***UNIVERSIDAD NACIONAL DE ASUNCIÓN***

***FACULTAD POLITÉCNICA***

**EDITOR DE ORGANIGRAMAS**

**Integrantes:**

* Atilio Sebastián Paredes Pérez.
* Santiago Damián Sotelo Rejala.
* Elias Marcelo Riveros Vera.
* Fernando David Fleitas Cáceres**.**

**Materia:** Algoritmos & Estructuras de Datos II.

**Carrera:** Ingeniería en Informática.

**Semestre:** 2do.

**AÑO: 2023.**

**San Lorenzo – Paraguay.**

**INVESTIGACIÓN**

* **PYQT5 y QT DESIGNER:**

PyQt5 es una biblioteca de enlace para el lenguaje de programación Python que permite crear aplicaciones de escritorio con interfaces gráficas de usuario (GUI, por sus siglas en inglés) utilizando el framework Qt. Qt es un framework multiplataforma ampliamente utilizado para desarrollar aplicaciones de software en C++ y ofrece una amplia gama de funcionalidades para la creación de interfaces de usuario avanzadas. PyQt5 proporciona una capa de abstracción sobre Qt y permite a los desarrolladores utilizar todas las capacidades de Qt en sus aplicaciones de Python.

* **GRAPHVIZ:**

Graphviz es un software abierto de libre distribución para graficar, que presenta información estructural en forma de diagramas y puede aplicarse en diversas áreas como el análisis de redes, bioinformática, ingeniería de software, bases de datos, diseño de sitios web, aprendizaje por computadora y tiene interfaces gráficas para otros dominios. Su modo de utilización se basa en el diseño de pequeños programas que toman descripciones de los diagramas de un lenguaje de texto simple y los dibuja en diversos formatos tales como archivos de imágenes, SVG, PDF ó para desplegarse en exploradores.

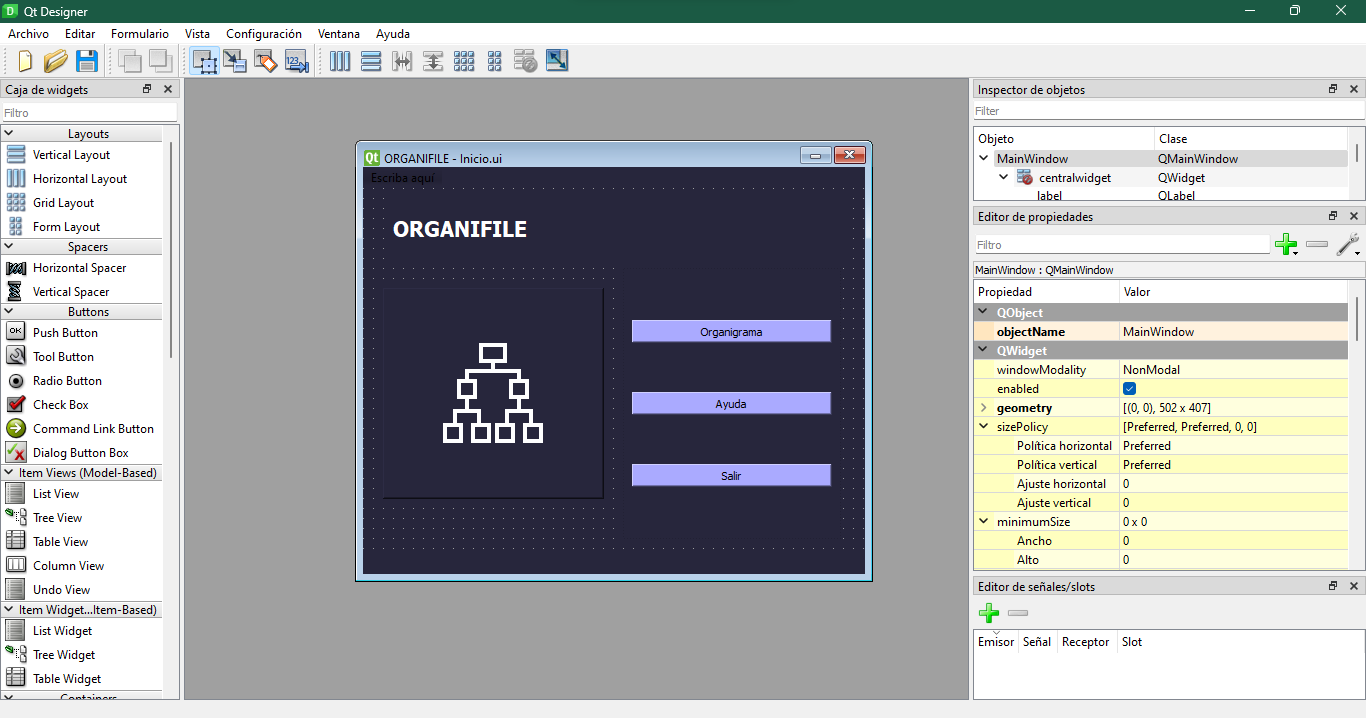
En la práctica, generalmente las gráficas se generan de fuentes de datos externas, (típicamente mediante el recorrido de la estructura de datos correspondiente) para crear un programa que pueda ser interpretado y ejecutado por alguno de los componentes de Graphviz y generar la imagen correspondiente.

* **SQLITE:**

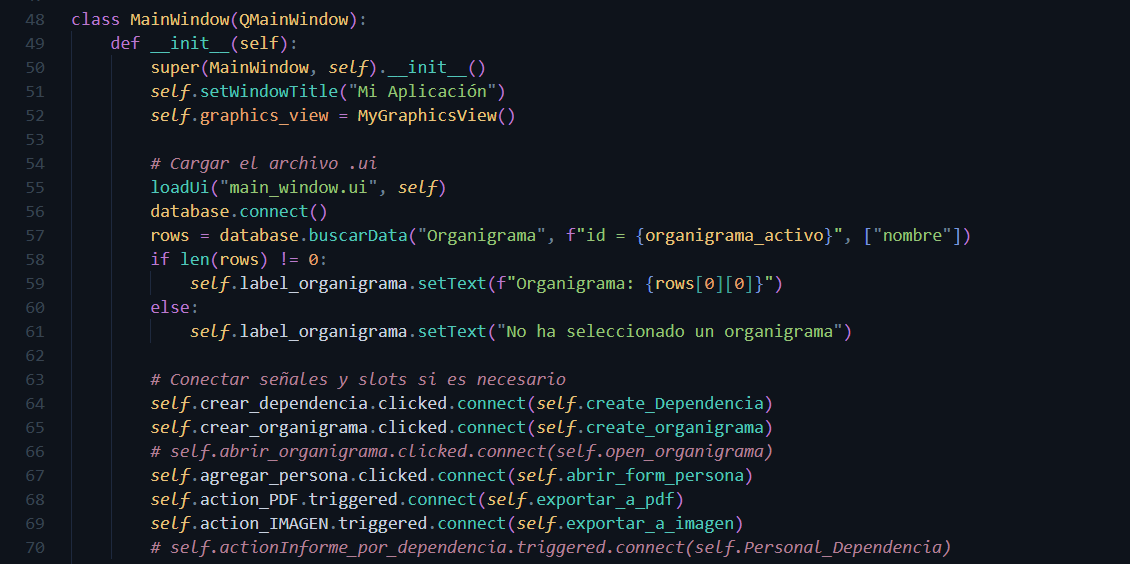
SQLite es una popular base de datos relacional, Se compone de una biblioteca en lenguaje C y se diferencia de otros sistemas de gestión de bases de datos al funcionar como un servidor propio e independiente, eliminando la necesidad de consultas y procesos separados.

La biblioteca SQLite se genera y almacena directamente en el archivo de la base de datos, lo cual lo hace práctico y accesible para aplicaciones más simples, tanto en entornos desktop como mobile, sitios web ligeros y sistemas con un número reducido de usuarios. SQLite utiliza el lenguaje SQL (Structured Query Language) para realizar consultas y manejar la base de datos. También puede combinarse con scripts en Python para análisis de datos más complejos.

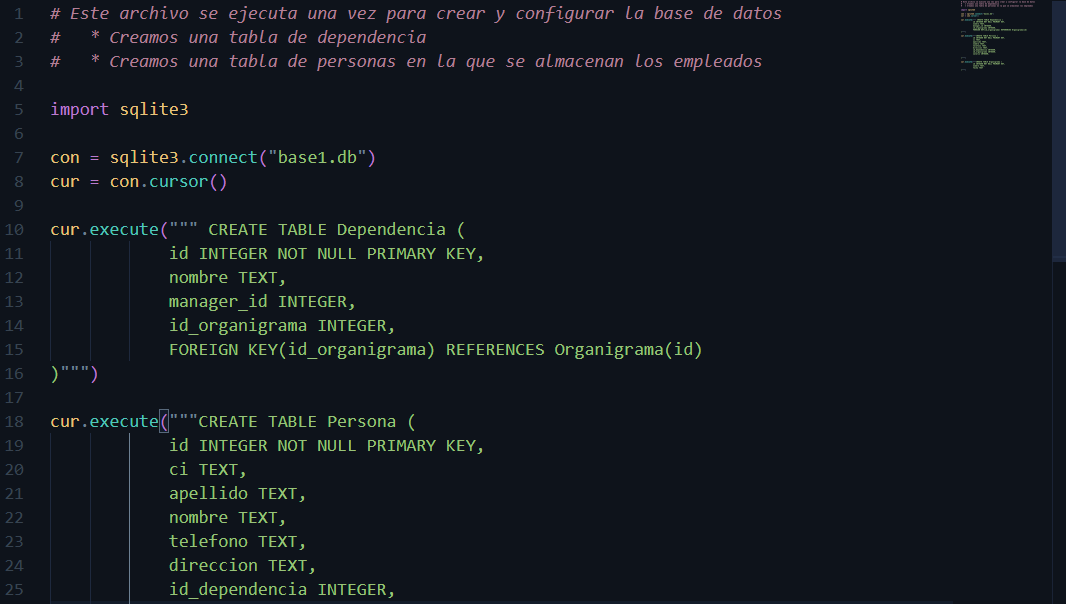
**¿COMO UTILIZAMOS LAS LIBRERIAS?**

 El entorno gráfico de la aplicación se crea utilizando la herramienta de Qtdesigner, la cual ofrece una amplia gama de utilidades, funciones y opciones para organizar una interfaz de usuario de manera efectiva.

Entorno de qtdesigner.

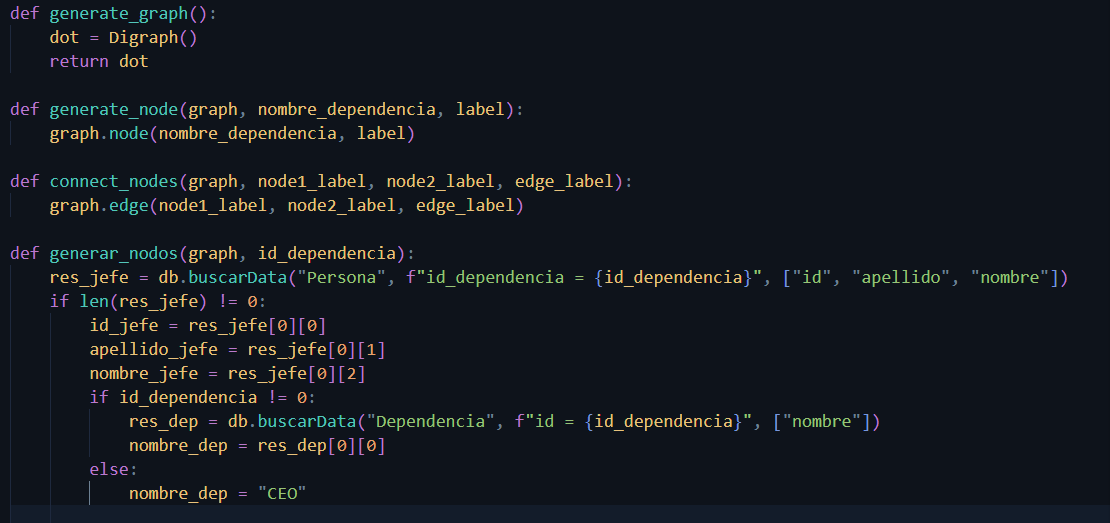
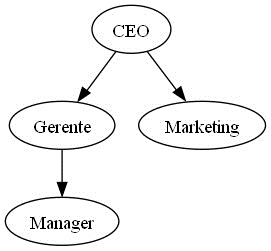
Con PYQT5 nos encargamos de darle vida a cada botón que tenga la interfaz gráfica a través de clases y métodos.

Carga de la ventana principal y definición de botones.

Sqlite3 se encarga de crear una estructura de base de datos para almacenar información sobre dependencias, personas (empleados) y el organigrama de una organización.

Creación de tablas en sqlite3.

Mediante Graphviz realizamos la función de mostrar en pantalla los resultados del organigrama, sus conexiones, nombres y dependencias.



Ejemplo de ilustración creada con graphviz.

Código de conexón de nodos.

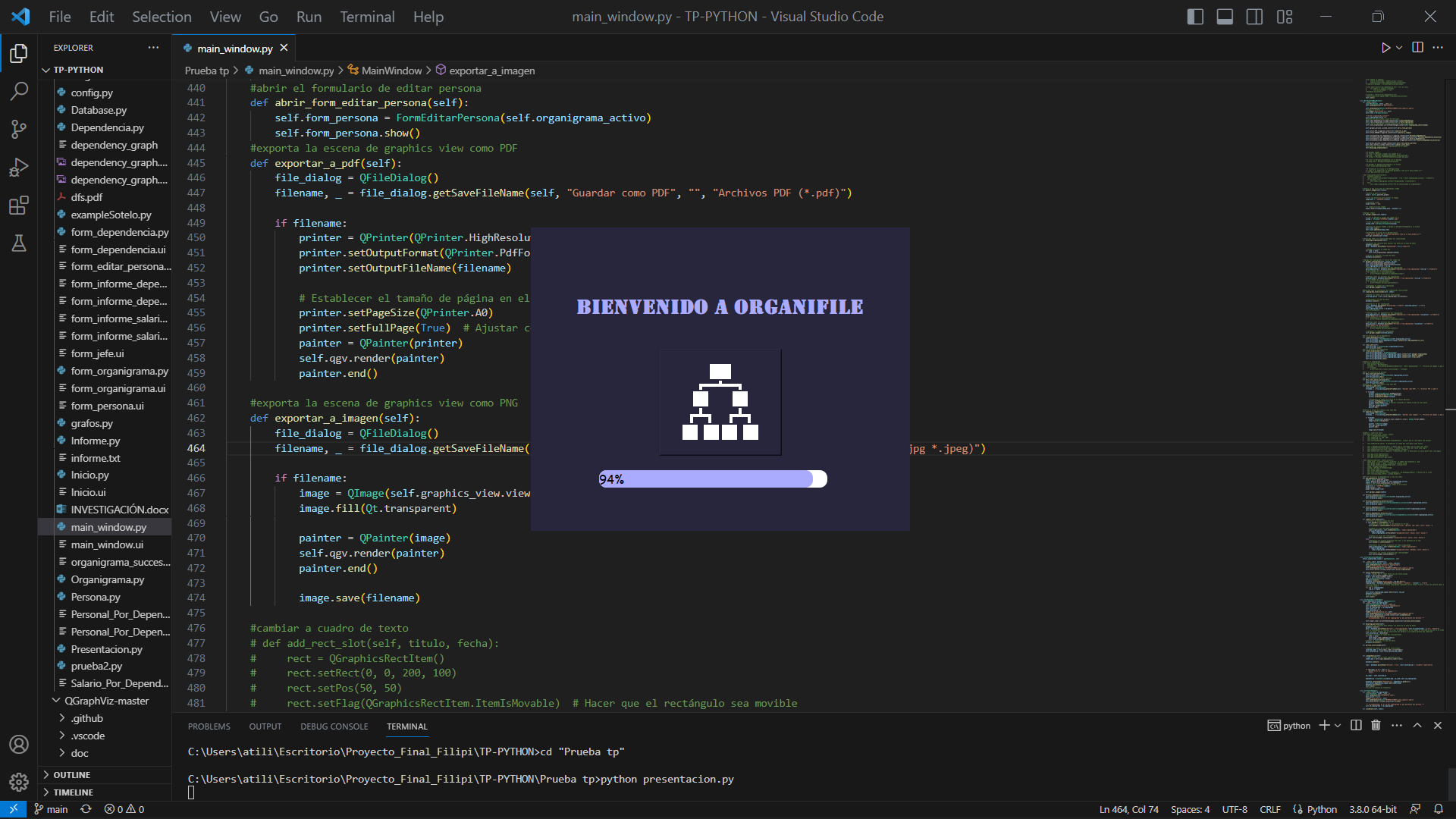
**DISEÑO**

**Icono del programa**

****

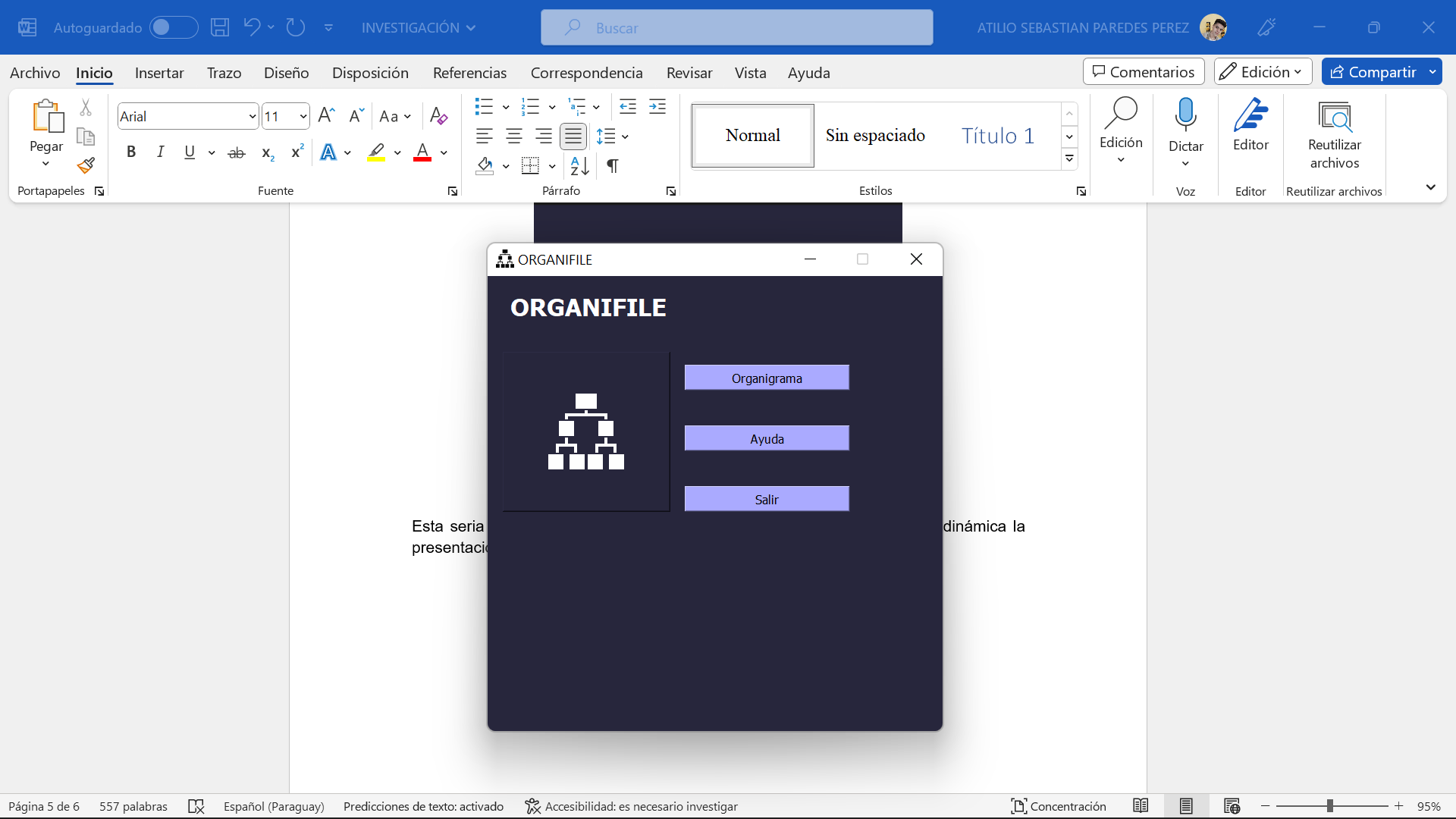
El icono del programa es sencillo representativo, y bastante minimalista, decidimos elegir un icono así para poder representar el significado del editor de organigramas, como organigrama y estructura organizativa.

**Pantalla de Carga**



Esta sería la pantalla de carga del programa, creada para que sea más dinámica la presentación del proyecto.

**Pantalla de Inicio del Programa**



Fue creada para que, al cargar toda la barra de progreso de la presentación, aparezca un menú, para elegir las opciones correspondientes.

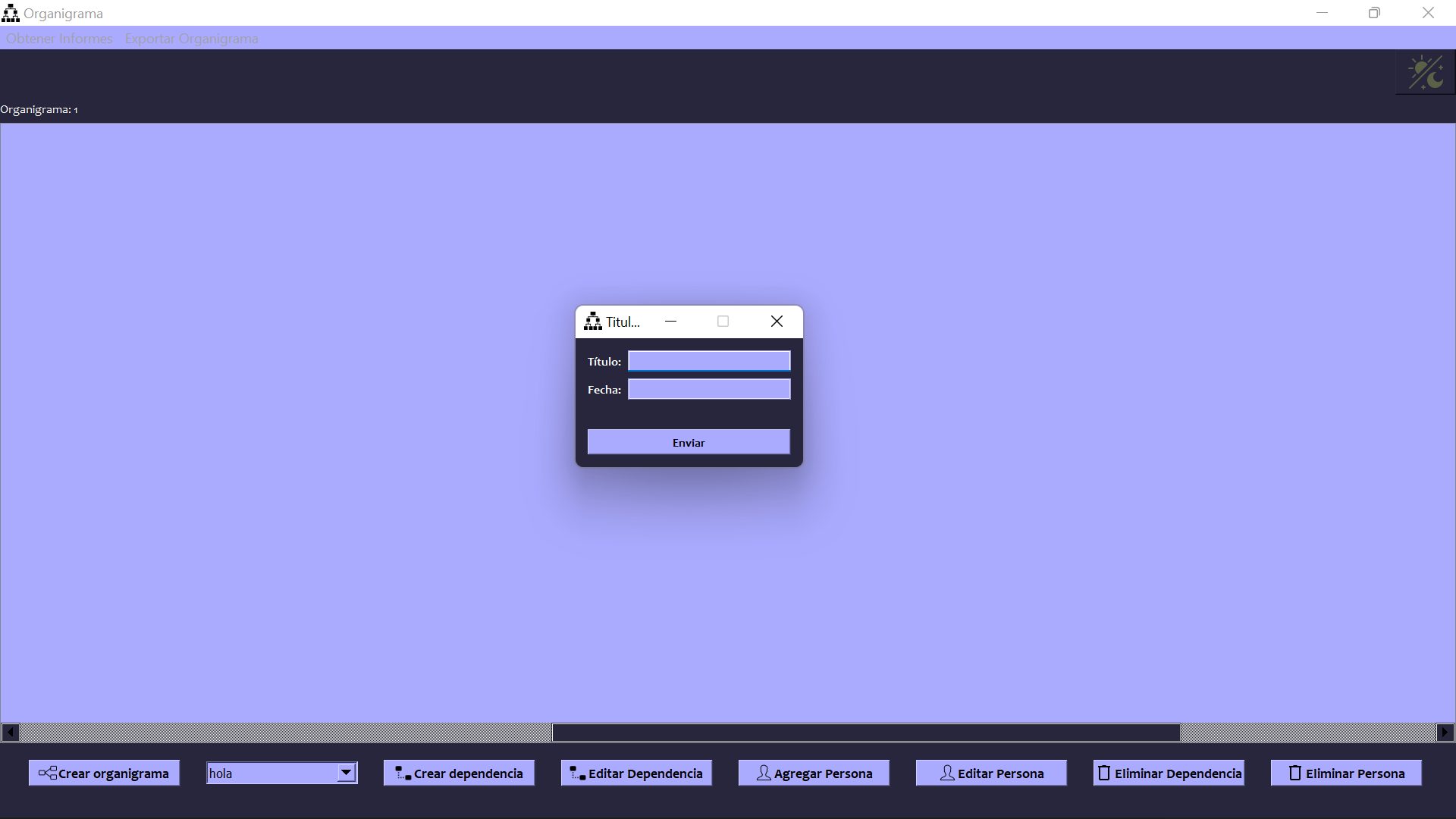
**Pantalla del menú Principal**



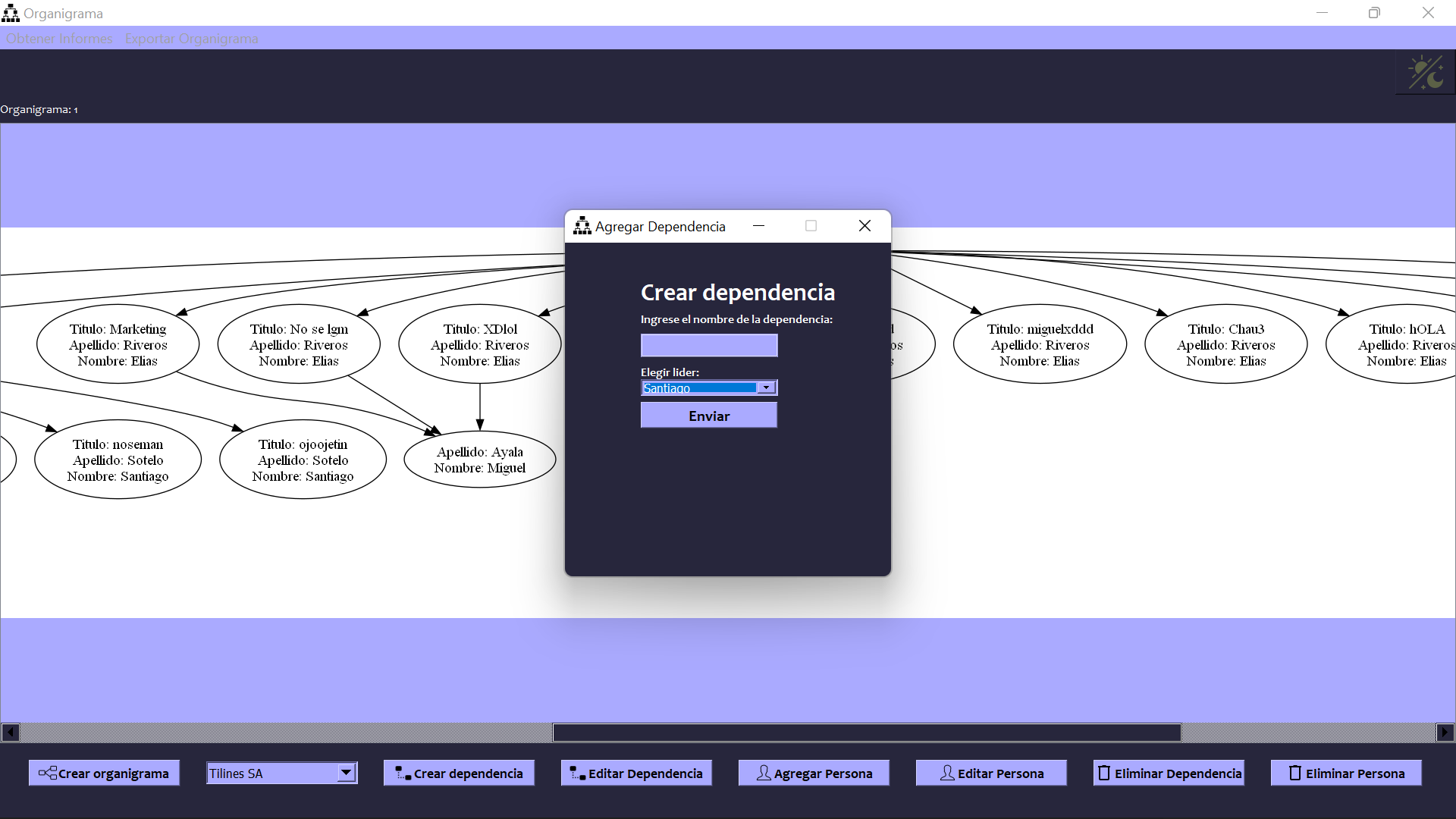
Esta es la interfaz del menú principal, inspirada en un editor manipulable y dinámico, está diseñada para poder visualizar el organigrama creado, poder manipular los nodos más fácilmente, agregar personas/dependencias, eliminar las personas/dependencias, obtener los informes y poder guardar en el formato deseado el organigrama creado, todo esto a solo unos clics de distancia.

**FORMULARIOS**

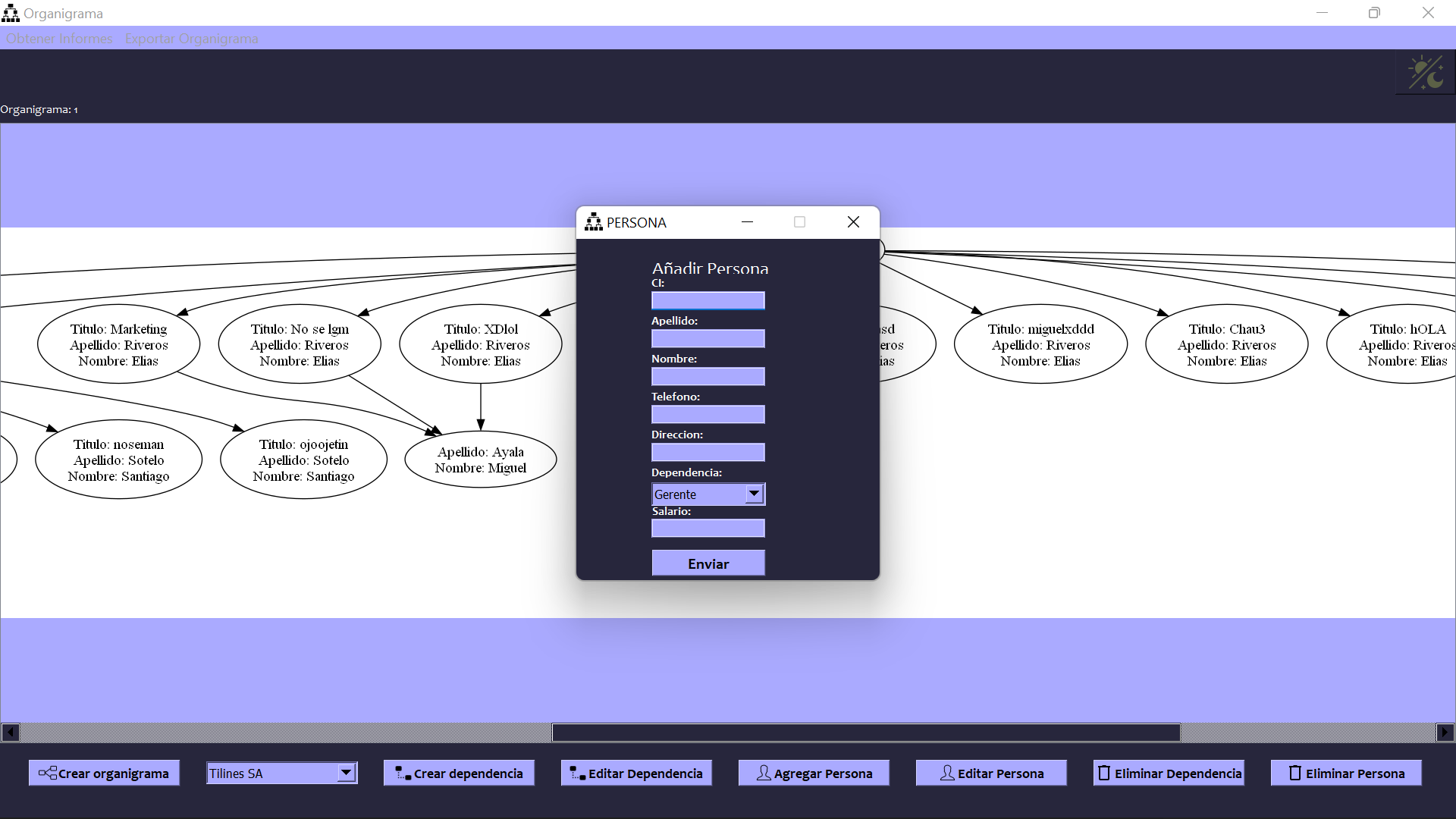
**Formulario para crear el organigrama**



Simple y dinámico, recibe el título del organigrama y la fecha de este.

**Formulario para crear dependencia**

Este formulario se utiliza para crear la dependencia, diseñado para ser fácil de utilizar.

**Formulario para agregar persona**

Con este formulario cargamos los datos solicitados por el programa y seleccionamos la dependencia mediante un selector desplegable llamado combo box.

**MANUAL DE AYUDA**

**FUENTES BIBLIOGRAFICAS**

* [**https://pypi.org/project/PyQt5/**](https://pypi.org/project/PyQt5/)
* [**https://www.riverbankcomputing.com/static/Docs/PyQt5/**](https://www.riverbankcomputing.com/static/Docs/PyQt5/)
* [**https://doc.qt.io**](https://doc.qt.io)
* [**https://graphviz.org**](https://graphviz.org)
* [**https://openai.com/blog/chatgpt**](https://openai.com/blog/chatgpt)
* [**https://iconos8.es**](https://iconos8.es)
* [**https://www.youtube.com/@codefirstwithhala**](https://www.youtube.com/@codefirstwithhala)
* [**https://www.youtube.com/@SpinnTV**](https://www.youtube.com/@SpinnTV)
* [**https://www.youtube.com/playlist?list=PL2PZw96yQChwe5ZrLoRjf8kjTYd3KKzKZ**](https://www.youtube.com/playlist?list=PL2PZw96yQChwe5ZrLoRjf8kjTYd3KKzKZ)
* [**https://python.readthedocs.io/en/latest/library/sqlite3.html**](https://python.readthedocs.io/en/latest/library/sqlite3.html)