Cristian David Rodríguez – 20171020072

Luis Felipe Corredor Espinosa – 20171020056

José David Sanabria Aponte - 20171020044.

Universidad Distrital Francisco José de Caldas

Facultad de Ingeniería – Ingeniería de Sistemas

Tkinter

Grupo 20-082(octubre 2020)

Tkinter Es una librería que proporciona a las aplicaciones de Python una interfaz de usuario fácil de programar, el paquete Tkinter está incluido en Python como un paquete estándar, por lo que no es necesario instalar nada para usarlo.

Hace parte a su vez de la biblioteca gráfica [Tcl/Tk](https://es.wikipedia.org/wiki/Tcl/Tk) para el lenguaje de programación Python. A pesar de su larga historia, su uso no está demasiado extendido entre los usuarios de equipos personales porque su integración visual con los sistemas operativos no era buena y proporcionaba pocos **widgets** (componentes) para construir los programas gráficos.  
  
Sin embargo, a partir de Tkinter 8.5 la situación dio un giro de ciento ochenta grados en lo que se refiere a integración visual, mejorando en este aspecto notablemente; también en el número de widgets que se incluyen y en la posibilidad de trabajar con estilos y temas, que permiten ahora personalizar totalmente la estética de un programa. Por ello, ahora Tkinter es una alternativa atractiva y tan recomendable como otras.

# **Tk/TCL**

Tk/Tcl ha sido durante mucho tiempo una parte integral de Python. Proporciona un conjunto de herramientas robusto e independiente de la plataforma para administrar ventanas. Disponible para desarrolladores a través del paquete [tkinter](https://docs.python.org/es/3/library/tkinter.html#module-tkinter) y sus extensiones, los módulos [tkinter.tix](https://docs.python.org/es/3/library/tkinter.tix.html#module-tkinter.tix) y [tkinter.ttk](https://docs.python.org/es/3/library/tkinter.ttk.html#module-tkinter.ttk).

El paquete [tkinter](https://docs.python.org/es/3/library/tkinter.html#module-tkinter) es una capa delgada orientada a objetos encima de Tcl/Tk. Para usar el módulo [tkinter](https://docs.python.org/es/3/library/tkinter.html#module-tkinter), no tiene que escribir ningún código Tcl, pero deberá consultar la documentación de Tk , y ocasionalmente, la documentación de Tcl. El módulo [tkinter](https://docs.python.org/es/3/library/tkinter.html#module-tkinter) es un conjunto de funciones que envuelven las implementaciones widgets Tk como clases de Python.

Las ventajas de los módulos [tkinter](https://docs.python.org/es/3/library/tkinter.html" \l "module-tkinter" \o "tkinter: Interface to Tcl/Tk for graphical user interfaces) son su velocidad y que generalmente se suministra de forma nativa con Python. Aunque su documentación es un poco deficiente.

# **Paso inicial**

Para poder empezar a usar tkinter para desarrollar la interfaz gráfica de un aplicativo lo primero que se debe hacer es importar el paquete Tkinter de la biblioteca Tk y posteriormente sus componentes en general.



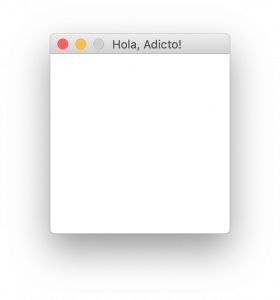
Otra opción es importar uno a uno los componentes que se van a usar en el aplicativo.



# **Componentes**

Para armar una vista con Tkinter, se usan componentes jerarquizados, que hacen parte de la interfaz que se está desarrollando. Algunos de los más comunes son:

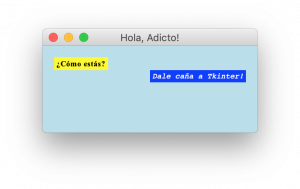
* *Tk*: es la raíz de la interfaz, donde vamos a colocar el resto de componentes.



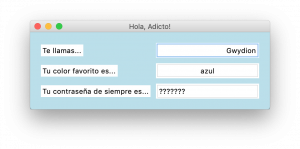
* *Frame*: marco que permite agrupar diferentes componentes.



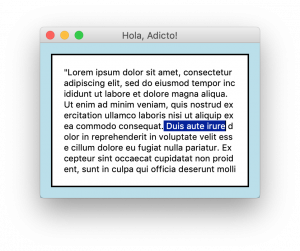
* *Label*: etiqueta estática que permite mostrar texto o imagen.



* *Entry*: etiqueta que permite introducir texto corto (típico de formularios).



* *Text*: campo que permite introducir texto largo (típico para añadir comentarios).



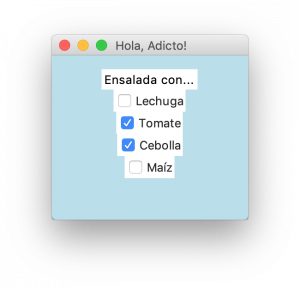
* *Button*: ejecuta una función al ser pulsado.



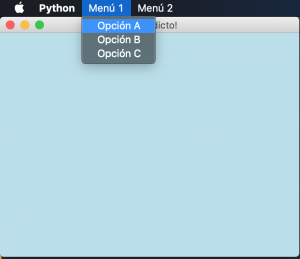
* *Radiobutton*: permite elegir una opción entre varias.



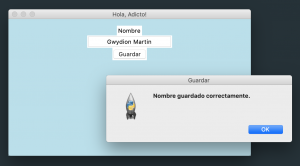
* *Checkbutton*: permite elegir varias de las opciones propuestas.



* *Menu*: clásico menú superior con opciones (Archivo, Editar…).



* *Dialogs*: ventana emergente (o *pop-up*).



# **Opciones de los componentes**

Para configurar un componente, simplemente se usa la declaración .config() junto con los argumentos que se quieran modificar. Algunas opciones son:

* *bg*: modifica el color de fondo. Se puede indicar con el color en inglés (incluyendo modificadores, como “darkgreen”) o su código RGB en hexadecimal (“#aaaaaa” para blanco).
* *fg*: cambia el color del texto.
* *cursor*: modifica la forma del cursor. Algunos de los más utilizados son “gumby”, “pencil”, “watch” o “cross”.
* *height*: altura en líneas del componente.
* *width*: anchura en caracteres del componente.
* *font*: nos permite especificar, en una tupla con nombre de la fuente, tamaño y estilo, la fuente a utilizar en el texto del componente. Por ejemplo, Font(“Times New Roman”, 24, “bold underline”).
* *bd*: modificamos la anchura del borde del widget.
* *relief*: cambiamos el estilo del borde del componente. Su valor puede ser “flat”, “sunken”, “raised”, “groove”, “solid” o “ridge”.
* *state*: permite deshabilitar el componente (state=DISABLED); por ejemplo, una *Label* en la que no se puede escribir o un *Button* que no se puede clickar.
* *padding*: espacio en blanco alrededor del widget en cuestión.
* *command*: de cara a que los botones hagan cosas, podemos indicar qué función ejecutar cuando se haga click en el mismo.

# **Colocación de los componentes**

Para ajustar la colación de un componente es necesario usar un gestor de geometría de componentes. Si no, el widget quedará creado, pero no se mostrará. Los tres más conocidos son:

* *Pack*: cuando añadimos un nuevo componente, se “hace hueco” a continuación de los que ya están incluidos (podemos indicar que se inserte en cualquiera de las 4 direcciones), para finalmente calcular el tamaño que necesita el widget padre para contenerlos a todos.
* *Place*: este es el más sencillo de entender, pero puede que no el más sencillo de utilizar para todo el mundo. Al insertar un componente, podemos indicar explícitamente la posición (coordenadas X e Y) dentro del widget padre, ya sea en términos absolutos o relativos.
* *Grid*: la disposición de los elementos es una matriz, de manera que para cada uno debemos indicar la celda (fila y columna) que queremos que ocupe. Podemos además especificar que ocupe más de una fila y/o columna (rowspan/columnspan=3), “pegarlo” a cualquiera de los 4 bordes de la celda en vez de centrarlo (sticky=W para el borde izquierdo).

**Referencias**

[1] [tkinter](https://docs.python.org/3/library/tkinter.html#module-tkinter) — Python interface to Tcl/Tk, [En línea]. Disponible en: <https://docs.python.org/3/library/tkinter.html>

[2] Interfaces gráficas de usuario con Tk, [En línea]. Disponible en: <https://docs.python.org/es/3/library/tk.html>

[3] Interfaces gráficas en Python con Tkinter, [En línea]. Disponible en: <https://www.adictosaltrabajo.com/2020/06/30/interfaces-graficas-en-python-con-tkinter/>