

---

## MANUAL TECNICO - PROYECTO NO. 2 -

---

201212891 – Edgar Rolando Ramírez López

### Resumen

El programa de Gestión de Drones es una aplicación que permite la administración integral de drones, sistemas de drones y mensajes relacionados con alturas. Su arquitectura está compuesta por múltiples clases, incluyendo Dron, SistemaDrones, Mensajes, y más, organizadas en un diseño de lista doblemente enlazada. Los archivos XML se utilizan para cargar y guardar datos. Los usuarios pueden interactuar con la interfaz de usuario proporcionada por main.py para gestionar drones individuales, sistemas de drones y mensajes. Cada componente tiene funcionalidades específicas, como la creación de mensajes con instrucciones y la asignación de alturas. El programa también incluye diagramas de los sistemas de drones. Este manual técnico detalla la estructura, funcionamiento y uso del programa, destinado a desarrolladores y usuarios avanzados en el ámbito de la gestión de drones.

### Palabras clave

Python, funciones, clases, listas, librerías.

### Abstract

*The Drone Management Program is an application that allows for comprehensive administration of drones, drone systems, and altitude-related messages. Its architecture consists of multiple classes, including Drone, DroneSystem, Messages, and more, organized in a doubly linked list design. XML files are used to load and save data. Users can interact with the user interface provided by main.py to manage individual drones, drone systems, and messages. Each component has specific functionalities, such as creating messages with instructions and assigning altitudes. The program also includes diagrams of drone systems. This technical manual details the structure, operation, and usage of the program, intended for developers and advanced users in the field of drone management.*

### Keywords

*Python, functions, class, lists, libraries.*

## Introducción

El Programa de Gestión de Drones es una aplicación avanzada diseñada para administrar y supervisar eficientemente drones, sistemas de drones y mensajes relacionados con alturas. Esta poderosa herramienta permite a los usuarios llevar un seguimiento detallado de múltiples drones, establecer sistemas de control y monitorear mensajes con alturas específicas. Con una interfaz de usuario amigable, este programa brinda una experiencia de usuario intuitiva y fácil de usar, lo que facilita la gestión de drones en diversas aplicaciones. Este manual técnico proporciona una visión detallada de la estructura y funcionamiento del programa para ayudar a los usuarios a aprovechar al máximo sus capacidades.

## Desarrollo del tema

En el desarrollo de la práctica se utilizó el lenguaje Python 3.11.1, el IDE utilizado fue Visual Studio Code y también se subió la practica a GitHub. El paradigma utilizado en el proyecto fue la programación con orientación a objetos. Las librerías utilizadas fueron xml.etree.ElementTree, graphviz y subprocess.

### a. Python

Python es un lenguaje de programación interpretado multiparadigma, multiplataforma y de código abierto. Diseñado para facilitar la programación orientada a objetos, permite trabajar con un gran número de estructuras de datos como diccionarios, listas, tuplas y conjuntos. Es compatible con muchos sistemas operativos, como Windows, Mac OS, Linux, y también se puede usar para el desarrollo web, scripting y desarrollo de aplicaciones. Ofrece una

extensa biblioteca estándar con muchas utilidades prácticas. Está aumentando su presencia en la industria de la tecnología y está ganando popularidad por su facilidad de uso y simplicidad.

### b. Visual Code Studio (VSCODE)

Un IDE es un entorno de desarrollo integrado, es una aplicación de software que ayuda a los programadores a desarrollar código de software de manera eficiente. Visual Studio Code es un editor de código fuente de código abierto y gratuito desarrollado por Microsoft para MacOS, Linux y Windows. Incluye soporte para desarrollo web, lenguajes de edición de línea de comandos y lenguajes de scripting modernos, así como lenguajes de programación como HTML, CSS, CoffeeScript, PHP, Ruby y Python. Ofrece características como resaltado de sintaxis, completado inteligente, intenciones y muchos más. La eficiencia y la facilidad de uso de Visual Studio Code han hecho de él una herramienta de desarrollo muy utilizada en todo el mundo.

### c. GitHub

GitHub es una plataforma de alojamiento web que ayuda a los desarrolladores a almacenar y mantener el código fuente para proyectos de software. Utiliza el sistema de control de versiones Git para permitir a los usuarios colaborar en el desarrollo de proyectos, realizar revisión de código, realizar tareas de seguimiento de errores y realizar pruebas. También ofrece una variedad de herramientas de integración de terceros, navegación web y soporte para proyectos públicos y privados.

#### d. Librería `xml.etree.ElementTree`

Biblioteca estándar de Python que se utiliza para analizar y manipular documentos XML de una manera simple y eficiente. Proporciona una interfaz de alto nivel para acceder a elementos y atributos XML, así como para crear y modificar documentos XML. `ElementTree` facilita la lectura y escritura de archivos XML, lo que lo hace útil para procesar datos estructurados en formato XML en aplicaciones Python.

#### e. Librería `subprocess`

Módulo en Python que proporciona una interfaz para trabajar con procesos (programas o comandos) externos al script de Python en sí. Permite a los programas de Python crear, ejecutar y comunicarse con otros procesos o ejecutar comandos del sistema operativo en la línea de comandos.

#### f. Librería `Graphviz`

Herramienta de software utilizada para crear y renderizar gráficos y diagramas, especialmente diagramas de grafos. Esta librería proporciona un conjunto de herramientas y funciones que permiten la creación y visualización de estructuras de datos gráficas de manera programática.

#### g. XML (Extensible Markup Language)

Lenguaje de marcado diseñado para almacenar y transportar datos de manera estructurada. XML se utiliza comúnmente para representar información en forma de texto legible tanto por humanos como por máquinas.

### Estructura del Código

Se describirán brevemente los componentes clave del programa, incluidas las clases y los módulos.

#### Nodo (`nodo.py`)

La clase `Nodo` es una clase base que se utiliza como componente fundamental para implementar una lista doblemente enlazada en tu programa. Tiene las siguientes propiedades y funcionalidades:

- **siguiente:** Un atributo que almacena una referencia al siguiente nodo en la lista.
- **anterior:** Un atributo que almacena una referencia al nodo anterior en la lista.

#### *Funcionalidad:*

- Permite la creación de nodos que contienen datos y enlaces a otros nodos.
- Se utiliza para crear la estructura de datos de una lista doblemente enlazada, donde los nodos pueden enlazarse en ambas direcciones (hacia adelante y hacia atrás).

#### Alturas (`alturas.py`)

La clase `Alturas` representa las alturas de los mensajes relacionados con los drones. Sus funcionalidades son las siguientes:

- **valor:** Almacena el valor numérico de la altura.
- **letra:** Almacena una letra relacionada con la altura.

#### *Funcionalidad:*

- Proporciona una forma de representar las alturas de los mensajes.

- Permite asociar letras a alturas numéricas para identificar mensajes de manera más comprensible.

### **Dron (dron.py)**

La clase Dron representa un dron individual. Sus funcionalidades son las siguientes:

- **nombre:** Almacena el nombre del dron.

#### **Funcionalidad:**

- Permite crear objetos que representan drones individuales.
- Proporciona un método `mostrar_Nombre_dron` que devuelve el nombre del dron.

### **Instrucciones (instrucciones.py)**

La clase Instrucciones representa instrucciones relacionadas con drones y alturas. Sus funcionalidades son las siguientes:

- **dron:** Almacena un objeto de la clase Dron que indica el dron asociado a la instrucción.
- **altura:** Almacena un objeto de la clase Alturas que indica la altura de la instrucción.

#### **Funcionalidad:**

- Permite crear objetos que representan instrucciones específicas.
- Proporciona dos métodos: `mostrar_instrucciones` y `mostrar_instrucciones_individual`, que devuelven cadenas formateadas con información sobre las instrucciones.

### **ListaDobleCircular (listaDobleCircular.py)**

La clase ListaDobleCircular implementa una lista doblemente enlazada circular, que es utilizada como estructura de datos principal en tu programa. Sus funcionalidades son las siguientes:

- La lista contiene nodos enlazados que almacenan información sobre drones, sistemas de drones y mensajes.

#### **Funcionalidad:**

- Permite la inserción, eliminación y búsqueda de elementos en la lista.
- Proporciona métodos para mostrar listas de drones, sistemas de drones y mensajes.
- Implementa la lógica de una lista doblemente enlazada circular, donde el último nodo está enlazado al primero, formando un ciclo.

### **Mensajes (mensajes.py)**

La clase Mensajes representa mensajes relacionados con sistemas de drones. Sus funcionalidades son las siguientes:

- **nombre:** Almacena el nombre del mensaje.
- **sistemaDrones:** Almacena un objeto de la clase SistemaDrones que indica el sistema de drones asociado al mensaje.

#### **Funcionalidad:**

- Permite crear objetos que representan mensajes.
- Proporciona métodos para mostrar información sobre los mensajes y sus instrucciones asociadas.

SistemaDrones (sistemaDrones.py)

La clase SistemaDrones representa sistemas de drones. Sus funcionalidades son las siguientes:

- **nombre:** Almacena el nombre del sistema de drones.
- **altura\_maxima:** Almacena la altura máxima que pueden alcanzar los drones en el sistema.
- **cantidad\_drones:** Almacena la cantidad de drones en el sistema.

Funcionalidad:

- Permite crear objetos que representan sistemas de drones.
- Proporciona métodos para mostrar información sobre los sistemas de drones y sus contenidos asociados.

Diagrama de clases

Una de las claves importantes del programa es el concepto de hereditario, el cual significa que una clase hereda las propiedades de otra clase y no es necesario copiarlas nuevamente en la nueva clase. En la figura 1 se muestra el diagrama de clases utilizado, la cual todas las clases heredan de la clase Nodo. Esto significa que las clases derivadas (Contenidos, Alturas, Mensajes, etc.) heredan las propiedades y métodos de la clase base Nodo.

La figura 1 muestra que la clase main utiliza la clase listadobleCircular para crear instancias de listas de las otras clases, y ya que hereda de la clase Nodo, no es necesario realizar una nueva clase lista para cada uno.

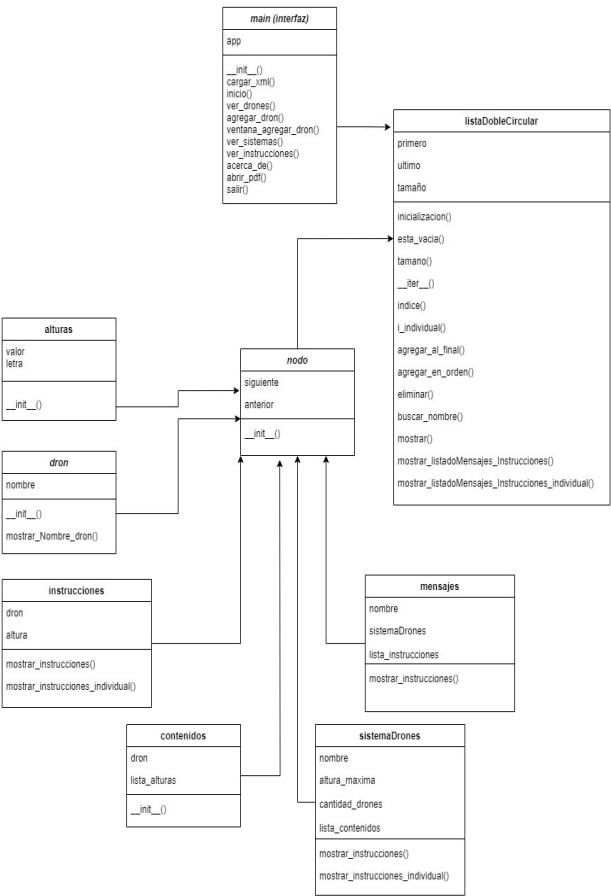


Figura 1. Diagrama de clases

Fuente: elaboración propia

Uso del Programa

El Programa de Gestión de Drones ofrece una variedad de funcionalidades para administrar drones, sistemas de drones y mensajes relacionados con alturas. A continuación, se detallan las principales tareas que se pueden realizar con el programa:

- Ejecute el archivo main.py para iniciar la aplicación.

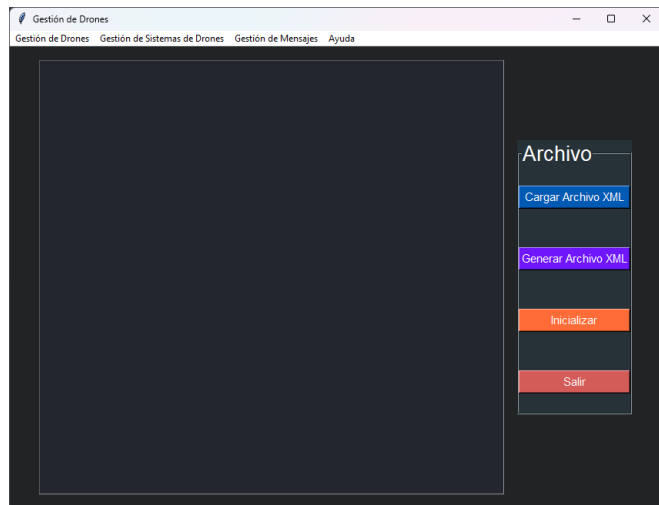


Figura 1. Ventana principal

Fuente: elaboración propia

En la parte de la derecha se apreciará el marco llamado Archivo, el cual se puede apreciar las funcionalidades de:

- **Cargar Archivo XML**
- **Iniciar**
- **Salir**

Para iniciar la gestión de drones, se debe de cargar un archivo XML valido.

En la parte superior se aprecian los menús de Gestión de Drones, Gestión de Sistemas de Drones, Gestión de Mensajes y Ayuda.

Al seleccionar el menú **Gestión de Drones**, se muestran 2 opciones y sus funciones son:

- **Ver listado de drones:** Se visualiza en el cuadro de texto de la ventana principal el listado de drones disponibles del sistema.
- **Añadir dron:** Abre una ventana pequeña donde el usuario puede ingresar un nuevo dron al sistema.

En el menú de Gestion de Sistema de Drones, se muestra la opción:

- **Ver gráficamente el listado de sistema de drones:** Se visualizar en una tabla el sistema de drones con las letras que se cargaron al sistema.

En el menú de Gestion de Mensajes, se muestran 2 opciones y sus funcionalidades son:

- **Ver listado de mensajes y sus instrucciones:** Se visualiza en el cuadro de texto de la ventana principal del programa, el listado de mensajes con sus instrucciones del programa.
- **Ver instrucciones para enviar un mensaje:** Se abre una nueva ventana donde se puede seleccionar un mensaje en un menú desplegable, luego al seleccionar el botón Mostrar Instrucciones y mensaje final, se visualizarán las instrucciones para ese mensaje y el mensaje decodificado final.

Y por último en el menú Acerca de, se mostrará una ventana con los datos del estudiante y un enlace para abrir la documentación del programa en formato pdf.

## Conclusiones

El programa ofrece una solución integral para la gestión de señales de audio en formato XML mediante el uso de listas enlazadas y funcionalidades de procesamiento de datos. Esto permite a los usuarios cargar, procesar y visualizar señales de manera eficiente. Su versatilidad y capacidades hacen que sea una herramienta valiosa para una variedad de aplicaciones relacionadas con el procesamiento de señales y la representación gráfica de datos de audio.

## Referencias bibliográficas

Welcome to. (2023, 15 febrero). Python.org. <https://www.python.org/>

Documentation for Visual Studio Code. (2021, 3 noviembre). <https://code.visualstudio.com/docs>

Hello World. (s. f.). GitHub Docs. <https://docs.github.com/en/get-started/quickstart/hello-world>

J. (2022, 16 enero). *Funciones en Python: Definición de función y para qué se utilizan*. J2LOGO. <https://j2logo.com/python/tutorial/funciones-en-python/>

Bustamante, S. J. (2021, 21 febrero). *Guía de funciones de Python con ejemplos*. freeCodeCamp.org. <https://www.freecodecamp.org/espanol/news/guia-de-funciones-de-python-con-ejemplos/>

Programación orientada a objetos. (s. f.). <https://www.ibm.com/docs/es/spss-modeler/saas?topic=language-object-oriented-programming>

J. (2022b, enero 16). *Programación orientada a objetos (POO) en Python*. J2LOGO. <https://j2logo.com/python/tutorial/programacion-orientada-a-objetos/>

## Anexos

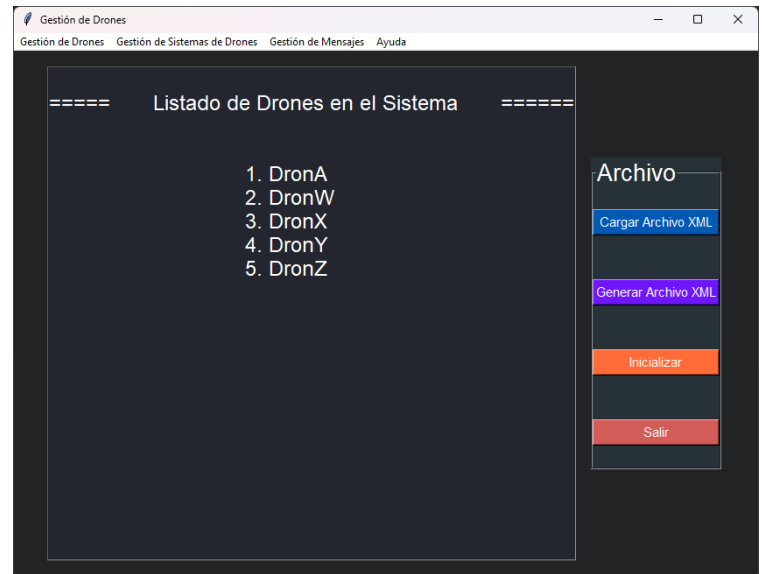


Figura 2. Listado de drones

Fuente: elaboración propia

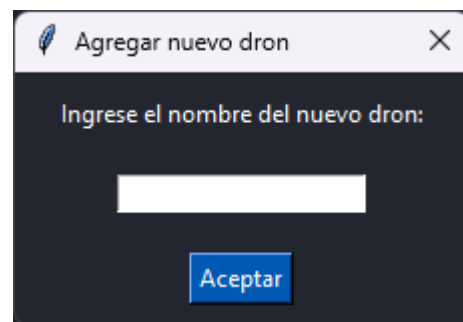


Figura 3. Agregar un nuevo dron

Fuente: elaboración propia

Sistema de Dron SDF			
DronW	DronX	DronY	DronZ
I	P	C	3

Sistema de Dron SD1		
DronX	DronY	DronZ
A	2	B
I	C	C
D	P	E
F	G	H
I	L	O
J	M	P
K	N	Q

Figura 4. Ver sistema de drones  
Fuente: elaboración propia

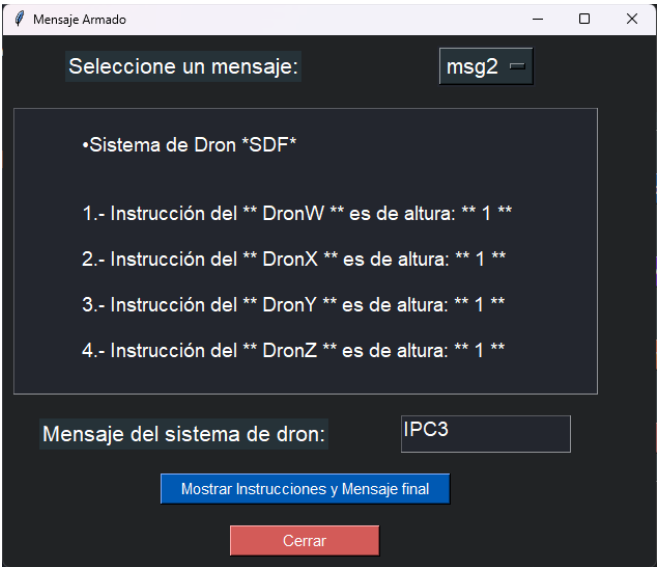


Figura 6. Decodificar mensaje  
Fuente: elaboración propia

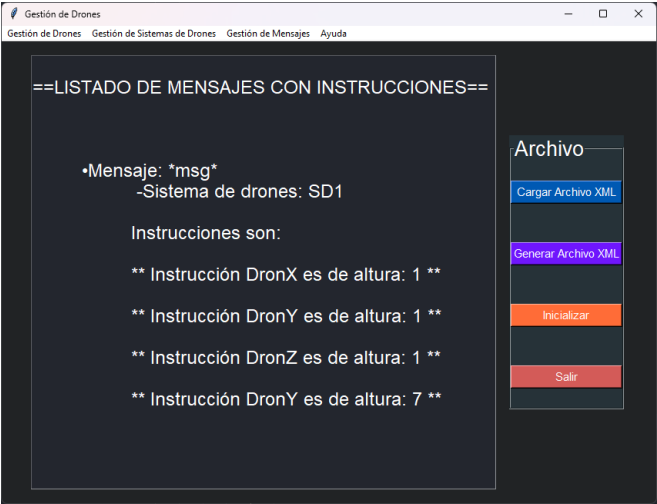


Figura 5. Listado mensajes  
Fuente: elaboración propia

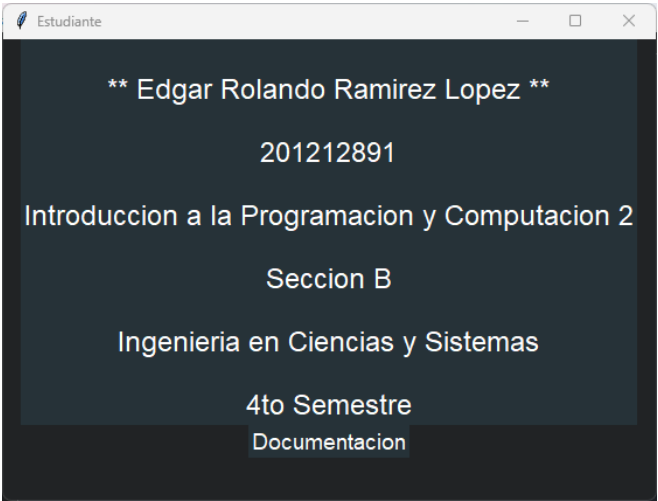


Figura 7. Datos de estudiante  
Fuente: elaboración propia