

Reporte Tarea 1

García Lugo Juan

August 2019

1 Introducción

En este artículo se encuentra el reporte acerca del desarrollo de la "Tarea01", la definición del problema, análisis del problema, selección de la mejor alternativa, una explicación del funcionamiento de la aplicación explicado de manera conceptual a través de un diagrama de flujo o pseudocódigo, el mantenimiento y depuración de la aplicación.

2 Definición del problema

Se necesita una aplicación que nos muestre en tiempo real el clima de las ciudades en la base de datos de un aeropuerto donde se están registrados los vuelos del día, el clima tiene que ser al tiempo al que se ejecute el programa.

La base de datos está definida de la siguiente manera, sea i un renglón de la base de datos con $i > 1$, $i =$ ciudad de origen, ciudad de destino, latitud origen, longitud origen, latitud destino, longitud destino.

3 Análisis del problema

En el problema nos piden una aplicación que nos proporcione el clima en tiempo real, de las ciudades de origen y de llegada, con sus respectivas coordenadas, entonces el flujo general del programa debería de ser el siguiente: leer la base de datos, obtener el clima en tiempo real de las ciudades, mostrar el clima de las ciudades de los vuelos en la base de datos, en este caso los problemas o puntos principales que tendremos que resolver para desarrollar la aplicación son los siguientes.

- Leer una base de datos con formato csv: al leer la base de datos tendremos que obtener las ciudades para poder conocer el clima de dicha ciudad.
- Obtener el clima en tiempo real: para ello se necesita de un servidor web que cumpla con lo que pedimos.

- Evitar saturar el servidor: para eso tendremos que modelar un cache donde se guarden las ciudades que ya han realizado una petición del clima en el web service

4 Selección de la mejor alternativa

Para realizar este proyecto había varias alternativas que eran bastante buenas, pero en este python ganó ya que es un lenguaje de alto nivel, fácil de aprender, con una sintaxis muy amena, es fácil leer archivos, y tiene una gran variedad de librerías que sirven para la interacción con un web service.

5 Diagrama de flujo

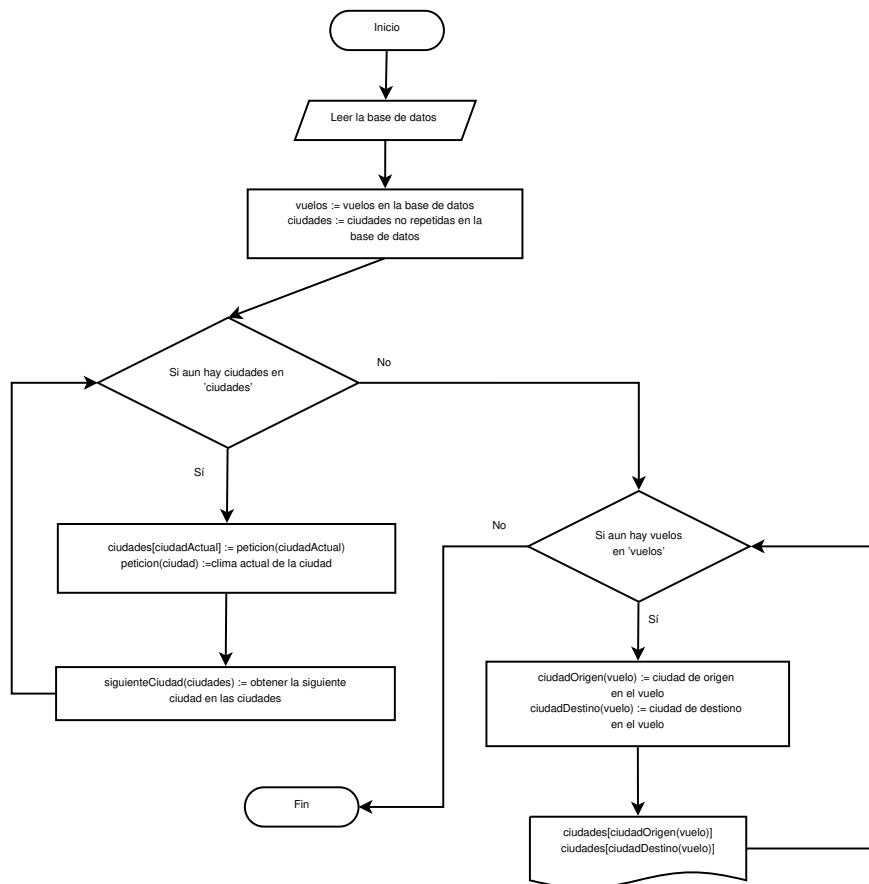


Figure 1: Diagrama de Flujo

Como podemos notar en la figura 1 el flujo del programa es bastante sencillo, esto se logra por la repartición de deberes en los distintos módulos en python de la aplicación, y librerías utilizadas , pero vamos a describirlo.

El flujo del programa del programa se describe de manera secuencial, pero no se describen todas rutinas que realiza el programa ya que la función del diagrama es sólo para mostrar en general que es lo que realiza el programa

- Se lee la base de datos.
- Se guardan los vuelos de la base de datos.
- A partir de los vuelos en la base de datos se obtiene un diccionario con todas las ciudades donde las llaves del diccionario son las ciudades.
- Se iteran las llaves del diccionario y con las mismas se realiza una petición a la web service, y el valor de cada llave será el resultado de la petición.
- Por último se iteran los vuelos guardados, que contienen las ciudades de origen y de llegada, ambas ciudades se ocupan como llaves para obtener el clima de dicha ciudad.

6 Mantenimiento y costo del software

Ahora que nuestro programa cumple con los requisitos funcionales, lo mejor sería implementar requisitos no funcionales tales como: una interfaz gráfica que permita al usuario añadir su base de datos a través de una ventana, manejar hilos de ejecución, para poder realizar las peticiones más rápido, y tener un mecanismo que actualice el clima de las ciudades cada cierto tiempo, suponiendo que se llegase a dar el mantenimiento mencionado, creo que un buen precio por el software sería de 10,000 con mantenimiento de un año