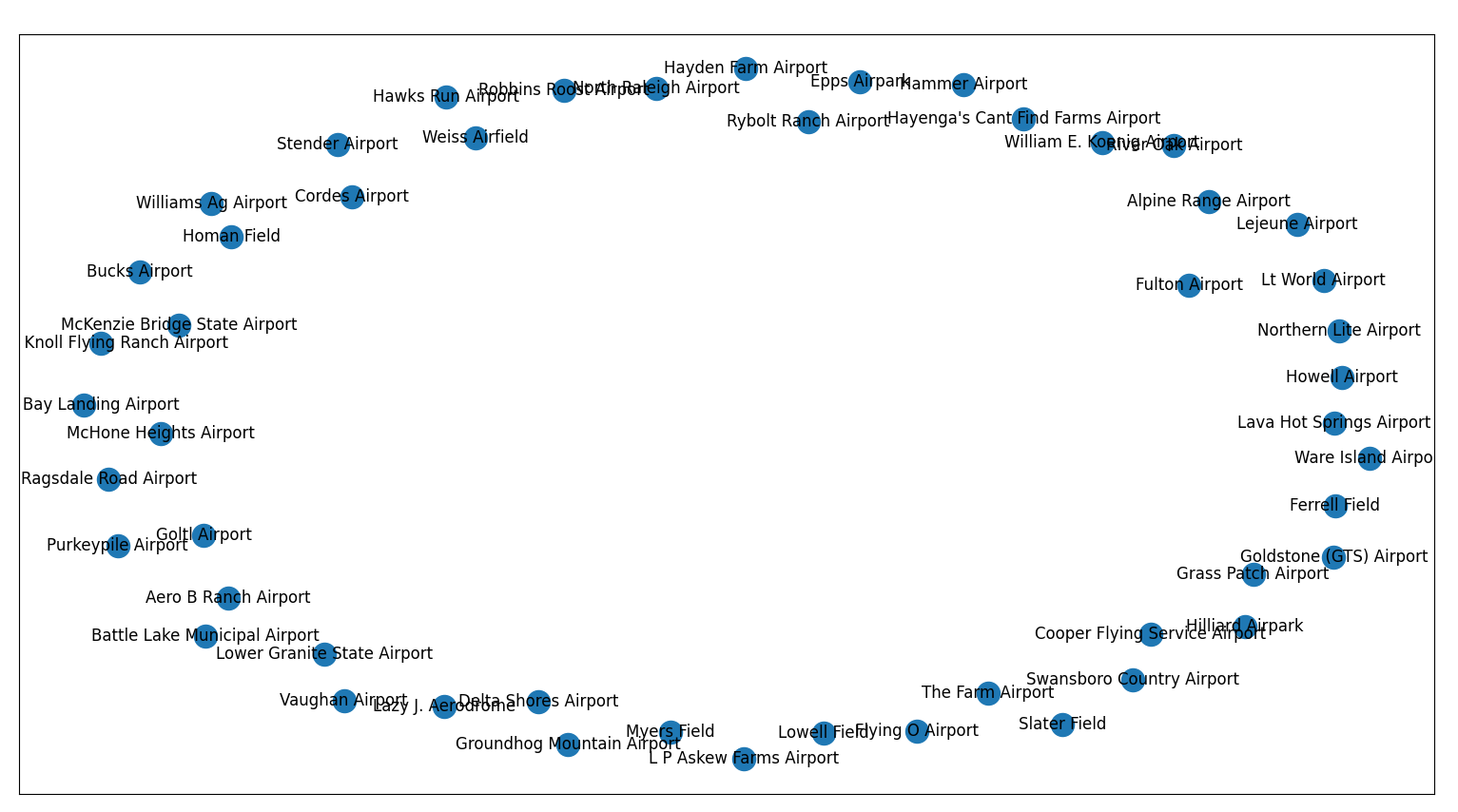
Dentro de la estructura datos grafo se tendrán como valores de los nodos un id de referencia de la clase aeropuerto y para los valores de las aristas se realizará un cálculo para determinar la distancia entre los puntos geodésico de los aeropuertos y a este valor se le multiplicará por un factor de congestión de acuerdo a las horas, además el cambio de tiempo de factor se realizará cada 3h y se actualizará los valores de las aristas del grafo, por último las rutas entre aeropuertos se actualizan cada 8h y se actualizará el grafo.

ATRIBUTOS

| id | identificador único del dataset |
| --- | --- |
| ident | código del aeropuerto |
| name | nombre formal del aeropuerto |
| latitude\_deg | latitud en grados sexagesimal |
| longitude\_deg | longitud en grados sexagesimal |
| elevation\_ft | altitud en unidad de pies |

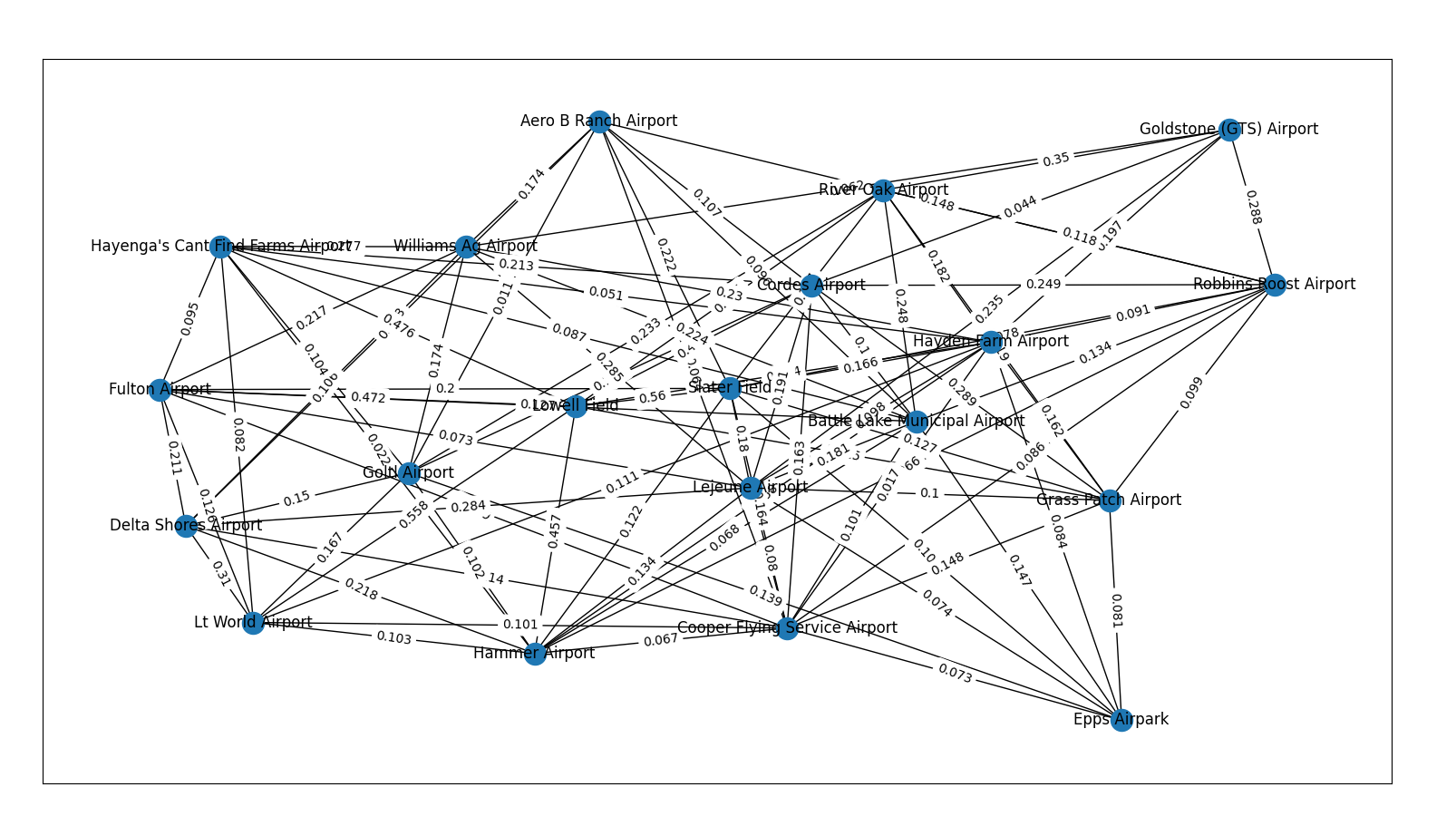
Visualización del conjunto de datos como grafo

Visualización de muestra de unos 50 nodos sin vértices



Fuente propia (ANEXO 3)

Visualización de muestra de unos 20 nodos con vértices



Fuente propia (ANEXO 4)

# Propuesta

Tras la cantidad de datos estudiados, se propone realizar una aplicación que permite trazar rutas tomando en cuenta la mejor ruta de vuelo. El desarrollo de la aplicación se realizará mediante metodología ágil para un desarrollo constante de la aplicación. Asimismo, se hará uso del lenguaje de programación Python y en cuanto la aplicación se hará uso de grafos como estructura de datos para la solución del problema.

Se propone resolver la problemática mediante el algoritmo de A\* para obtener la mejor ruta, además se encontrará la menor distancia recorrida utilizando el algoritmo de floyd-warshall.

# 

# Diseño del aplicativo

## Análisis de requisitos

Los objetivos de proyecto son:

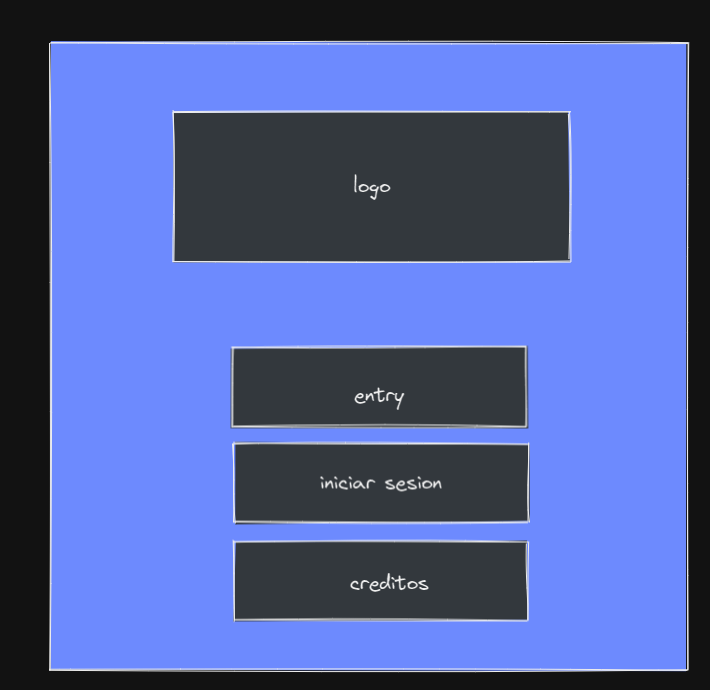
* Nuestro proyecto cumpla con brindar la ruta mínima y obtener la menor distancia recorrida entre dos aeropuertos

Para realizar el proyecto debemos tener en cuenta los siguientes requerimientos:

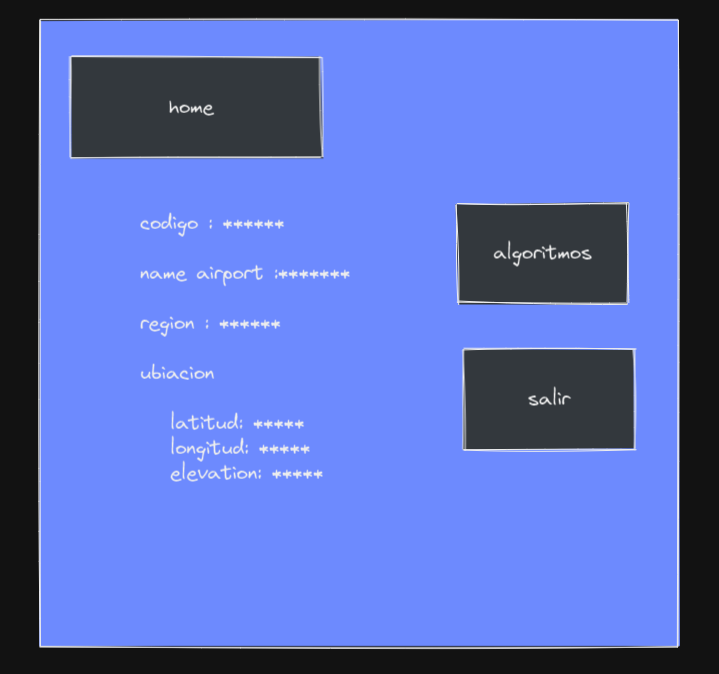
* Investigar los algoritmos floyd-warshall y A\* para su posterior implementación
* Investigar el uso de framework para interfaz gráfica tkinter
* Implementación del algoritmo floyd-warshall
* Implementación del algoritmo A\*
* Actualización de los datos del grafo en tiempo real
* Realizar la interfaz gráfica que cumpla con objetivos de proyecto

## Diseño de sistema

Para el diseño del proyecto en cuanto al fronted se planeó las siguientes visualizaciones:



Esta es la primera página principal del programa, en esta se muestra la portada del proyecto. Los widgets (elementos gráficos, botones, entradas de texto, labels, etc.) en este apartado son un label para insertar el logo del proyecto, una entrada de texto para ingresar el código de aeropuerto, el botón de iniciar sesión para entrar a la segunda pantalla y el botón créditos para la pantalla final en el cual se muestra los nombres de los participantes en el proyecto y a la profesora a del curso.



Esta es la segunda pantalla que se planteó usar los siguientes widgets: labels para mostrar los datos del aeropuerto, un botón para abrir la ventana de algoritmos y, por último, un botón salir para regresar a la ventana principal.



En esta pantalla, es donde se ejecutarán los algoritmos de floyd-warshall y A\*, los widgets en este apartado son un label donde se colocará el texto de la ruta más corta al aeropuerto, luego se tendrá una entrada donde recibe como parámetro el código del aeropuerto, por último, se tendrá dos botones: uno donde se ejecutará el algoritmo de A\* para obtener la ruta más corta y el segundo un botón donde se ejecutará el algoritmo de floyd-warshall donde se obtendrá la distancia de la ruta más corta.

En cuanto al backend se planteó la siguiente estructura de diseño:

La estructura de datos será un grafo por matriz de adyacencia y lista de adyacencia para tener mayor versatilidad de implementación de los algoritmos.

Además, se implementará una tabla de hash para tener acceso a los aeropuertos (clase principal) teniendo como id un número.

se requieren los siguientes algoritmos implementados en el lenguaje Python

* algoritmo de lectura de datos
* algoritmo de aleatoriedad
* algoritmo de actualización de aristas y su valor de ponderación.
* algoritmo para determinar la distancia entre dos puntos geodésicos
* algoritmo de creación de relaciones
* algoritmo de asignación de aristas dado una relación
* algoritmo de A\*
* algoritmo de floyd-warshall
* algoritmo de búsqueda en una tabla de hash

## Implementación

La implementación se llevó a cabo en Visual Studio Code, bajo el lenguaje de programación Python.

Los frameworks utilizados son:

* Tkinter
* Numpy
* Random
* Math
* Datetime
* Copy
* Csv

El repositorio de GitHub del proyecto es el siguiente:

<https://github.com/JosephRamirez-F4fu/Proyecto-SkyShip-Complejidad-Algoritmica-cc184-2202-cc41.git>

## Verificación

Se realizarán las pruebas del proyecto realizado para demostrar su funcionamiento:

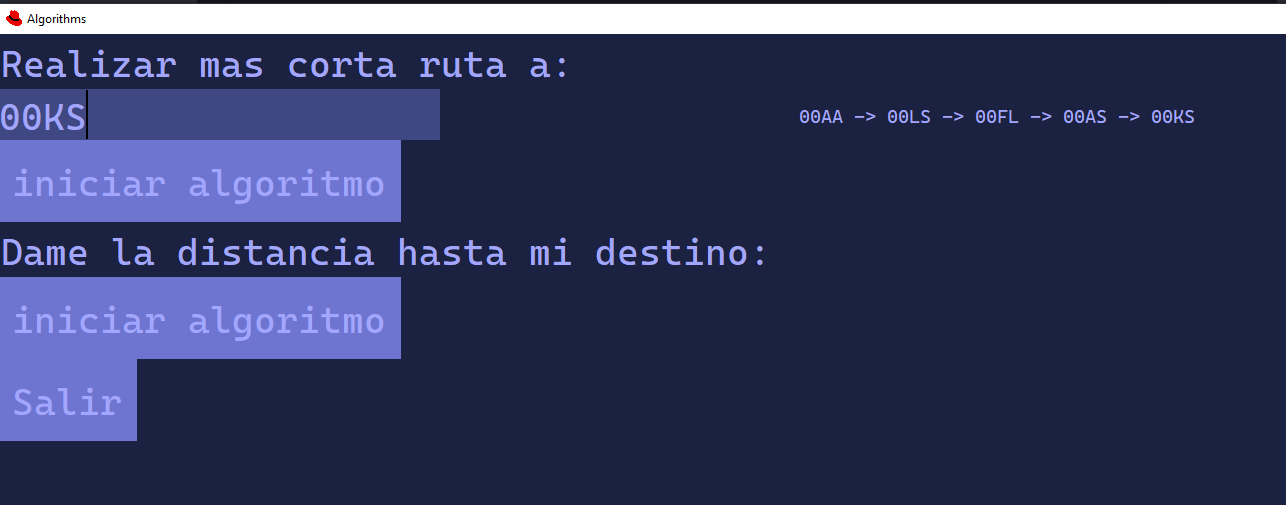
* Con 20 nodos

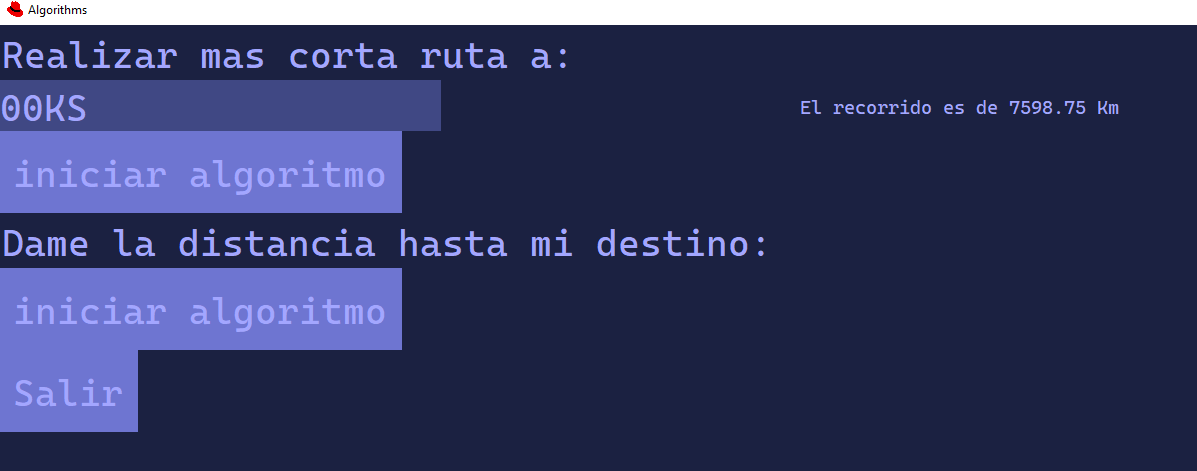
Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Teams

Descripción generada automáticamente

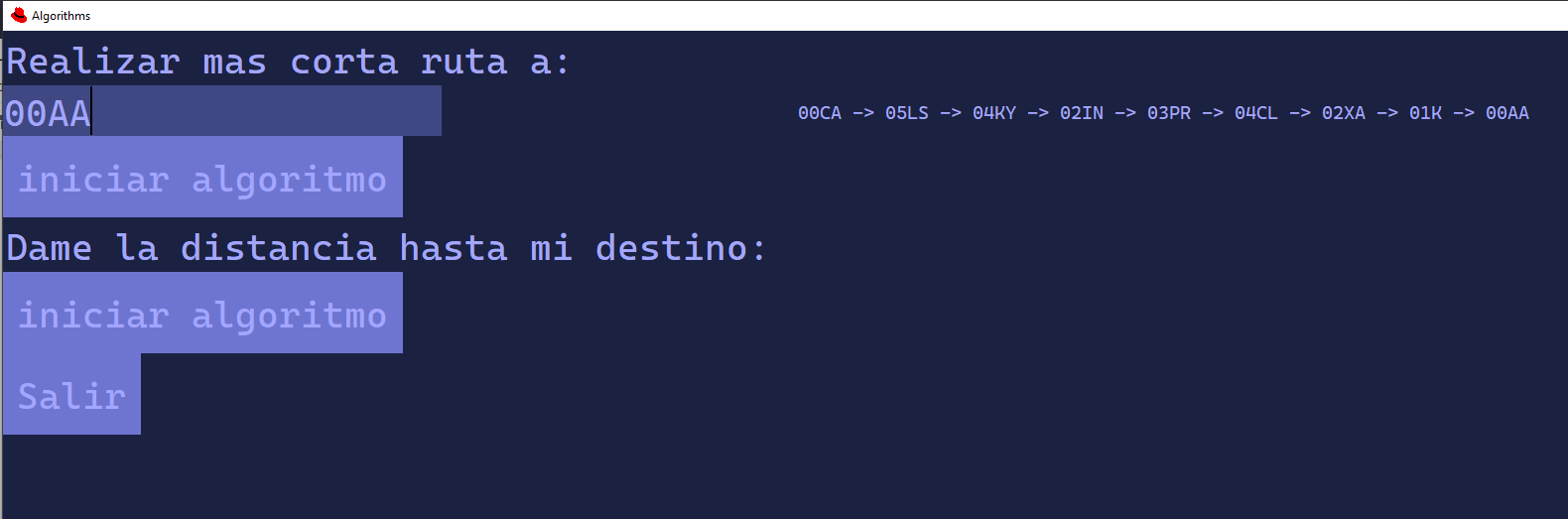


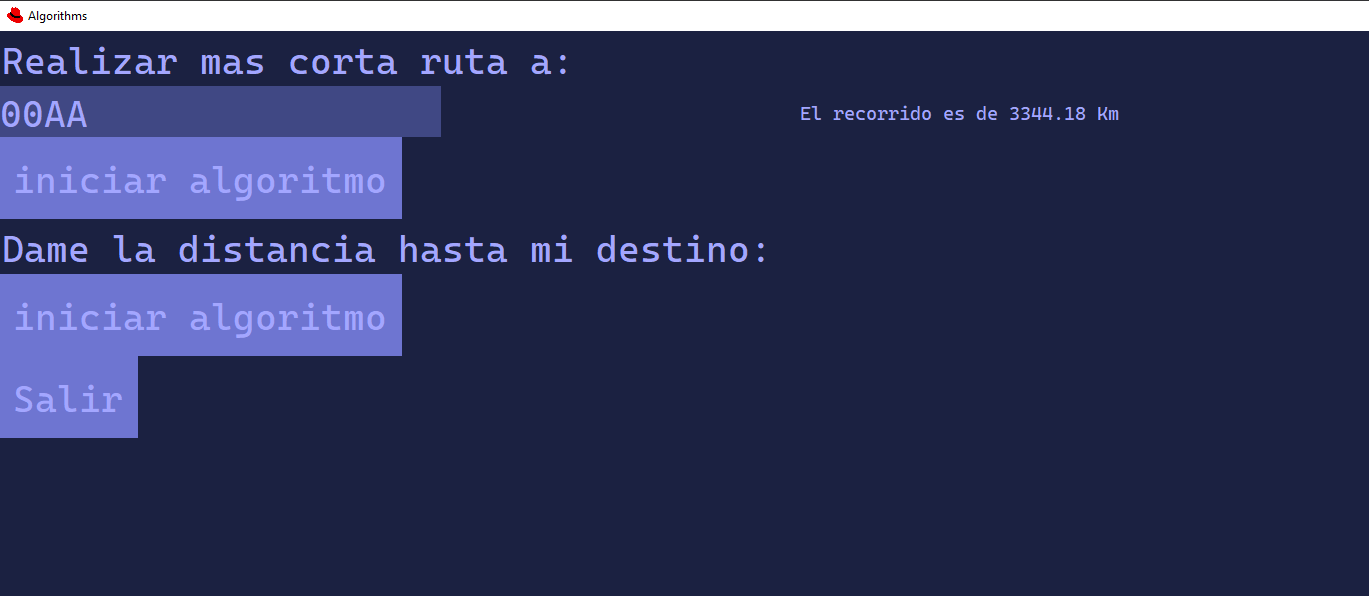


* Con 200 nodos

Interfaz de usuario gráfica, Texto, Aplicación, Teams

Descripción generada automáticamente

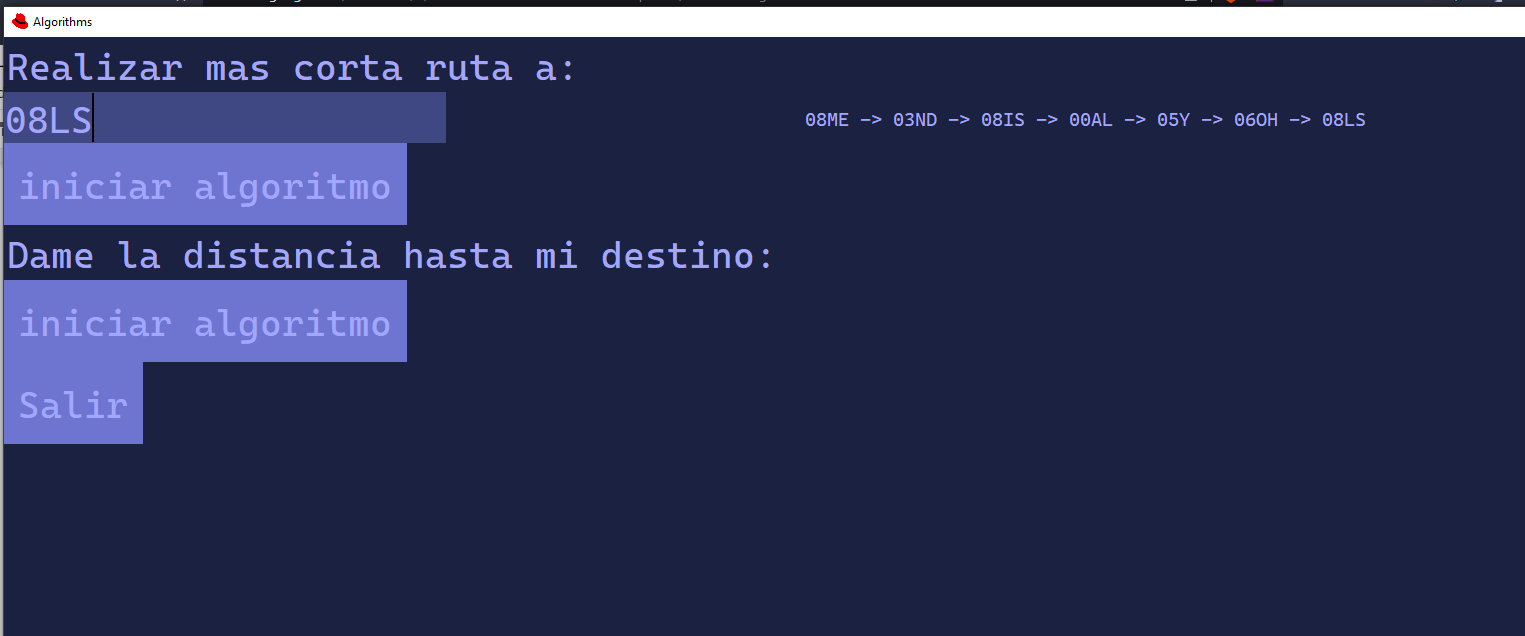


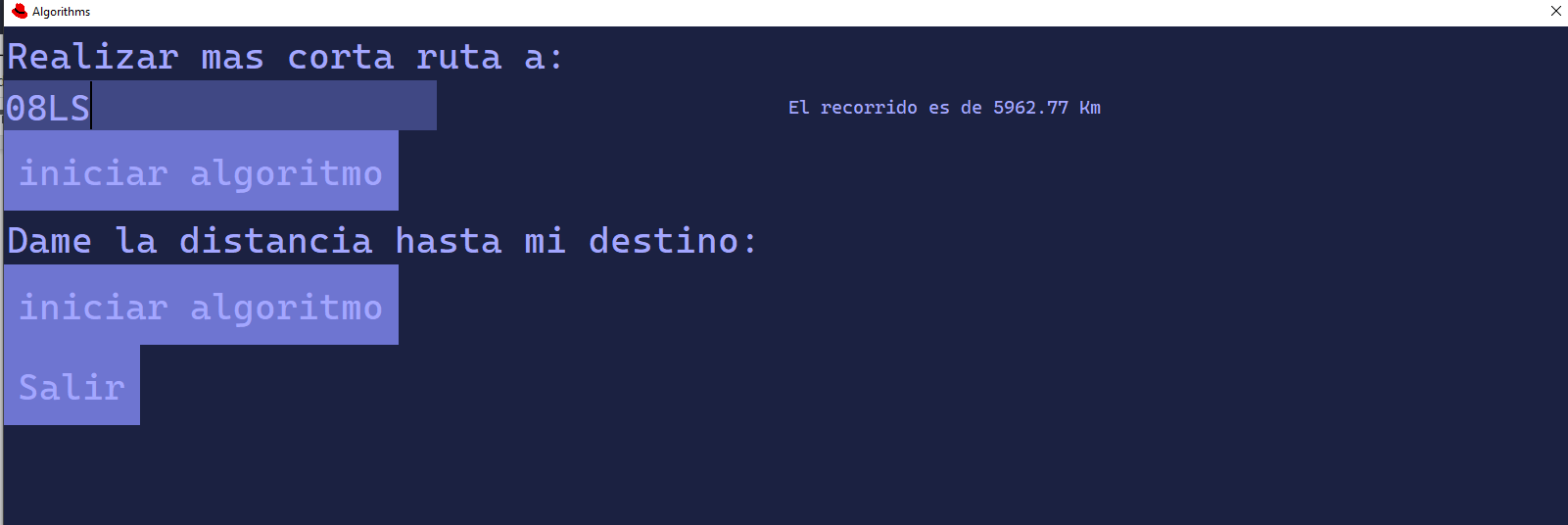


## Mantenimiento o puesta en producción

Aquí se pondrá en ejecución de los 1500 nodos, cual es el objetivo del proyecto.

# 





# Validación y puesta en producción

Como se puede observar en el apartado de mantenimiento en producción tenemos entradas en el código del aeropuerto del cual desea realizar una ruta, luego de ingresar un código del aeropuerto de manera satisfactoria podremos ingresar al apartado home y luego ir al apartado algoritmos. Luego, podremos ingresar otro aeropuerto de destino mediante su código y en caso sea correcto podremos visualizar la ruta si y en caso de que dicha ruta no sea posible se mostrará que la dicha ruta no es posible.

# Conclusiones

Se concluye que es posible resolver la problemática propuesta usando los algoritmos planteados. Además, de los algoritmos se puede concluir que A\* mantiene tiempo de ejecución más bajo que el Floyd-warshall debido que este tiene un O(n^3) pero este tiempo asintótico se compensa debido hacer que haga uso de programación dinámica y luego resolver la problemática conserva las respuestas sin tener que volver a calcularlas. También, la interfaz gráfica tkinter es fue muy versátil para la realización de la interfaz gráfica porque nos permitió un escalamiento rápido. Incluyendo, la realización del trabajo mediante módulos y el uso de objetos favoreció en la distribución de requerimientos a realizar.

# Bibliografía

*Our airports. (2022). Our Airports @Open data*. [Conjunto de datos] <https://ourairports.com/data/>

Los vuelos se encarecerán aún más, según el presidente de la IATA. (2022, 11 julio). *BBC News Mundo*. Recuperado 25 de septiembre de 2022, de <https://www.bbc.com/mundo/noticias-62117381>

World Energy Trade. (2022, 1 marzo). *Los costos del transporte de petróleo se disparan en las rutas clave tras la invasión rusa*. Recuperado 25 de septiembre de 2022, de <https://www.worldenergytrade.com/finanzas-energia/mercado/los-costos-del-transporte-de-petroleo-se-disparan-en-las-rutas-clave-tras-la-invasion-rusa>

Jain, S. (2021, 5 marzo). *The Insider’s Guide to A\* Algorithm in Python*. Python Pool. <https://www.pythonpool.com/a-star-algorithm-python/>

*Floyd-Warshall Algorithm*. (s. f.). <https://www.programiz.com/dsa/floyd-warshall-algorithm>

John Ortiz Ordoñez. (2020, 6 marzo). *Python - Ejercicio 744: Calcular Distancia entre Dos Puntos Definidos por la Latitud y la Longitud*. YouTube. <https://www.youtube.com/watch?v=ylTHl3bF9nY>